

# IT2School

Gemeinsam IT entdecken



## IT2School - Basismodule

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Impressum IT2School

## Herausgeber:

Wissensfabrik – Unternehmen für Deutschland e.V.  
Ruthenstr. 23  
67063 Ludwigshafen  
www.wissensfabrik.de

## Konzepterstellung und Umsetzung:

OFFIS e.V. – Institut für Informatik  
Carl von Ossietzky Universität Oldenburg  
Prof. Dr. Ira Diethelm  
Melanie Schaumburg  
Anatolij Fandrich

Nils Pancratz  
Mirko Janssen  
Mareike Daeglau  
Annette Diruf  
Estherk Nabo

Für inhaltliche Anregungen danken wir dem Arbeitskreis Bildung der Wissensfabrik – Unternehmen für Deutschland e.V. im Speziellen der Projektgruppe sowie allen Lehrkräften und Unternehmensvertreter\*innen, die aktiv an der Pilotphase beteiligt waren oder uns Feedback gegeben haben:

## Projektgruppe:

Leitung: Dr. Franziska Hutzler (Wissensfabrik)  
Christiane Bauer (SAP)  
Siegfried Czock (Bosch)  
Michael Detmer (Wissensfabrik)  
Matthias Dietel (IBM) Christian Greger (Trumpf)  
Stefan Hüppe (Böhringer Ingelheim)  
Axel Jentzsch (Wissensfabrik)  
Peter Kusterer (IBM)  
Reinhard Pittschellis (Festo Didactic)  
Markus Riefling (BASF)  
Ingmar Sassmann (BASF)  
Thomas Schmitt (Deutsche Telekom Stiftung)  
Birgit Schmitz (Deutsche Telekom Stiftung)  
Andreas Schneider (Trumpf)  
Peter Schubert (Softwarekontor)

Steffi Feldhaus (Berufskolleg Kohlstraße, Wuppertal)  
Daniel Jungblut (SAP)  
Martin Kempa (Gesamtschule Melsungen)  
Markus Knak (Graf-Anton-Günther Gymnasium Oldenburg)  
Bernadette Krüger (Oberschule Lemwerder)  
Torsten Klaus (Trumpf)  
Hannes Koderisch (Privatgymnasium Schwetzingen)  
Harald Rothkirch (Gymnasium Neue Oberschule, Braunschweig)  
Eva Nickel (Softwarekontor)  
Gerburg Lubor (Softwarekontor)  
Klaus-Dieter Neff (Leonardo da Vinci Gymnasium Neckargemünd)  
Frank Röhr (Erich-Kästner-Gesamtschule Bochum)  
Carsten Rohe (Gymnasium Damme)  
Christiane Schicke (Inselschule Langeoog)  
Tobias Stuckenberg (Paulusschule Oldenburg)  
Armin Tischler (Gymnasium Damme)  
Holger de Vries (KGS Rastede)  
Prof. Dr. Carsten Schulte (Uni Paderborn)  
Benjamin Piétza (FU Berlin)

## Pilotierung:

Torsten Barth (Gemeinschaftsschule Lauenburgische Seen)  
Nadine Bergner (RWTH Aachen - Schülerlabor Infosphere)  
Eric Böhmfeld (Dräger)  
Miriam Böhnke (SAP)  
Jens Eschen (Realschule Rhaderfehn)

## Urheber- und Nutzungsrechte:

Dieses Werk bzw. Inhalt steht unter einer Creative Commons Lizenz (**Namensnennung, nichtkommerziell, Weitergabe unter gleichen Bedingungen**).



Darüber hinaus ist die Nutzung an Privatschulen und Bildungseinrichtungen freier Träger gestattet, auch wenn dafür Gebühren erhoben werden, solange die Materialien im Unterricht oder in Betreuungsangeboten für Schulkinder verwendet werden. Gleiches gilt für

Fortbildungsangebote für Lehrkräfte. Alle Teile dieses Werkes sind vom Herausgebenden und von der für die Erstellung verantwortlichen Redaktion sorgfältig erwogen und geprüft worden. Eine Haftung des Herausgebenden bzw. der für die Redaktion verantwortlichen Institutionen für etwaige Personen-, Sach- oder Vermögensschaden, die sich aus dem Gebrauch dieses Werkes ergeben oder ergeben könnten, ist ausgeschlossen. In diesem Handbuch werden geschlechtsumfassende Formulierungen und der Genderstern verwendet. Sollte dies aus Gründen der Lesbarkeit an manchen Stellen nicht möglich sein, gelten die dort verwendeten Personenbezeichnung gleichermaßen für alle Geschlechter.

## Bildnachweise:

Die Bildnachweise sind jeweils neben dem Bild angegeben. Ist dies nicht der Fall, stammen die Bilder von [www.pixabay.com](http://www.pixabay.com) (Creative Commons - CC0) oder die Rechte liegen bei den Entwicklern des Konzeptes und der Wissensfabrik. Grafische Gestaltung: [www.active-screen.de](http://www.active-screen.de) Illustration: Christoph J. Kellner, Animation / Illustration / Graphic Recording, studio animanova

## Inhalt Ordner Basismodule

### **Einleitung**

#### **Modul B1 – Blinzeln**

Vom Blinzeln zum Verschlüsseln

#### **Modul B2 – Internet**

Die Internetversteher

#### **Modul B3 – Codes**

Codes im Supermarkt und Unternehmen

#### **Modul B4 – 3D-Druck**

3D-Druck, Modellierung und Augmented/Virtual Reality

#### **Modul B5 – Programmieren**

Leichter ProgrammierEinstieg

#### **Modul B6 – Mein Anschluss**

MocoMoco – Mein besonderer Anschluss

#### **Modul B7 – Meine App**

App Inventor

#### **Modul B8 – Calliope Mini**

Der Calliope mini Mikrocontroller

## Online verfügbare Aufbau-, Erweiterungs- und Methodenmodule

#### **Modul A1 – Mobilfunk**

Vom Mobilfunk zu Big Data

#### **Modul A2 – Kryptologie**

Kryptologie

#### **Modul A3 – Programmieren II**

Objektorientierte Programmierung mit Python

#### **Modul E1 – IT Kinderleicht**

IT und Informatik spielend entdecken

#### **Modul E2 – Wearable**

Smarte Kleidung selbst gestalten

#### **Modul E3 – Robotik**

BB8 selber bauen

#### **Modul E4 – Webseiten**

Erstellung von Webseiten

#### **Modul M1 – Design Thinking**

#### **Modul M2 – Projektmethode**

Die hier aufgeführten (klassischen) Module von IT2School werden durch die (neuen) Module zum Thema künstliche Intelligenz ergänzt. Diese sind im Ordner „IT2School Module zur künstlichen Intelligenz“ und online verfügbar.

# IT2School

Gemeinsam IT entdecken



## Einleitung

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Inhalt

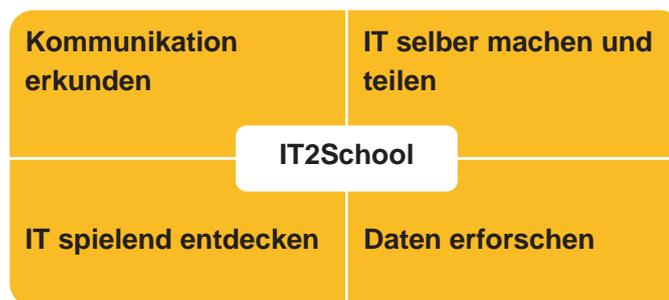
1	Einführung.....	3
2	Die Module von IT2School - eine Übersicht.....	4
3	Aufbau der Module.....	7
3.1	Einleitung des Moduls.....	9
3.2	Bedeutung und Ziele des Moduls.....	9
3.3	Rolle der Unternehmensvertreter*innen.....	9
3.4	Inhalte des Moduls.....	9
3.5	Unterrichtliche Umsetzung .....	10
3.6	Einbettung in verschiedene Fächer und Themen .....	10
3.7	Anschlussthemen, Literatur und Links .....	10
4	Kommentare zu den Arbeitsmaterialien .....	10
5	Die Akteure des Projekts und ihre Rollen.....	11
6	Bestandteile von IT2School-Infobox.....	12



# 1 Einführung

Informatik ist im Allgemeinen die Wissenschaft von der automatischen Verarbeitung von Information und Daten mit speziellen Systemen, die oft unter dem Begriff Informationstechnologie (IT) zusammengefasst werden. Diese Systeme sind überall, oft aber versteckt. In diesem Projekt können Schüler\*innen unterschiedlicher Schulstufen Informatik und IT suchen, erforschen, ausprobieren und spielend entdecken. Sie werden so zu Expertinnen und Experten, die IT kreativ einsetzen, Neues erfinden und eigene Ideen umsetzen. Ziel ist es, Informatik auf spielerische Weise zu vermitteln. Kinder und Jugendliche sollen Informatiksysteme verstehen und hinterfragen, aktiv gestalten und selbstbewusst mit ihr interagieren.

Hierfür stellt das Projekt IT2School der Wissensfabrik mehrere Module zur Verfügung, die in vier Cluster unterteilt sind. Innerhalb dieser Cluster stehen sieben Basismodule und drei Aufbaumodule einsatzbereit zur Verfügung.



In den Modulen werden die grundlegenden Themen der Informatik wie Kommunikation, Programmierung, Verständnis von Netzwerken, Zusammenspiel von Hard- und Softwarekomponenten und Darstellung von Informationen in den Blick genommen und für die Klassenstufen 4 bis 10 (Basismodule) bzw. 7 bis 13 (Aufbaumodule) aufbereitet. Bezüglich des Schwierigkeitsgrades, des Aufwandes oder der Einsatzdauer im Unterricht lassen sich die Module wie folgt einstufen:

**Basis:** Ohne tiefgehende Vorkenntnisse auf Seiten der Lehrkräfte leicht durchführbar; Schnelles Erfolgserlebnis sowohl für Lehrkräfte als auch für Schüler\*innen.

**Aufbau:** Mehr Vorbereitung oder Vorkenntnisse von Lehrkräften nötig; beschäftigt sich mit komplexeren Themen der Informatik und erzeugt längeres sowie tieferes Verständnis bei den Schüler\*innen.

**Erweiterung:** Die Erweiterungsmodule bieten Ideen für weitere Themen und Projekte. Zum Teil werden die Module von einzelnen Unternehmen bereitgestellt.

Die Basismodule können flexibel nach den Wünschen und Interessen der Lehrkräfte, Schüler\*innen sowie der beteiligten Unternehmensvertreter\*innen zusammengestellt werden. Es stehen sowohl Module bereit, die weitgehend ohne technischen Einsatz durchführbar sind, als auch solche, die sehr software- oder hardwareaffin sind. Auch der Einbezug der Unternehmensvertreter\*innen ist flexibel gestaltbar. Die Aufbaumodule besitzen einen mehr oder weniger starken Bezug zu den Basismodulen und können im Anschluss durchgeführt werden. Dabei steht Ihnen auch bei den Aufbaumodulen genügend Spielraum für eigene Ideen und Wünsche zur Verfügung.

Im Unterschied zu anderen Aktivitäten zur Förderung von IT und Informatik in Schulen, die sich oft auf einen Aspekt (z. B. CS unplugged, Programmieren, Physical Computing oder Computational Thinking) konzentrieren oder nur Materialien bereitstellen, nimmt dieses Projekt die IT ganzheitlich in den Blick und bringt gemeinsam mit den Unternehmensvertreter\*innen die verschiedenen thematischen Facetten von Informatik in die Schule.

## 2 Die Module von IT2School - eine Übersicht

Basismodule			
Nr.	Titel	Thema	Klassenstufe
B1	Vom Blinzeln zum Verschlüsseln	In diesem Modul entdecken die Schüler*innen die analogen Wurzeln der IT: Sie erstellen ein Blinzelprotokoll, haben die Möglichkeit einen Morseapparat und ein Dosentelefon zu bauen und befassen sich mit der Übertragung und Verschlüsselung von Informationen.	4. - 10. Klasse
B2	Die Internetversteher	Dieses Modul erklärt altersgerecht die Funktionsweise des Internets anhand von Pappmodellen und einem Rollenspiel.	4. – 10. Klasse
B3	Codes im Supermarkt und Unternehmen	Dieses Modul befasst sich mit der Funktionsweise und den Einsatzmöglichkeiten von optischen Codes. Die Schüler*innen erfahren, wo sie überall zu finden sind, wie man sie erstellt und welche Informationen sie bereitstellen.	7. - 10. Klasse
B4	3D-Druck, Modellierung und Augmented/Virtual Reality	In diesem Modul lernen die Schüler*innen nicht nur die Möglichkeiten des 3D-Drucks kennen, sondern können selbst eigene Figuren modellieren und erhalten die Möglichkeit diese auszudrucken.	4. – 10. Klasse
B5	Leichter ProgrammierEinstieg	Dieses Modul liefert einen Einstieg in die Programmierumgebung „Scratch“. Sie ist speziell für Kinder und Jugendliche entwickelt worden und bietet daher einen einfachen und intuitiven Einstieg in das Programmieren.  <b>Hinweis:</b> Wir empfehlen dringend dieses Modul durch B6 zu erweitern, da die Schüler*innen so weitere kreative Möglichkeiten zur Entwicklung eigener Projekte erhalten.	4. - 10. Klasse
B6	MocoMoco – Mein besonderer Anschluss	In diesem Modul erfahren die Schüler*innen, wie Eingabegeräte funktionieren. Mithilfe des mitgelieferten Controllers können in Sekundenschnelle leitende Alltagsgegenstände als Eingabegeräte an den Computer angeschlossen werden - sogar Mitschülerinnen und Mitschüler oder Lehrkräfte. So tun sich viele kreative Möglichkeiten auf, Eingabegeräte selbst zu gestalten und anschließend dafür eigenständig Programme zu schreiben.	4. - 10. Klasse
B7	Meine App – App	Die Schüler*innen haben die Möglichkeit mit	7. - 10.Klasse

	Inventor	der Entwicklungsumgebung App Inventor, eine eigene kleine App für Android-Smartphones zu programmieren.	
B8	Der Calliope mini Mikrocontroller	Im Rahmen dieses Moduls lernen die Schüler*innen die Hardware des Calliope mini näher kennen und befassen sich spielerisch mit dem Programmieren.	4. - 10. Klasse

<b>Aufbaumodule</b>			
Nr.	Titel	Thema	Klassenstufe
A1	Vom Mobilfunk zu Big Data	Dieses Modul kann direkt auf die Inhalte des Basismoduls B2 aufbauen und erklärt inhaltlich zum einen die Funktionsweise des Mobilfunks und zum anderen die Möglichkeiten zur Verarbeitung und Auswertung von Mobilfunkdaten. Hierdurch ergibt sich ein anschaulicher Übergang vom Mobilfunk zu Big Data und den gesellschaftlichen Auswirkungen von IT im Alltag.	7. - 13. Klasse
A2	Kryptologie	In diesem Modul lernen die Schüler*innen Möglichkeiten und Anwendungsbereiche der Kryptologie kennen. Neu erworbenes Wissen wird dabei praktisch angewendet und die Schüler*innen können zum Abschluss ihre eigenen Daten und Kommunikationswege absichern.	7. - 13. Klasse
A3	Programmieren II	Eine Vertiefung der gemachten Programmiererfahrungen anhand der Basismodule B5 + B6 und B7 kann mit diesem Modul erfolgen. Die Schüler*innen beschäftigen sich spielerisch mit der textuellen Programmierung in Python und können sich in eigenen Projekten kreativ entfalten.	10. - 13. Klasse

<b>Erweiterungsmodule</b>			
Nr.	Titel	Thema	Klassenstufe
E1	IT kinderleicht	Dieses Erweiterungsmodul stellt verschiedene Möglichkeiten vor, wie auch schon jüngere Kinder auf spielerische Weise IT entdecken können. Hierfür bieten sich beispielsweise die kleine Roboter oder Littlebits an. Die kleinen, elektronischen Bauelemente lassen sich magnetisch miteinander verbinden, wodurch einfache Schaltkreise entstehen. Hierdurch	Kita oder Vorschule bis ca. 6. Klasse

		wird ein spielerischer Zugang zu den Grundlagen der Elektronik und Programmierung ermöglicht.	
E2	Wearables	In diesem Modul befassen sich die Schüler*innen mit tragbaren und interaktiven Systemen und Controllern, wie sie beispielsweise in smarten Kleidungsstücken und Accessoires oder Smart Watches und Fitness Trackern Anwendung finden. Sie haben die Möglichkeit in eigenen Projekten selbst smarte Kleidung zu designen und zu erstellen.	8. - 13. Klasse
E3	Robotik	In diesem Modul erhalten die Schüler*innen eine Anleitung, wie der kleine Roboter selbst gebaut werden kann, der einem BB8 aus StarWars ähnelt. Mithilfe von verschiedensten Materialien wie Styroporkugeln und Pappmaché, sowie einem BlueCoLight-Controller oder einem Arduino-Board kann der Roboter programmiert und mithilfe einer programmierten App über das Smartphone gesteuert werden.	6. – 13. Klasse
E4	Webseiten	Es existieren einige einfach zu bedienende Online-Editoren zur Erstellung von Webseiten. Die Schüler*innen können damit nicht nur Webseiten mittels HTML (Hypertext-Markup-Language) erzeugen, sondern auch online abspeichern und veröffentlichen. Dabei steht ihnen während der Nutzung meist eine Live-Vorschau zur Verfügung. Dies bietet den Vorteil, dass die Schüler*innen nicht nur unter Anleitung, sondern auch experimentell vorgehen können.	6. – 13. Klasse

<b>Methodenmodule</b>			
<b>Nr.</b>	<b>Titel</b>	<b>Thema</b>	<b>Klassenstufe</b>
M1	Design Thinking	Dieses Modul führt in die Innovationsmethode Design Thinking ein. Mit Hilfe dieser Methode lassen sich neue Produkte und innovative Ideen umsetzen.	4. - 13. Klasse
M2	Projektmethode	Dieses Modul führt in den projektorientierten Unterricht ein. Dies ist eine Form des handlungsorientierten Unterrichts bei dem Schüler*innen eine Aufgabe oder eine Problemstellung selbständig - von der Planung	4. - 13. Klasse

über die Durchführung bis zur Präsentation - innerhalb einer Gruppe bearbeiten.

Die aktuellen Materialien von IT2School stehen kostenfrei online auf der Webseite der Wissensfabrik zur Verfügung:

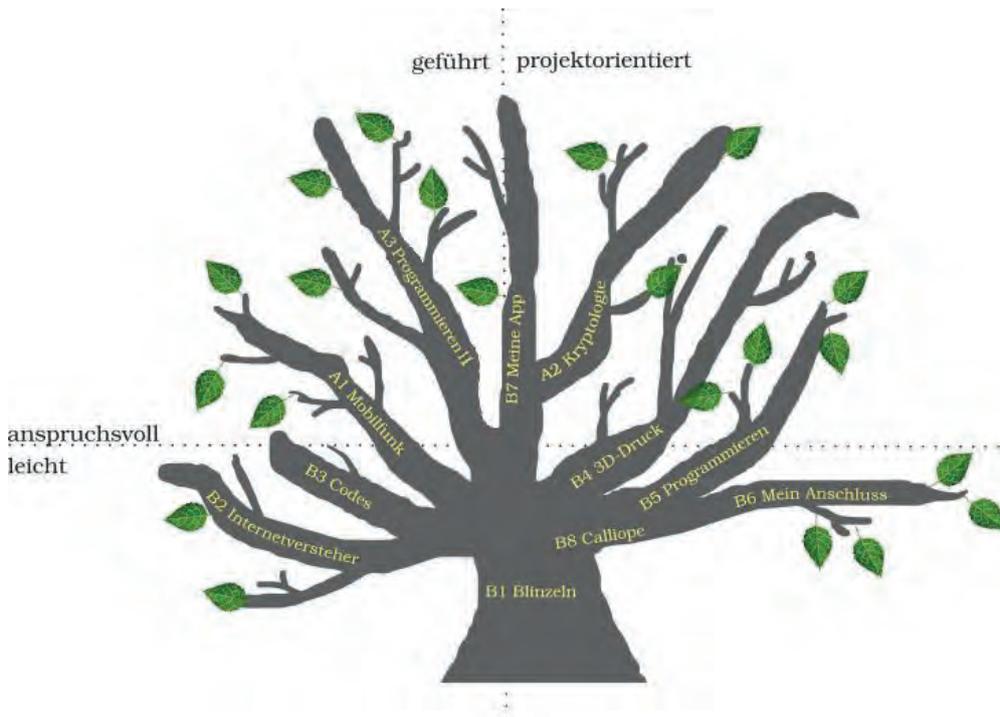


<https://www.wissensfabrik.de/downloadmaterial-it2school/>

Die Basismodule sind Inhalt dieses Ordners. Die Aufbaumodule, Erweiterungsmodule und Methodenmodule stehen ausschließlich online zur Verfügung. Die hier aufgeführten (klassischen) Module von IT2School werden durch die (neuen) Module zum Thema künstliche Intelligenz ergänzt. Diese sind im Ordner „IT2School Module zur künstlichen Intelligenz“ und online verfügbar.

### 3 Aufbau der Module

Das Modul *B1 - Vom Blinzeln zum Verschlüsseln* bildet die Basis für alle Module. Es bietet einen einfachen und vor allem analogen Einstieg in die Grundlagen der Informatik. Einige der Module sind sowohl für die Schüler\*innen als auch für die Lehrkraft einfach in der Umsetzung und können ohne weitere Vorkenntnisse durchgeführt werden. Andere Module benötigen tiefergehendes Wissen seitens der Lehrkraft, beispielsweise im Umgang mit dem *App Inventor* in Modul B7. Zudem unterscheiden sich die Module in ihrem methodischen Charakter: Einige sind durch Arbeitsblätter klar strukturiert, andere offen und projektorientiert.



Je nachdem, wie der Unterricht gestaltet werden soll (geführt oder projektorientiert) und wie umfassend die Kenntnisse im Bereich der Informatik bereits sind, können passgenaue Module ausgewählt werden. Daraus ergeben sich ganz unterschiedliche mögliche Verläufe. Hier einige Beispiele:

### Beispiel 1: Leichter Einstieg ohne Technik

Haben Sie selbst noch wenig Kenntnisse im Bereich der Informatik oder haben Sie vielleicht keine gute technische Ausstattung, dann können Sie folgenden Verlauf nehmen:



Diese drei Module benötigen keinen Computer und sind besonders für einen leichten Einstieg zu empfehlen.

### Beispiel 2: Projektwoche

Wenn Sie keinen wöchentlichen Informatikunterricht haben, besteht die Möglichkeit einzelne Module auch als Projektwoche anzubieten. Dabei können einige Module auch kombiniert werden, wie in diesem Beispiel:



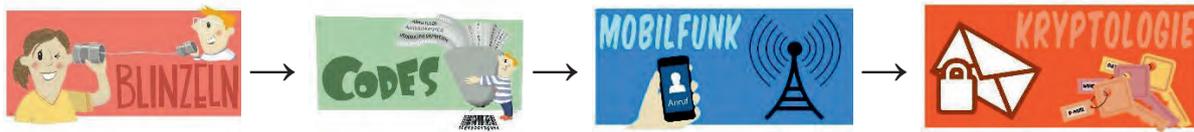
### Beispiel 3: Thema Programmieren

Möchten Sie einen Schwerpunkt auf das Thema Programmieren legen, ist folgender Verlauf denkbar:



### Beispiel 4: Daten, Information und Sicherheit

Wenn Sie sich eher mit der Darstellung und Verwendung von Daten und Informationen in informatischen Systemen beschäftigen wollen und auch das Thema der Daten- und Informationssicherheit als wichtig empfinden, dann bietet sich folgender Verlauf an:



### Beispiel 5: IT spielend entdecken

Soll in Ihrem Unterricht das spielerische Entdecken und das kreative Gestalten im Mittelpunkt stehen, dann können Sie folgende Module auswählen:



Im Folgenden wird der Aufbau aller Basis- und Aufbaumodule beschrieben.

### 3.1 Einleitung des Moduls

Jede Modulbeschreibung wird mit einem Steckbrief eingeleitet, dem die wichtigsten Kerninformationen entnommen werden können. So wird der inhaltliche Anspruch des Moduls umrissen und es finden sich hier Erläuterungen zur Zielgruppe beziehungsweise der empfohlenen Klassenstufe, zum geschätzten Zeitaufwand, zu Lernzielen, Vorkenntnissen und Voraussetzungen sowie dem benötigten Material.

### 3.2 Bedeutung und Ziele des Moduls

Bedeutsamkeit und Begründung des Themas sowie dessen Zielsetzungen stellen wesentliche Aspekte bei der Planung des Unterrichts dar. Lehrkräfte erfahren daher in diesem Abschnitt mehr über die wichtigsten Inhalte und Alltagsbezüge des Moduls, sodass es schnell und einfach mit den Bildungsplänen abgeglichen werden kann.

### 3.3 Rolle der Unternehmensvertreter\*innen

Die Einbindung der Unternehmensvertreter\*innen kann bei den verschiedenen Modulen variieren. Daher werden in diesem Abschnitt verschiedene Möglichkeiten erörtert. In welcher Form die Einbindung des Unternehmensvertretenden in den Unterricht letztlich erfolgt, sollte vor Projektstart gemeinsam besprochen werden (siehe auch Kapitel 5).

### 3.4 Inhalte des Moduls

Um die Orientierung über mögliche Unterrichtsverläufe zu erleichtern, erklären und gliedern wir hier das Thema kurz und bündig. Das verschafft einen Gesamtüberblick und ermöglicht es, gezielter in die Module einzusteigen.

### 3.5 Unterrichtsliche Umsetzung

Die methodisch-didaktischen Hinweise geben einen Überblick, wie das Modul im Unterricht umgesetzt werden kann. Folgende Informationen sind hier zu finden:

- zum benötigten Zeitrahmen,
- zu geeigneten Sozialformen und Methoden,
- zur Einbindung der Unternehmensvertreter\*innen,
- eine exemplarische Beschreibung des Unterrichtsgeschehens sowie
- eine Übersicht über das benötigte Material.

Dieser tabellarische Überblick hilft, die Planung schnell und einfach vorzunehmen.

### 3.6 Einbettung in verschiedene Fächer und Themen

Auch wenn eine direkte Anbindung von IT2School an die verschiedenen Curricula der Bundesländer nicht leistbar ist, haben wir versucht, die wichtigsten Kompetenzfelder der Module zu extrahieren und mit den Bildungsstandards und Curricula verschiedener Bundesländer abzugleichen. Das Ergebnis findet sich in diesem Abschnitt wieder und zeigt Möglichkeiten auf, wie die IT-Themen (auch außerhalb des Informatikunterrichts in anderen Fächern) eingebunden werden können.

### 3.7 Anschluss Themen, Literatur und Links

Am Ende jeder Modulbeschreibung finden sich Anregungen für eine Vertiefung oder Weiterführung der Themenstellung. So verweisen wir auf IT2School-Module, die sich als Anschlussmodule eignen und wir liefern allgemeine Vorschläge für den weiteren Unterrichtsverlauf. Darüber hinaus gibt es zahlreiche Links und Literaturtipps mit weiterführenden Informationen.

## 4 Kommentare zu den Arbeitsmaterialien

Die Arbeitsmaterialien helfen Schüler\*innen, sich eine Thematik zu erarbeiten und leiten mit Arbeitsaufträgen zur kreativen Umsetzung und Gestaltung an. Alle Arbeitsmaterialien für Schüler\*innen sind in der Kopfzeile **orange** gekennzeichnet. Die Arbeitsmaterialien für Lehrkräfte beinhalten weiterführende Informationen oder Lösungen zu den Aufgabenstellungen. Diese Arbeitsmaterialien sind **grün** gekennzeichnet. **Blau** gekennzeichnet sind Zusatzmaterialien, die von den Lehrkräften als Alternativen zu den vorgeschlagenen Materialien genutzt werden können.

Weitere Kennzeichnungen betreffen die Schulstufe: So sind Arbeitsblätter, die für die Grundschule konzipiert wurden, an der Abkürzung **GS** zu erkennen, die für die Sekundarstufe tragen die Abkürzung **Sek. I**. Bei allen Arbeitsmaterialien ohne eine solche Kennzeichnung kann anhand des Leistungsstandes der Klasse selbst entschieden werden, ob die Materialien geeignet sind. Die Kennzeichnungen **SuS** (für Schüler\*innen) und **L** (für Lehrkräfte) werden gebraucht, weil bei einigen Arbeitsmaterialien selbst bei gleicher Benennung eine Differenzierung zwischen Lehrkräften und Schüler\*innen notwendig ist.

## Legende



Material für Schüler\*innen (GS/ Sek I)



Material für Lehrkräfte sowie Unternehmensvertreter\*innen



Zusatzmaterial

## 5 Die Akteure des Projekts und ihre Rollen

Im Rahmen des Leuchtturmprojekts IT2School der Wissensfabrik können zwischen Schulen und Unternehmen Bildungspartnerschaften abgeschlossen werden. Die Ausgestaltung dieser Bildungspartnerschaft kann sehr individuell sein, gleichwohl ist es hilfreich die Rollen, die die Projektbeteiligten einnehmen können, genau zu definieren und abzugrenzen.

Im Vordergrund stehen die **Lernenden**. Ihnen wird in diesem Projekt Raum gegeben, IT und die Grundlagen der Informatik zu entdecken und spielerisch auszuprobieren. Darüber hinaus besteht durch die gemeinsame Umsetzung des Projekts mit einem Unternehmen die Möglichkeit, die Arbeitswelt von heute und deren reale IT-Anwendungen zu erkunden.

Die **Lehrkraft** sollte sich als Lernbegleitung verstehen. Sie schafft ein Umfeld, in dem Schüler\*innen die handlungsorientierten Projekte umsetzen können. Darüber hinaus wählt die Lehrkraft die Module und Inhalte passend zum Leistungs- und Wissensstand der Klasse aus.

Das Tandem aus **Lehrkraft** und **Unternehmensvertreter\*in** bestimmt gemeinsam den Umfang und die Ausgestaltung der Bildungspartnerschaft.

Die besondere Position **der Unternehmensvertreter\*innen** zeigt sich durch die vielen Rollen, die er oder sie übernehmen kann:

Als *Multiplikator\*innen* für die Inhalte von IT2School besuchen die Unternehmensvertreter\*innen Fortbildungen und gibt das Wissen an die Partnerschule weiter. In Absprache mit Lehrkräften der Partnerschule besteht zudem die Möglichkeit, dass die Unternehmensvertreter\*innen einen aktiven Part (*Co-Teacher*) bei der Umsetzung der Module in der Klasse übernehmen. Die Verantwortung für den Unterricht bleibt dabei stets bei der Lehrkraft.

Als *Special-Guest* kann ein\*e Unternehmensvertreter\*in im Unterricht über die IT-Praxis in der eigenen Firma berichten.

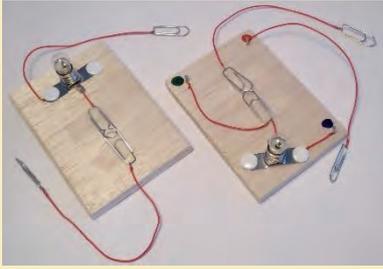
Als *Gastgeber\*in* kann das Unternehmen zu einzelnen Schwerpunkten gemeinsam mit der Lehrkraft die Schüler\*innen zu Exkursionen zum Unternehmen einladen.

Es wird empfohlen, dass Lehrkräfte an Fortbildungen zu IT2School teilnehmen. Die Fortbildung für die Lehrkraft wird dabei durch das Partnerunternehmen finanziert. Die Umsetzung des Projekts kann dann nach gemeinsamer Absprache mit den Unternehmensvertreter\*innen geplant werden.

Akteur*in	Rolle	Funktion/Aufgabe
Schüler*innen	<ul style="list-style-type: none"><li>Lernende</li></ul>	<ul style="list-style-type: none"><li>werden aktiv in die Planungen mit einbezogen</li><li>haben im Rahmen der Module die Möglichkeit,</li></ul>

		handlungsorientiert in Projekten IT zu entdecken
Lehrkraft	<ul style="list-style-type: none"> <li>• Lernbegleiter*in</li> </ul>	<ul style="list-style-type: none"> <li>• wählt die Inhalte entsprechend des Wissensstands der Klasse</li> <li>• unterstützt die Schüler*innen bei der Umsetzung und lässt ihnen Freiraum zum eigenständigen Entdecken von IT</li> </ul>
Unternehmensvertreter*in	<ul style="list-style-type: none"> <li>• Multiplikator*in</li> <li>• Co-Teacher</li> <li>• Special-Guest</li> <li>• Gastgeber*in</li> </ul>	<p>Je nach Rolle</p> <ul style="list-style-type: none"> <li>• schult Lehrkräfte</li> <li>• unterstützt Lehrkräfte</li> <li>• berichtet aus der IT-Praxis</li> <li>• ermöglicht Schüler*innen einen Einblick in die Arbeitswelt</li> <li>• ermöglicht Betriebserkundungen</li> </ul>

## 6 Bestandteile von IT2School-Infobox

Modul B1 Blinzeln		
Vom Blinzeln zum Verschlüsseln		
Material	Stückzahl <sup>1</sup>	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
<p>Morseapparat</p> <p>Material für einen einfachen Morseapparat (GS) oder einen Crosslink-Morseapparat (Sek I)</p>	<ul style="list-style-type: none"> <li>• 2 Holzbrettchen (ca. 10 x 4 cm)</li> <li>• 2 Lämpchen 4,5V</li> <li>• 2 Flachbatterien 4,5V</li> <li>• Klingeldraht</li> <li>• 4 Reißzwecken</li> <li>• 8 Büroklammern</li> <li>• 6 kleine Nägel (abhängig von der Dicke des Holzes)</li> <li>• Klebestreifen</li> </ul> <p><b>Klassensatz:</b> 10 x</p>	

<sup>1</sup> Bezogen auf die Infobox; Gesonderte Angabe für Klassensatz

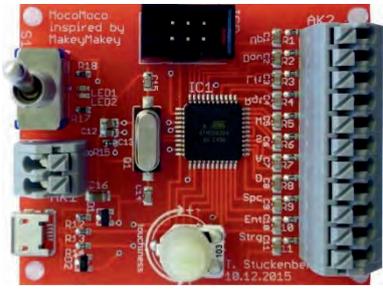
Modul B2 Internet Die Internetversteher		
Material	Stück	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
Pappaufsteller	<ul style="list-style-type: none"> <li>• 9 x Router</li> <li>• 1 x Webserver (Frag Finn)</li> <li>• 1 x Webserver (Schule)</li> <li>• 1 x DNS</li> <li>• 1 x Provider</li> <li>• 2 x Heim-Internetrouter</li> <li>• 4 x Client</li> </ul>	
Websites für Pappaufsteller	<ul style="list-style-type: none"> <li>• 1 x Schulhomepage</li> <li>• 1 x Frag Finn-Website</li> </ul>	
Stationskarten	<ul style="list-style-type: none"> <li>• 1 x Client</li> <li>• 1 x Provider</li> <li>• 1 x DNS</li> <li>• 1 x Heim-Internetrouter</li> <li>• 1 x Webserver Schule</li> <li>• 1 x Webserver Frag Finn</li> <li>• 20 x Router</li> </ul>	
Protokollheft	1 x	
Gruppenkärtchen	<ul style="list-style-type: none"> <li>• 5 x Router</li> <li>• 5 x Provider</li> <li>• 5 x DNS</li> <li>• 5 x Heim-Internetrouter</li> <li>• 5 x Webserver</li> <li>• 5 x Clients</li> </ul>	

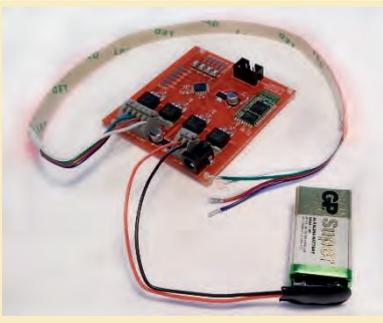
Modul B3 Codes Codes im Supermarkt und im Unternehmen		
Material	Stück	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
Miniprodukte	1 x <b>Klassensatz: 3 x</b>	

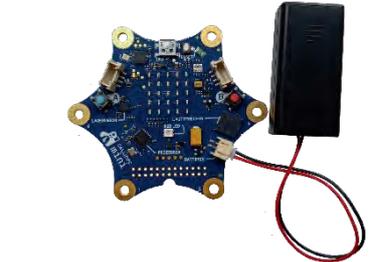
Modul B4 3D-Druck 3D-Modellierung und -Druck		
Material	Stück	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
VR-Brille	1 x <b>Klassensatz: 10 x</b>	

Modul B5 Programmieren Leichter ProgrammierEinstieg		
Material	Stück	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
Scratch-Würfel-Set	1 x Würfel-Set <b>Klassensatz: 10 x</b>	
Buch „Programmieren supereasy“	1 x	



Modul B6 Mein Anschluss Mein besonderer Anschluss		
Material	Stück	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
MocoMoco	1 x <b>Klassensatz: 5 x</b>	
Knete	Auswahl	

Modul B7 Meine App App Inventor		
Material	Stück	Bild
Modulbeschreibung inkl. Arbeitsmaterialien	1 x	
BlueCoLight mit LED-Lichterkette	1 x <b>Klassensatz: 5 x</b>	

Modul B8 Calliope mini Mein besonderer Anschluss		
Material	Stück	Bild
Calliope mini + Batteriehalter	1 x	

Micro-USB Kabel	1 x	
Knete	1 x	
Krokodilklemme	2 x	
LED	2 x	
Gummiband	1 x	
Kupferband	1 x	

# IT2School

Gemeinsam IT entdecken



## Modul B1 – Blinzeln

Vom Blinzeln zum Verschlüsseln

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Inhalt

1	Vom Blinzeln zum Verschlüsseln .....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls .....	4
4	Rolle der Unternehmensvertreter*innen .....	4
5	Inhalte des Moduls .....	5
6	Unterrichtliche Umsetzung .....	5
6.1	Grober Unterrichtsplan .....	6
6.2	Stundenverlaufsskizzen .....	7
6.2.1	Variante 1 .....	7
6.2.2	Variante 2 .....	8
7	Einbettung in verschiedene Fächer und Themen .....	11
8	Anschlussthemen .....	12
9	Literatur und Links .....	13
10	Arbeitsmaterialien .....	14
11	Glossar .....	14
12	FAQs und Feedback .....	15



# 1 Vom Blinzeln zum Verschlüsseln

In diesem Modul entdecken die Schülerinnen und Schüler die analogen Wurzeln der Informatik und Informationstechnologie (IT) und befassen sich mit den Grundlagen der digitalen Kommunikation. Es geht dabei insbesondere um die Codierung und Übertragung von Informationen.

Die Schülerinnen und Schüler entwickeln u. a. ein eigenes Blinzeln-Protokoll und erfahren auf diese einfache Weise, wie die Übertragung von Daten im Binärsystem funktioniert. Darüber hinaus bauen sie einen Morseapparat und befassen sich mit der Bildspeicherung und Darstellung durch den Computer. Dieses Modul ist sehr einfach durchzuführen, da keine digitalen Medien benötigt werden. Darüber hinaus ist es besonders schüleraktivierend und auch die Lehrkräfte können es ohne besondere Vorkenntnisse umsetzen.



<b>Lernfeld/Cluster:</b>	Kommunikation erkunden	
<b>Zielgruppe/Klassenstufe:</b>	X	4. bis 5. Klasse
	X	6. bis 7. Klasse
	X	8. bis 10. Klasse
		11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	3-8 Stunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Grundlagen der digitalen Kommunikation verstehen</li> <li>• Funktion von Algorithmen verstehen, eigene Protokolle entwerfen und entwickeln</li> <li>• Einsatzmöglichkeiten von Protokollen entdecken</li> <li>• Funktionsweise des Binärcodes verstehen</li> <li>• verschiedene Übertragungswege von Information kennenlernen</li> <li>• Grundlagen der Verschlüsselung kennenlernen</li> </ul>	
<b>Vorkenntnisse der Schüler*innen</b>	Keine	
<b>Vorkenntnisse der Lehrkraft:</b>	Keine	
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Keine	
<b>Sonstige Voraussetzungen:</b>	Keine	



## 2 Warum gibt es das Modul?

In diesem Modul erfahren die Schülerinnen und Schüler mehr über die Grundlagen der Informationstechnologie (IT), insbesondere der digitalen Kommunikation – ganz ohne Computer. Durch diesen analogen Einstieg ohne notwendige Vorkenntnisse ist die Hemmschwelle für alle Projektbeteiligten sehr gering. Sie entdecken die Inhalte gemeinsam und spielerisch in unterschiedlichsten Arbeitsformen. Die hier behandelten Grundlagen der Informatik und IT haben in unserer mediengeprägten Welt wesentliche Bedeutung. Kinder wachsen heute selbstverständlich damit auf, dass sie Fotos auf Bildschirmen hin und her bewegen, Nachrichten versenden oder Videos mit einem Klick erstellen können. Doch die wenigsten wissen, wie die Geräte funktionieren und wie die historischen Wurzeln dieser Errungenschaften aussehen.

Ziel dieses Moduls ist es daher, die Anfänge und die Grundlagen der (digitalen) Kommunikation nachvollziehbar und für Kinder und Jugendliche begreifbar zu machen. Die Kinder erwerben dabei nicht nur informatische Grundkompetenzen, sondern entwickeln auch handwerkliche Fähigkeiten, wie beispielsweise beim Bau eines Morse-Apparates. Zudem gewinnen sie Problemlöse-Kompetenzen beim Entwickeln eigener Protokolle und durch die Gruppenarbeit Sozialkompetenzen, wie Teamfähigkeit, Kooperation und Kommunikation.

## 3 Ziele des Moduls

- Grundlagen der (digitalen) Kommunikation kennenlernen
- Codierung von Information (Text oder Bilder) in Zeichen
- Sinn und Zweck von Codierung für Übertragungsmöglichkeiten kennenlernen
- Protokolle einsetzen und entwickeln
- Bildspeicherung und Darstellung durch Zahlen
- erste Verschlüsselungstechniken kennenlernen

## 4 Rolle der Unternehmensvertreter\*innen

Im *Modul B1 – Blinzeln* hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten aktiv mitzuwirken. Hier einige Anregungen:

- Unterstützung der Lehrkraft - Co-Teacher: Etwa beim Bau des Morse-Apparates oder beim Bau der Cäsar-Scheibe.
- Bericht aus dem Unternehmen – Special-Guest: Mögliches Thema: Wie hat sich die Kommunikation im Unternehmen im Lauf der Zeit verändert? (Hauspost, Telegramm, Faxgerät, E-Mail, ...). Ggf. zur Anschauung etwas mitbringen, wie beispielsweise ein altes Telefon mit Wählscheibe; bietet sich direkt nach der Einführung „Von den Anfängen der fernen Kommunikation“ an.
- Gemeinsame Exkursion in ein nahegelegenes Museum: Viele Museen in Deutschland befassen sich mit dem Thema Kommunikation, etwa Fernmeldemuseen (Bremen, Dresden, Mühlhausen), Postmuseen (Regensburg, Recklinghausen, Rheinhausen) oder Museen für Kommunikation (Berlin, Frankfurt, Hamburg, Nürnberg).

## 5 Inhalte des Moduls

Kommunikation ist ein wesentlicher Bestandteil unseres Lebens und kann sich in verschiedenen Formen und über verschiedene Medien vollziehen. Das Spektrum reicht von der allgemeinen mündlichen Sprache über die Gebärdensprache und Blindenschrift (analoge Kommunikation) bis hin zum Morsecode und der binären Kommunikation innerhalb von informatischen Systemen (digitale Kommunikation).

Die Grundlage einer jeden erfolgreichen Kommunikation ist die Absprache über die übermittelten Zeichen, deren Bedeutung, aber auch die Art und Weise der Kommunikation (z. B. wie wird eine Kommunikation begonnen und wie beendet). Solche Absprachen nennt man auch Protokolle. Daher sind Protokolle nichts anderes als eine Ansammlung von Regeln. Solche Protokolle sind nicht nur in der Informatik von großer Bedeutung. Historische Beispiele wie der Morsecode oder Rauchzeichen zeigen, dass sie schon früh notwendig waren.

Um den Schülerinnen und Schülern die Bedeutung von Protokollen für eine gelingende Kommunikation anschaulich zu machen, entwickeln sie eigene Protokolle in verschiedenen Szenarien, etwa beim Bau eines Dosentelefons, bei der Übermittlung von Informationen mithilfe der Armsprache oder eines selbstgebauten Morseapparats.

Als Vertiefung und um den Schülerinnen und Schülern zu zeigen, welche Zeichen neben der Sprache übermittelt werden können, wird in diesem Modul auch die Kodierung von Informationen behandelt. Die Kodierung ist in der digitalen Kommunikation entscheidend, um Texte, Bilder oder Ton erfolgreich zu übertragen. Neben den Protokollen spielt hierbei das Binärsystem eine wichtige Rolle. Jeder Computer kennt nur zwei Zustände, um Informationen zu speichern oder zu übertragen: 0 (aus) und 1 (ein). Die Schülerinnen und Schüler kodieren und übermitteln in diesem Modul selbst eigene Bilder.

Hierbei beschränken wir uns jedoch nicht auf die Kodierung von Bildern, sondern befassen uns auch mit der Verschlüsselung. Einfache Verfahren, wie die Cäsar-Verschlüsselung stellen sogenannte Substitutionsverfahren dar. Diese sind Bestandteile des Verschlüsseln und nichts anderes als eine Form von Kodierungen.

## 6 Unterrichtliche Umsetzung

Dieses Modul zeichnet sich dadurch aus, dass es völlig ohne Computer auskommt und besonders einfach und aktivierend ist. Für die Schülerinnen und Schüler ist es spannend zu erfahren, wie man lange vor dem Internet auch über weite Strecken Nachrichten und Bilder übertragen hat.

Zu Beginn ermöglichen wir einen Bezug zur Realität mithilfe des Kinotrailers von „Schmetterling und Taucherglocke“. Im Mittelpunkt dieses Films steht der Chefredakteur der Modezeitschrift „Elle“, Jean-Dominique Bauby, der lediglich mit Blinzeln kommuniziert, da er vom Locked-In-Syndrom<sup>1</sup> betroffen ist. Dies soll Anlass geben, diese Kommunikationsform selbst auszuprobieren und dazu eigene Protokolle zu entwerfen, um das System zu verbessern und Fehler zu vermeiden. Für jüngere Schülerinnen und Schüler oder als Alternative kann die Geschichte auch von der Lehrkraft kurz erzählt werden, ohne den Trailer zu zeigen.

---

<sup>1</sup> Erklärung im Glossar

Außerdem können die Schülerinnen und Schüler einen eigenen Morse-Apparat oder ein Dosen-telefon bauen, sowie sich aktiv mit der Übertragung von Bildern befassen, indem sie eigene Pixelstrukturen entwerfen. Abschließend kann sich die Klasse mit ersten Möglichkeiten der Verschlüsselung beschäftigen und eine Cäsarscheibe bauen.

## 6.1 Grober Unterrichtsplan

### Variante 1

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Einführung in die Thematik mit „Schmetterling und Taucherglocke“, die Kinder/Jugendlichen entwerfen erste eigene Protokolle für die Kommunikation durch Blinzeln.
Vertiefung	Die Schülerinnen und Schüler befassen sich mit den Anfängen der Fernkommunikation, in Form von Gruppenarbeiten werden unterschiedliche Aufgaben gelöst: Bau eines Morse-Apparates, Bau eines Dosen-telefons, Entwicklung eines Protokolls für Armsprache, Bilddarstellung im Binärcode.

### Variante 2

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Einführung in die Thematik mit „Schmetterling und Taucherglocke“, die Kinder/Jugendlichen entwerfen erste eigene Protokolle für Blinzeln-Kommunikation.
Vertiefung	Die Schülerinnen und Schüler befassen sich mit den Anfängen der Fernkommunikation, sie entwickeln ein Protokoll für Armsprache und bauen einen Morse-Apparat.
Vertiefung	Die Schülerinnen und Schüler befassen sich mit der Bilddarstellung und Speicherung durch Zahlen.
Vertiefung	Verschlüsselungstechniken werden vorgestellt, die Schülerinnen und Schüler basteln eine Cäsar-Scheibe zum Verschlüsseln.

## 6.2 Stundenverlaufsskizzen

### 6.2.1 Variante 1

Variante 1 ist eine Kurzform, die auch für die Grundschule geeignet ist. Dafür werden etwa 3 Stunden benötigt. Die einzelnen Themen lassen sich auf mehrere Gruppen verteilen und in Form einer Gruppenaufgabe bearbeiten.

#### Abkürzungen/Legende

AB = Arbeitsblatt/Arbeitsblätter; L = Lehrkraft; MuM = Mitschüler\*innen; SuS = Schüler\*innen; UV = Unternehmensvertreter\*in

#### Einführung

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Begrüßung, Vortrag, Präsentation	Begrüßung und ggf. Vorstellung des UV, zum Einstieg in die Thematik kann der Kino-Trailer zu „Schmetterling und Taucherglocke“ angesehen werden, dazu sollte kurz erklärt werden, was das Locked-In-Syndrom ist.  (Link zum Trailer: <a href="http://www.filmstarts.de/kritiken/71654-Schmetterling-und-Taucherglocke.html">http://www.filmstarts.de/kritiken/71654-Schmetterling-und-Taucherglocke.html</a> )	Trailer „Schmetterling und Taucherglocke“, Beamer, Computer, Lautsprecher
10 Min.	Hinführung	Plenum, gemeinsames Lesen	B1.1 „Schau mir in die Augen“ wird ausgeteilt und gemeinsam gelesen; Klären von Fragen.	B1.1 Sek. I oder B1.1 GS
50 Min.	Erarbeitung	Tandemarbeit	SuS teilen sich in Gruppen auf und bearbeiten die Aufgaben von B1.1.	
10 Min.	Sicherung	Plenum	Präsentation der Lösungen	
10 Min.	Transfer	Plenum, Rundgespräch	Abschlussfrage: Was hat das mit Informatik/IT zu tun?	

#### Vertiefung

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Plenum, Rundgespräch	B1.2 wird ausgeteilt und gemeinsam gelesen: Fragen: Welche Formen der Fernkommunikation fallen euch ein? Welche Möglichkeiten der Fernkommunikation nutzt ihr selbst? Wie verlief es mit der Fernkommunikation vor dem Internet? Ideen werden gesammelt.	B1.2 Sek. I
60 Min.	Erarbeitung	Gruppenarbeit	Einführung in die Gruppenarbeit; SuS teilen sich in vier Gruppen auf, jede Gruppe erhält einen eigenen Arbeitsauftrag: <ul style="list-style-type: none"> <li>• die Armsprache</li> <li>• der Morse-Apparat</li> <li>• Malen nach Zahlen (Bildübertragung)</li> <li>• das Dosentelefon</li> </ul>	B1.3 Sek. I oder B1.3 GS, B1.4 Sek. I oder B1.4 GS, B1.5, B1.7 GS; Batterien, Draht, Nägel, Reißzwecken, Hammer, Dosen, Schnur, Lämpchen
20 Min.	Sicherung	Plenum	Die Ergebnisse der Gruppenarbeit werden präsentiert und besprochen.	

### 6.2.2 Variante 2

Die Variante 2 umfasst vier Einheiten, die in etwa 4 bis 6 Stunden umgesetzt werden können, je nachdem ob alle Aufgaben der Arbeitsblätter oder nur Teile bearbeitet werden. **Die Einführung ist identisch mit Variante 1 und wird daher nicht nochmals aufgeführt.**

### Vertiefung (Morse-Apparat)

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
5 Min.	Einstieg	Plenum	B1.2 wird ausgeteilt und die Einführung gemeinsam gelesen.	B1.2
20 Min.	Erarbeitung	Tandemarbeit	B1.3 wird ausgeteilt und die erste Aufgabe wird gelöst: In Tandemarbeit entwickeln SuS ein Protokoll für eine Armsprache und probieren es aus.	B1.3 Sek. I oder B1.3 GS
10 Min.	Sicherung	Plenum	Wie wurde die Aufgabe gelöst: SuS präsentieren Beispiele ihrer Kommunikation.	

30 Min.	Erarbeitung	Gruppenarbeit	B1.4 wird ausgeteilt und SuS bauen nun einen eigenen Morseapparat.	B1.4 Sek. I oder B1.4 GS, Batterien, Lämpchen, Draht, Wäscheklammern, Holzbretchen, Büroklammern
15 Min.	Erarbeitung	Gruppenarbeit	SuS lösen die dazugehörigen Aufgaben von B1.4, die restlichen Aufgaben ggf. als Hausaufgabe.	B1.4 Sek. I oder B1.4 GS
5 Min.	Sicherung	Plenum	SuS präsentieren ihre Morseapparate und Lösungswege.	

### Vertiefung (Bildspeicherung und Darstellung)

Zeit	Phase	Sozialform/Impuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Plenum, Lehrkraftvortrag	Wiederholung: Was hatten wir bisher? Blitzprotokoll, Fernkommunikation, Bau eines Morseapparates; wir haben gelernt, wie man Information in Zeichen umwandeln und übertragen kann. Eingangsfragen: <ul style="list-style-type: none"> <li>• Wie können Bilder gespeichert und übertragen werden, wenn der Computer nur Zahlen kennt?</li> <li>• Wann muss ein Computer Bilder speichern können? (z. B. Zeichenprogramm, Computerspiele, Multimedia-Anwendungen)</li> </ul>	
15 Min.	Hinführung	Lehrkraftvortrag	Erklärung der Pixelstruktur und wie mithilfe von Zahlen Bilder oder Buchstaben dargestellt werden, Austeilung von B1.5.	
20 Min.	Erarbeitung	Tandemarbeit	Malen nach Zahlen: Die vorgegebenen Raster müssten mit Hilfe der Zahlen richtig ausgemalt werden, S/W-Bilder und auch Farbbilder.	B1.5, Stifte in verschiedenen Farben
5 Min.	Sicherung	Plenum	Die Ergebnisse werden verglichen.	
25 Min.	Erarbeitung	Einzelarbeit	SuS erstellen ein eigenes Raster und entwerfen ein Bild für ihre Nachbarin oder ihren Nachbarn.	

10 Min.	Erarbeitung	Einzelarbeit	Die fertigen Bilder werden getauscht und bearbeitet, ggf. auch als Hausaufgabe möglich.
5 Min.	Sicherung	Tandemarbeit	Die ausgemalten Bilder werden gegenseitig kontrolliert.

### Vertiefung (Verschlüsselung)

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
5 Min.	Einstieg	Plenum, Lehrkraftvortrag	Eingangsfrage: Wie kann man Nachrichten geheim halten? Kennt jemand Geheimschriften?	
15 Min.	Hinführung	Lehrkraftvortrag	Erklärungen zu Verschlüsselung; Austeilen von B1.6; L gibt Hinweise zu Lösung der ersten Übungsaufgaben.	B1.6 Sek. I oder B1.6 GS
30 Min.	Erarbeitung	Einzel- oder Tandemarbeit	Bau einer Cäsar-Scheibe	Pappe, Schere, Kleber, Bleistift, Geodreieck, Zirkel, Musterklammern
15 Min.	Erarbeitung	Einzel- oder Tandemarbeit	SuS lösen die dazugehörigen Aufgaben.	
5 Min.	Sicherung	Plenum	Vergleichen der Ergebnisse	
15 Min	Erarbeitung	Einzel- oder Tandemarbeit	Wie kann der Code geknackt werden? SuS lösen die Aufgaben auf dem Arbeitsblatt.	
5 Min.	Sicherung	Plenum	Vergleichen der Ergebnisse	

## 7 Einbettung in verschiedene Fächer und Themen

Die Auseinandersetzung mit den Grundlagen der Kommunikation stellt Bezüge zum Fach Deutsch her. Die Einheit zur Fernkommunikation, die den Bau des Morse-Apparats einschließt, kann im Rahmen der Physik (Stromkreis) behandelt werden.

Das Locked-In-Syndrom lässt sich im Zusammenhang mit Signalverarbeitung der Nerven im Fach Biologie aufgreifen. Der Mathematikunterricht bietet die Möglichkeit, die Binärzahlen und das Thema Verschlüsselung einzubinden und im Fach Kunst kann das Arbeitsblatt „Malen nach Zahlen“ zur Ergänzung bei der Erklärung von Bildformaten dienen. Dadurch kann das gesamte Modul oder es können einzelne seiner Teile in ganz verschiedenen Fächern durchgenommen werden.

Darüber hinaus haben wir im Folgenden aufgeführt, welche Kompetenzen aus den Bildungsstandards der Kultusministerkonferenz oder der einzelnen Rahmenlehrplänen der Länder durch das Modul B1 von IT2School unterstützt werden:

### Deutsch

Die Schülerinnen und Schüler ...

- können über Verstehens- und Verständigungsprobleme sprechen (Grundschule).
- kennen die Grundfaktoren sprachlicher Kommunikation (Sek I).
- kennen „Sprache in der Sprache“ und können ihre Funktion unterscheiden: Standardsprache, Umgangssprache, Dialekt, ... (Sek I).
- erkennen und reflektieren Faktoren und Ursachen gelingender und misslingender Kommunikation (Sek I).
- erarbeiten Lösungsansätze zur Behebung von Kommunikationsstörungen (Sek I).

### Mathematik

Die Schülerinnen und Schüler ...

- können ebene Figuren in Gitternetzen abbilden (verkleinern und vergrößern) (Grundschule).
- können arithmetische und geometrische Muster selbst entwickeln, systematisch verändern und beschreiben (Grundschule).
- arbeiten bei der Lösung von Problemen im Team, entwickeln eigene und vorgegebene Lösungswege (Sek I).
- können Lösungswege beschreiben und begründen (Sek I).
- können selbst Probleme formulieren und Lösungsideen entwickeln (Sek I).

## Sachunterricht

Die Schülerinnen und Schüler ...

- können an einem Beispiel aus ihrer Alltagswelt technische Funktionsweisen beschreiben.
- können an einem Beispiel Weiterentwicklung, Veränderung und Folgen technischer Erfindungen erläutern.
- verstehen einfache mündlich und/oder visuell dargebotene Bauanleitungen und können sie umsetzen.
- können Informationen darstellen.

## Informatik

Schülerinnen und Schüler ...

- strukturieren Daten im Kontext einer gegebenen Problemstellung.
- entwickeln Modelle und stellen diese dar.
- setzen ihre Problemlösungen in ausführbare Prozesse um.
- überprüfen, ob ein vorliegendes Verfahren ein Problem löst.
- vergleichen unterschiedliche Lösungsansätze und nennen Vor- und Nachteile.
- begründen Zusammenhänge im Kontext der Informatik.
- unterscheiden zwischen Informationen und ihrer Repräsentation durch Daten.
- entwerfen Algorithmen und stellen diese geeignet dar.

## 8 Anschluss Themen

Das Modul B1 – Vom Blinzeln zum Verschlüsseln ist eine geeignete Basis für alle weiteren Module im Rahmen von IT2School. Um Ihnen die Entscheidung zu erleichtern haben wir hier konkrete Vorschläge für Sie zusammengestellt:

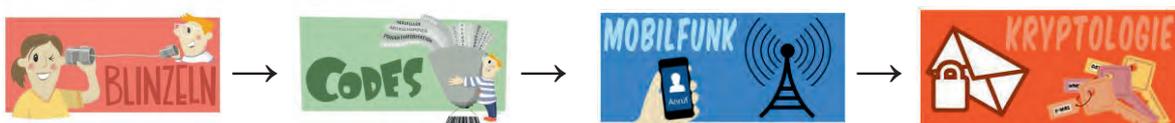
### Beispiel: Leichter Einstieg ohne Technik

Möchten Sie analog und niedrigschwellig weitermachen, dann empfehlen wir folgende Module:



### Beispiel: Daten, Information und Sicherheit

Möchten Sie nach dem Modul „Blinzeln“ die Thematik der Daten, Information und Sicherheit vertiefen, dann könnte der folgende Weg der richtige sein:



Die Schülerinnen und Schüler erfahren etwas über verschiedene Darstellungsformen von Informationen und Datensicherheit und befassen sich mit dem Thema Big Data.

### Beispiel: IT spielend entdecken

Der Umgang von Informationen und deren Weiterverarbeitung ist ein wichtiger Bestandteil vieler Programme. Entsprechend liegt es nahe auch eigene Programme zu entwickeln. Im Modul B5 erfolgt hierfür ein sehr einfacher Einstieg mit der grafischen Programmierumgebung Scratch. Mit der Erweiterung B6 Mein Anschluss, kann diese Thematik ausgebaut werden und die Schülerinnen und Schüler lernen zusätzlich kennen, wie Eingabegeräte für die Programmierung genutzt werden können.



## 9 Literatur und Links

- Gallenbacher, Jens (2012): **Abenteuer Informatik. IT zum Anfassen** – von Routenplanern bis Online-Banking. 3. Auflage. Spektrum Akademischer Verlag
- Bell, Tim et al. (2015): **CS unplugged – Computer Science without a Computer**. Sammlung von Materialien, die Schülerinnen und Schüler an die Informationstechnik heranführt, ohne dass fortgeschrittene Fähigkeiten wie Programmierkenntnisse erforderlich sind: <http://csunplugged.org>
- YouTube-Channel zu **CS unplugged**: <https://www.youtube.com/user/csunplugged>
- Tröger, Peter (2014): **Informatik ohne Stecker**. Beruht auf den Materialien von Tim Bell, deutsch. Online: <http://www.troeger.eu/unplugged>
- Trailer zum Film **“Schmetterling und Taucherglocke”**: Lieferte Idee zur Einheit; Einstiegsmöglichkeit für dieses Modul. Online: <http://www.filmstarts.de/kritiken/71654-Schmetterling-und-Taucherglocke.html>
- Hunkin, Tim & Garrod, Rex (1993): **Secret life of machines: Fax machine**. Video, das die Funktionsweise von Faxgeräten erklärt. Online: <https://www.youtube.com/watch?v=laCfs5Xb-EI>
- Hunkin, Tim & Garrod, Rex: **The Fax machine**. Comic, der die Entstehung des Faxgerätes erklärt. Online: [http://www.secretlifeofmachines.com/secret\\_life\\_of\\_the\\_fax\\_machine.shtml](http://www.secretlifeofmachines.com/secret_life_of_the_fax_machine.shtml)

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B1.1	Schau mir in die Augen	Arbeitsblatt dient zum Einstieg in das Modul. SuS befassen sich mit der Kommunikation beim Locked-In-Syndrom. Sie entwerfen ein eigenes Blinzeln-Protokoll zur Kommunikation und reflektieren diese.
 B1.2	Von den Anfängen der Fernkommunikation	Arbeitsblatt zu historischen Eckdaten und Arten der ersten Fernkommunikation.
 B1.3	Die Armsprache	Kennenlernen der Arm-Sprache und Entwicklung eines eigenen Protokolls zur Übermittlung von Zahlen.
 B1.4	Der Morse-Apparat	Arbeitsblätter zur Historie des Morse-Apparats. Beinhaltet Anregungen, einen eigenen Morseapparat zu bauen und sich Nachrichten zu übermitteln.
 B1.5	Malen nach Zahlen	Arbeitsblätter zur Übermittlung von Daten durch die Kodierung von Bildern.
 B1.6	Die Cäsar-Verschlüsselung	Arbeitsblätter mit Anleitung zur Cäsar-Verschlüsselung
 B1.7	Das Dosentelefon	Arbeitsblatt mit Anleitung zum Bau eines Dosentelefons sowie weitere Aufgaben wie z. B. der Entwicklung eines Protokolls.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Algorithmus	Als Algorithmus wird eine Handlungsvorschrift bezeichnet, deren einzelnen Handlungsanweisungen eindeutig und deterministisch (endlich; zeitlich begrenzt) sind. Algorithmen beschreiben meist, wie gegebene Problemstellungen gelöst werden oder bestimmte Tätigkeiten durchzuführen sind.
Locked-In-Syndrom	Zustand, in dem ein Mensch völlig gelähmt, aber trotzdem bei vollem Bewusstsein ist (deutsch: Gefangensein- bzw. Eingeschlossenein-Syndrom); Kommunikation meist nur durch Augenbewegungen möglich. Ursachen sind Verletzungen im Gehirn, z. B. nach Gefäßverschluss (Schlaganfall).

Pixel	Bildpunkt zur Darstellung von Farbe auf einem Monitor, Kunstwort aus <i>pictures</i> (kurz „pix“) und <i>element</i> (kurz „el“)
Protokoll	Vereinbarung über den Informationsaustausch zwischen zwei Systemen, Regeln und Formate werden festgelegt, um eine vollständige und fehlerfreie Kommunikation zu gewährleisten.

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Schau mir in die Augen!

Stellt euch vor: Ein Mann kann sich nicht mehr bewegen und auch nicht mehr sprechen. Er kann nur noch mit den Augen blinzeln.

Mit der Hilfe einer Freundin kann er aber trotzdem Dinge sagen. Die Freundin sagt dafür langsam das Alphabet auf und immer beim richtigen Buchstaben blinzelt der Mann.

Den richtigen Buchstaben schreibt die Freundin auf. Auf diese Weise kann der Mann Wörter diktieren.



## Aufgaben

1. Probiere mit Blinzeln deinem Nachbarn ein Wort zu diktieren. Beachte, dass das Wort nicht zu lang ist.
2. Ist das richtige Wort angekommen oder gab es Probleme? Hat dein Nachbar vielleicht etwas falsch verstanden?
  - a. Was kann man machen, wenn man sich mal vertan hat?
  - b. Was kann man machen, damit das Diktieren schneller geht?
3. Stellt Vermutungen auf und recherchiert, welche Funktion die Buchstabentafel im Bild hat.

# Schau mir in die Augen!

Der Chefredakteur der Modezeitschrift „Elle“, Jean-Dominique Bauby, war 43 Jahre alt als ein Schlaganfall sein Leben vollständig änderte. Von da an konnte er weder sprechen noch sich bewegen. Die einzige Möglichkeit zur Kommunikation, die ihm blieb, war mit dem linken Auge zu blinzeln.

Mit einem einfachen System, das ihm seine Logopädin vorschlug, gelang es ihm sogar, ein Buch zu schreiben<sup>1</sup>. Dabei sagt eine Hilfsperson langsam das Alphabet auf und beim richtigen Buchstaben blinzelt der Patient. Diesen Buchstaben notiert die Person und beginnt wieder im Alphabet von vorn.



## Aufgaben

1. Probiere mit diesem System deinem Nachbarn ein Wort zu diktieren.
2. Beobachtet und diskutiert die Schwierigkeiten dabei.
3. Stellt weitere Regeln auf, um das System zu verbessern, z. B.
  - a. wenn man sich mit einem Buchstaben vertan hat.
  - b. wenn man das System beschleunigen will.
4. Stellt Vermutungen auf und recherchiert, welche Funktion die Buchstabentafel im Bild hat.

## Hintergrund

Die Grundlage einer jeden erfolgreichen Kommunikation sind die Absprachen über die übermittelten Zeichen und ihre Bedeutung. Dazu gehört auch die Reihenfolge, in der sie auftreten. Solche Absprachen (Protokolle) sind nicht nur in der Informatik von großer Bedeutung, sondern wurden schon früh zur Kommunikation entwickelt. Neben den gesprochenen und geschriebenen Sprachen sind auch Gestik und Symbole aller Art Teil dieser Protokolle. Zur Kommunikation über größere Entfernungen wurde z. B. der Morse-Code entworfen. Auch Faxgeräte benötigen Protokolle, um die Texte und Bilder zu übertragen und natürlich die Telekommunikation allgemein, ob über das Festnetz oder mit Mobiltelefonen. Die Flexibilität des Internets gewährleistet die Protokollfamilie TCP/IP und bildet damit die Basis für viele andere Dienste, wie zum Beispiel E-Mail, SSH oder das WWW.

---

<sup>1</sup> In „Schmetterling und Taucherglocke“ (Originaltitel: Le scaphandre et le papillon) beschreibt er seine Situation sehr eindrucksvoll und gibt damit bisher einmalig Einblick in die Situation eines Patienten mit „Locked-in-Syndrom“. Das Buch wurde in viele Sprachen übersetzt und auch verfilmt. Dominique Bauby starb kurz bevor es veröffentlicht wurde.

# Von den Anfängen der Fernkommunikation

Eine der wichtigsten Voraussetzungen für die Entwicklung der Menschheit und ihrer kulturellen und technischen Errungenschaften ist die Fähigkeit der Menschen, untereinander zu kommunizieren und Informationen auszutauschen. Eine wichtige Rolle spielen dabei unsere Sinnesorgane: Sie gestatten es uns, unsere Umwelt wahrzunehmen und Signale an die Umgebung zu senden. Für den Informationsaustausch bedienen wir uns meist der **akustischen Kommunikation**.

Im Kindesalter haben wir von den Erwachsenen das Sprechen gelernt – die Sprache ist das Werkzeug für die Informationsweitergabe. Wir können die Sprache mit Hilfe unserer Stimmbänder benutzen, in dem wir die dazu passenden Laute formen. Zugleich können wir mithilfe der gelernten Sprache und den dazu erlernten Schriftzeichen Informationen austauschen.

Kommunikation findet aber auch ohne die Benutzung von Sprache – also nonverbal – statt: Unsere Gestik und Mimik können von unseren Mitmenschen interpretiert werden und damit auch als Informationsquelle dienen. Gehörlose Menschen benutzen die Gebärdensprache für die Kommunikation. In diesen Fällen spricht man von **optischer Kommunikation**.

Für die Kommunikation über größere Entfernungen sind wir jedoch auf **Kommunikationsträger** angewiesen: Dies waren in früheren Zeiten menschliche Boten, die eine Information wie eine Ware transportierten – davon übrig geblieben ist die Briefpost, die es trotz der heute weit verbreiteten elektronischen Kommunikation immer noch gibt.

Leider benötigt diese Kommunikationsform immer viel Zeit; deshalb versuchten die Menschen schon in sehr frühen Jahren andere Wege zu gehen. Von den amerikanischen Ureinwohnern wissen wir, dass sie sich über Rauchzeichen verständigten, die Chinesen haben Drachen mit unterschiedlichen Farben und Fähnchen zum Himmel steigen lassen, in einigen Regionen Afrikas gab es die Buschtrommel und die Ägypter hatten eine Fackelsprache, aus denen die Seefahrer später die Flaggensignale ableiteten. In Europa war die Kommunikation mittels Brieftauben verbreitet; aber auch optische Signale mit Laternen und Zeigermasten, die noch bis in die Mitte des vergangenen Jahrhunderts in der Seefahrt eingesetzt wurden. Allen Übertragungen war aber gemeinsam: Es durften keine größeren Hindernisse im Kommunikationsweg sein.



F. Remington - The Smoke Signal

## Aufgaben

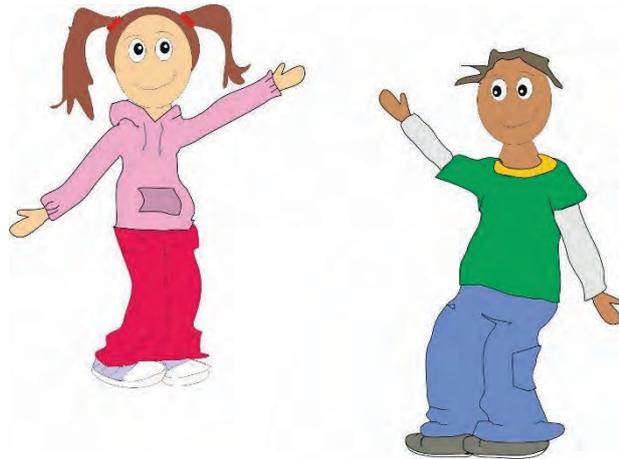
1. Welche Übertragungsmöglichkeiten fallen dir noch ein?
2. Erstelle eine Zeitleiste, in der die verschiedenen Kommunikationsarten angeordnet sind. Nutze das Internet für deine Recherchen.

Abbildung: F. Remington – The Smoke Signal. Quelle: (Public Domain)  
[https://commons.wikimedia.org/wiki/File:Frederic\\_Remington\\_smoke\\_signal.jpg](https://commons.wikimedia.org/wiki/File:Frederic_Remington_smoke_signal.jpg) [17.11.2015]

# Die Armsprache

Habt ihr schon mal versucht, nur mit den Armen zu reden? Man kann mit seinen Armen Zahlen übertragen, je nachdem in welcher Stellung die Arme sind. Die Finger dürfen dabei nicht gezeigt werden.

Zum Beispiel, wenn der linke Arm ausgestreckt nach oben zeigt, könnte das 1 bedeuten.



## Aufgaben

Vereinbare mit deinem Partner Regeln, wie die Armstellung für die Zahlen 0 bis 9 aussehen soll.

Habt ihr die Regeln vereinbart? Dann versucht folgende Daten zu übertragen. Dabei dürft ihr aber nicht sprechen!

1. Übertragt euch gegenseitig mit dieser Zeichensprache euer Geburtsdatum.
2. Jetzt übertragt ihr das Geburtsdatum von einem Mitglied eurer Familie. Wie lange braucht ihr dazu?
3. Es soll mit dieser Sprache auch der Name des Familienmitglieds übertragen werden – was müsst ihr zusätzlich noch vereinbaren? Kann man noch mehr Zeichen nur mit den Armen darstellen?

# Die Armsprache

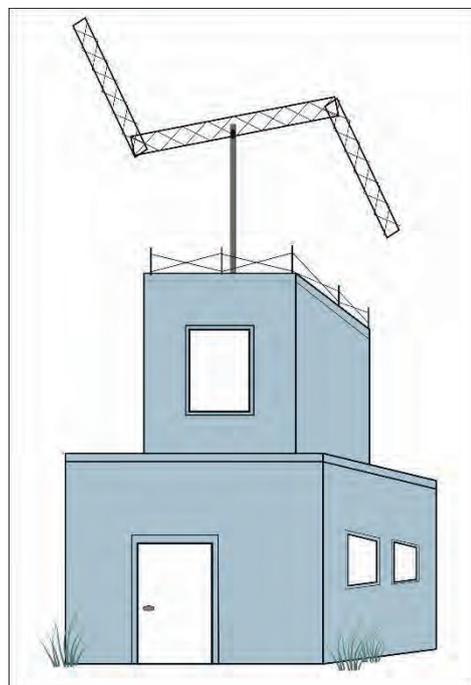
Eine Form der optischen Sprachübermittlung ist die Armsprache. Je nachdem, wie die Stellung der Arme ist, lassen sich beispielsweise Zahlen oder Buchstaben übertragen.

## Aufgaben

Vereinbare mit deinem Partner nur durch Einsatz der abgewinkelten Arme eine Kommunikation, die es erlaubt, die Ziffern von 0 bis 9 zu übertragen. Die Finger sollen dabei nicht verwendet werden. Wichtig: Während der nachfolgenden gesamten Datenübertragung darf nicht gesprochen werden!

1. Übertragt euch gegenseitig mit dieser Zeichensprache das Geburtsdatum eurer Mutter. Wie lange braucht ihr dazu?
2. Es soll mit dieser Sprache auch der Vorname der Mutter übertragen werden – was müsst ihr zusätzlich vereinbaren?
3. Schreibt, ohne dass euer Partner es sieht, einen willkürlichen Satz auf, der etwa 100 Buchstaben umfasst. Stoppt die Zeit, die ihr braucht, um diesen Satz zu übertragen. Ihr könnt jetzt eure Übertragungsgeschwindigkeit ermitteln. Teilt dazu die Übertragungszeit durch die Anzahl der Buchstaben.
4. Wie viele Übertragungsfehler sind bei eurer Übertragung entstanden?

Der französische Geistliche Claude Chappe (1763-1805) entwickelte 1791 mit seinen Brüdern unter Zuhilfenahme von zwei synchron laufenden Uhren den ersten optischen „Fernschreiber“ (lateinisch Telegraph). Dazu hinterlegten sie das Ziffernblatt der Uhren mit Codezeichensegmenten. Wenn der Sekundenzeiger über das zu sendende Codesegment strich, wurde von der Sendestation ein Gong betätigt, und auf der Gegenseite konnte nun das Zeichen abgelesen werden. Immerhin gelang es damit, 400 m zu überbrücken. Später ersetzten sie den Gong und die Uhren durch Signalmasten mit beweglichen Armen und konnten so bei Tageslicht bis zu 15 km überbrücken. Durch 22 günstig gelegene Zwischenstationen gelang es schon drei Jahre später, die 225 km lange Strecke von Paris nach Lille in Betrieb zu nehmen. In den nachfolgenden Jahren gelang es, ganz Frankreich mit einem System von Signalmasten zu versehen. Auch in England und den USA wurden ähnliche Systeme entwickelt und zur Kommunikation eingesetzt.



Optische Telegrafstation

# Der Morseapparat

## Das Morse- Alphabet

Der Morseapparat wurde vor über hundert Jahren zur Fernkommunikation verwendet. Entwickelt wurde er vom Amerikaner Samuel Morse. Nach ihm wurde auch das **Morsealphabet** benannt, das im Jahr 1865 in Paris genau festgelegt wurde. Samuel Morse hat jedem Buchstaben eine Abfolge von Punkten und Strichen zugeordnet. Dies nennt man Code. Dabei bedeuten Punkte kurze Signale und Striche lange Signale.

Buchstaben, die besonders häufig benutzt werden, bekamen einen kurzen Code zugewiesen, z. B. das E. Buchstaben, die selten benutzt werden, wie das Q oder das Y, bekamen einen langen Code. Rechts kannst du diese Zuweisung sehen. Kannst du dir vorstellen warum?

Nachrichten kann man mit Hilfe eines Morseapparates oder aber auch einfach mit Licht oder akustisch mit Klopfzeichen übertragen.

## Aufgaben

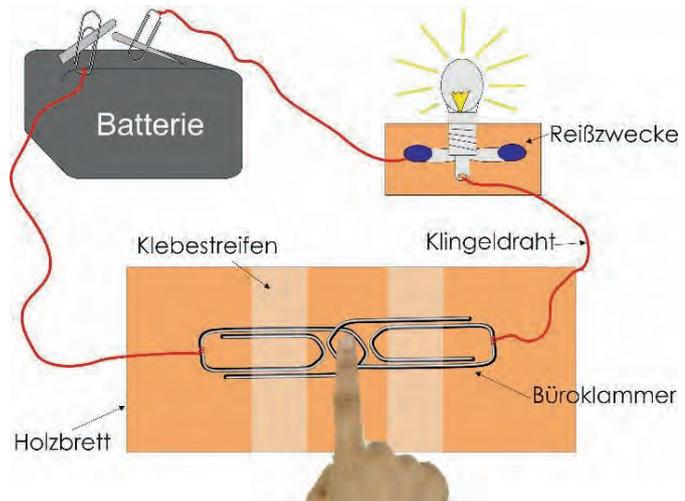
1. Schreibe ein Wort in Morsecode.
2. Tausche das Wort mit jemandem in der Klasse. Versuche das neue Wort herauszufinden.
3. Nun versuche ein Wort per Klang zu übertragen. Für Punkt (auch „kurz“ genannt) sagst du „Dit“, für Strich „Daa“ (auch "lang" genannt). Nach jedem Buchstaben musst du eine kurze Pause von etwa einer Sekunde machen, damit der andere Zeit zum Aufschreiben hat!
4. Baut nun einen Morseapparat und sendet euch gegenseitig eine Nachricht.

a	.-
b	-...
c	-.-
d	..
e	.
f	..-
g	--
h	....
i	..
j	.-.-
k	-.-
l	.-..
m	--
n	-. .
o	---
p	.-.-
q	--.-
r	.-.
s	...
t	-
u	..-
v	...-
w	.-.-
x	-.-. .
y	-.--
z	--..
1	.-.-.-
2	..-.-
3	...--
4	....-
5	.....
6	-....
7	--...
8	---..
9	----.
0	-----
SOS	...---...

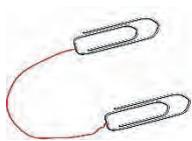
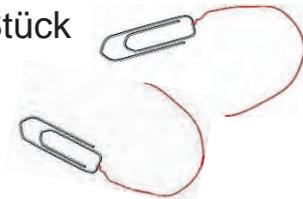
# Bauanleitung für einen Morseapparat

Du kannst dir leicht einen Morseapparat bauen, mit dem du jemandem eine Nachricht schicken kannst. Du benötigst dafür:

- 1 großes Holzbrettchen oder 2 kleinere Holzbrettchen (am besten Balsa-Holz)
- 1 Lämpchen 4,5 V
- 1 Flachbatterie 4,5 V
- Klingeldraht
- 2 Reißzwecken
- 4 Büroklammern
- Klebestreifen

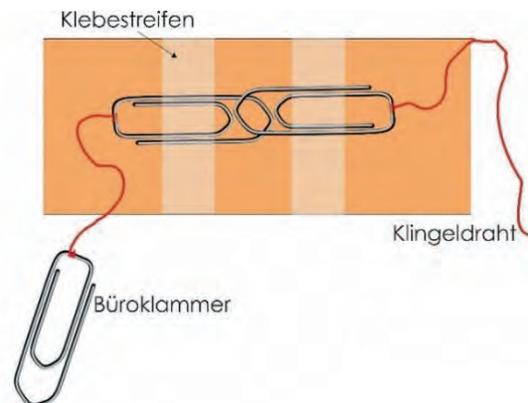


Zuerst verbindest du eine Büroklammer mit einem Stück Klingeldraht. Du kannst dir auf dem Foto abgucken, wie genau es aussehen soll. Davon benötigen wir zwei, deshalb machst du das gleich noch einmal.



Jetzt machst du das noch ein drittes Mal und befestigst diesmal an beiden Enden des Klingeldrahts je eine Büroklammer.

Dann nimmst du ein Stück Holz und den Klingeldraht mit den beiden Büroklammern. Eine Büroklammer legst du auf das Holzbrett und klebst sie mit einem Stück Klebestreifen fest. Nun nimmst du den Klingeldraht, an dem nur eine Büroklammer befestigt ist.



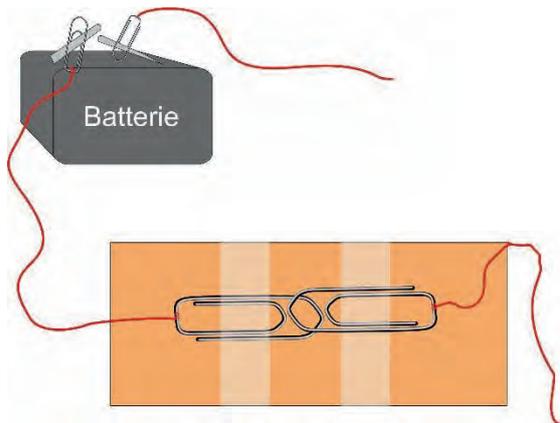
Diese Büroklammer biegst du etwas auseinander. Etwa so:



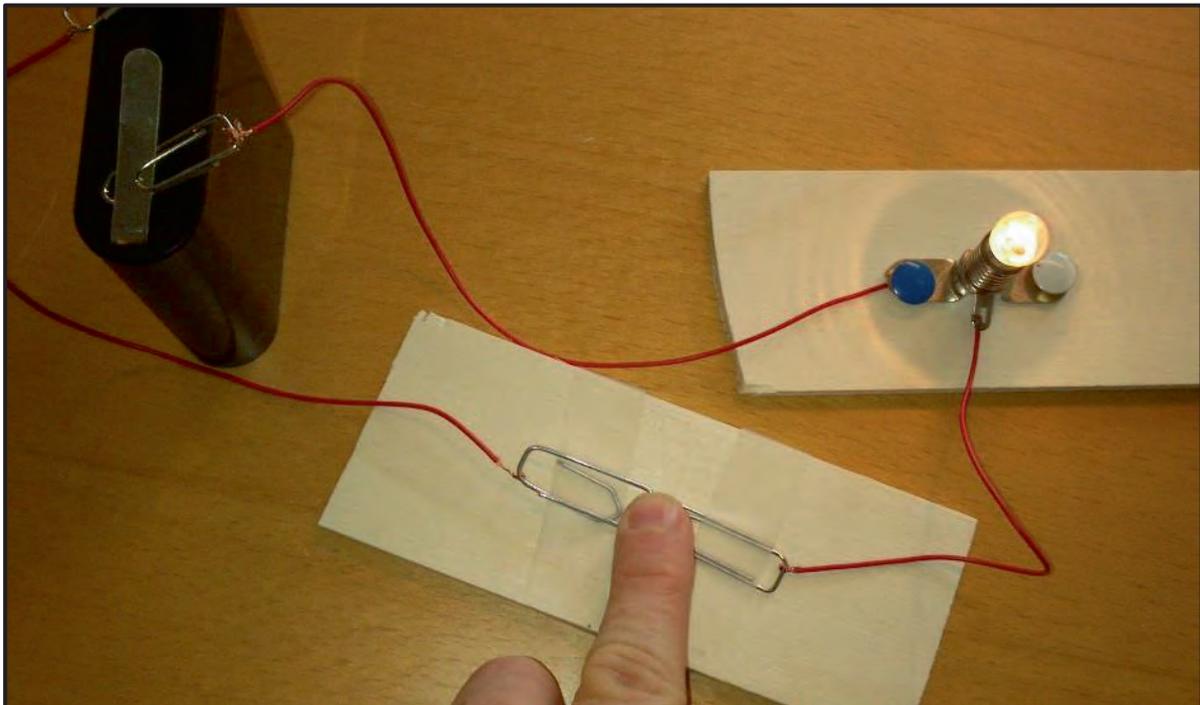
Jetzt befestigst du diese Büroklammer auch auf dem Holzbrettchen. Der Klebestreifen geht aber nur über die kurze Biegung der Büroklammer.

Die beiden Büroklammern, die nun noch frei liegen, befestigst du an der Batterie.

Die beiden Enden des losen Klingeldrahts befestigst du an der kleinen Glühbirne. Mit Hilfe der Reißzwecken kannst du die Glühbirne auf einem Stück Holz befestigen.



Nun ist dein Morseapparat schon fertig. Wenn du die gebogene Büroklammer herunterdrückst, sollte dein Lämpchen leuchten. Aber wie funktioniert das?



# Der Morseapparat

Mit dem Aufkommen des Elektromagnetismus konnten zum ersten Mal auch elektrische Schreiber eingesetzt werden. Der Amerikaner Samuel Morse entwickelte dazu nach mehrjähriger Tüftelarbeit 1838 das nach ihm benannte Morsealphabet, das in veränderter Form 1865 in Paris standardisiert wurde und noch heute seine Gültigkeit besitzt.

Samuel Morse hat jedem Buchstaben einen Code in Form von Punkten und Strichen zugeordnet. Punkte bedeuten kurze Signale und Striche lange Signale.

Buchstaben, die besonders häufig benutzt werden, bekamen einen kurzen Code zugewiesen, z. B. das E. Buchstaben, die selten benutzt werden, wie beispielsweise das Q oder das Y, bekamen einen langen Code.

Nachrichten kann man mit Hilfe eines Morseapparates oder aber auch einfach mit Licht oder akustisch mit Klopfzeichen übertragen.

## Aufgaben

1. Schreibe jemandem eine Nachricht mit Hilfe des Morsealphabets. Trenne dazu die einzelnen Morse-Buchstaben durch senkrechte Striche. Zwei senkrechte Striche signalisieren Satzanfang bzw. Satzende.
2. Tausche den Morse-Text aus und versuche den neuen Text zu entschlüsseln.
3. Versuche nun ein Wort akustisch zu übermitteln. Für Punkt (auch „kurz“ genannt) sagst du „Dit“, für Strich „Daa“ (auch „lang“ genannt). Nach jedem Buchstaben musst du eine kurze Pause von etwa einer Sekunde machen, damit Zeit zum Aufschreiben bleibt!
4. Baut nun einen Morseapparat und sendet euch anschließend gegenseitig eine Nachricht.

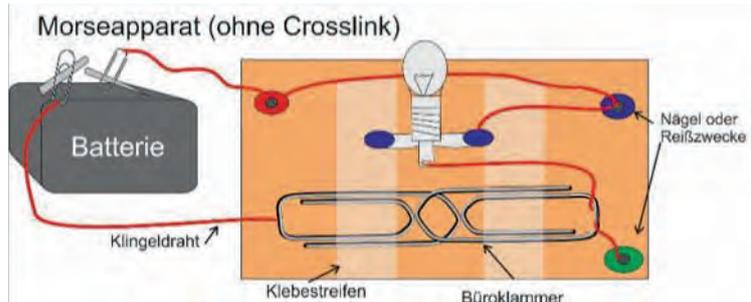
## Das Morse- Alphabet

a	.-
b	...-
c	-. -.
d	-. .
e	.
f	...-
g	-. -.
h	....
i	..
j	...-
k	-. -.
l	.. -.
m	--
n	-. -.
o	---
p	...-
q	-. -.
r	-. .
s	...
t	-
u	..-
v	...-
w	..-
x	-. -.
y	-. -.
z	-. -.
1	....-
2	...--
3	...--
4	....-
5	....
6	-. ...
7	-. ...
8	-. ...
9	-. ...
0	----
SOS	...---...

# Bauanleitung für einen Morseapparat

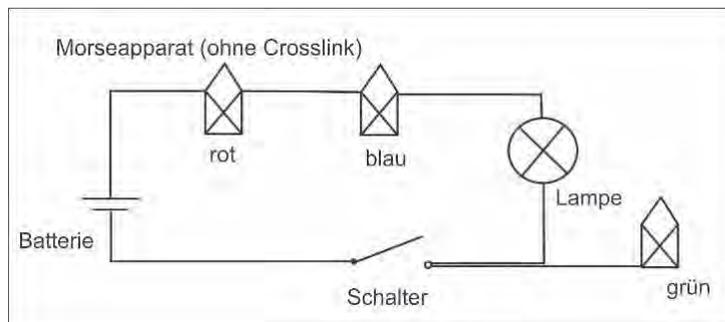
Du kannst dir leicht einen Morseapparat bauen, mit dem du deinem Partner eine Nachricht schicken kannst. Du brauchst dazu:

- 1 großes Holzbrettchen oder 2 kleinere Holzbrettchen (am besten Balsa-Holz)
- 1 Lämpchen 4,5 V
- 1 Flachbatterie 4,5 V
- Klingeldraht
- diverse Nägel
- 2 Büroklammern
- Klebestreifen (Tesafilm)



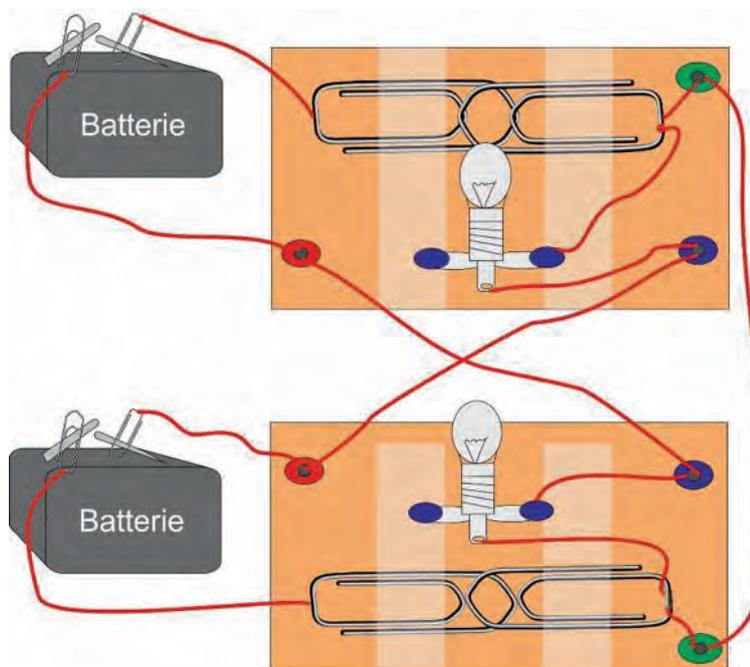
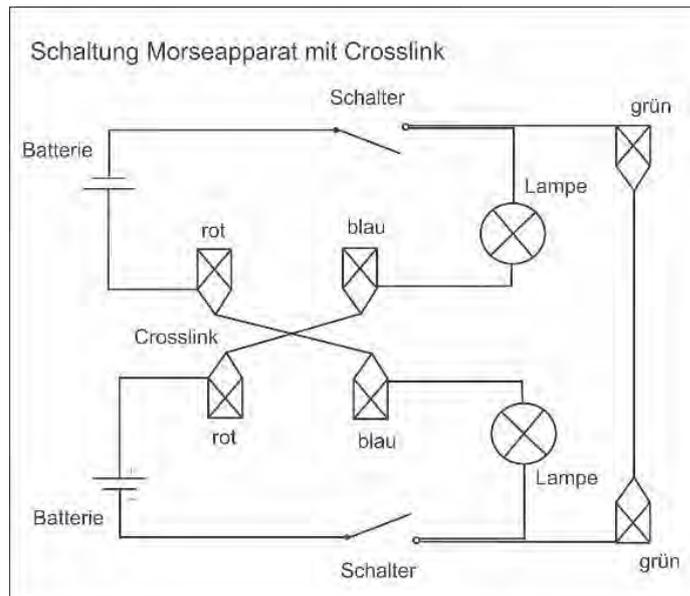
Die Büroklammern sowie die Nägel werden mit dem Klingeldraht verbunden. Für die Verdrahtung kannst du dich an dem oberen Schaubild und dem folgenden Schaltbild orientieren.

Der rote Nagel ist über eine verkabelte Büroklammer mit dem Pluspol der Batterie verbunden. Der Minuspol der Batterie ist über eine verkabelte Büroklammer mit dem „Schalter“ verbunden. Dieser besteht aus zwei festgeklebten Büroklammern, wobei eine etwas auseinandergebogen ist, sodass man sie runterdrücken kann. Die zweite Büroklammer des Schalters ist mit dem grünen Nagel und der Fassung des Lämpchens verbunden. Von der Fassung des Lämpchens geht dann auch nochmal eine Verbindung zum blauen Nagel.



Für einen Selbsttest deines Morseapparates verbindest du den roten und den blauen Nagel (*loopback*). Die Lampe muss angehen, wenn du die Taste drückst.

Für die Kommunikation brauchst du natürlich zwei Morseapparate. Sie werden über Kreuz (*crosslink*) miteinander verbunden, d. h. die grünen Nägel beider Stationen (auch *ground* genannt, Kurzbezeichnung GND) werden direkt miteinander verbunden, während der rote Nagel (Kurzbezeichnung TxD, *transmitted data*) mit dem blauen von der Gegenstation verbunden ist. Umgekehrt wird der rote Nagel der Gegenstation mit deinem blauen Nagel (Kurzbezeichnung RxD, *received data*) verbunden. Und nun viel Spaß!



## Aufgaben

5. Können beide Stationen gleichzeitig senden, wenn auf jeder Seite 2 Personen an der Station sind (Ein Hörer/ein Geber)?
6. Du hast zwischen deinem Partner, den du aber nicht sehen und hören kannst, eine Morse-Verbindung installiert. Was musst du außer dem Morsealphabet verabreden, damit du jederzeit Informationen austauschen kannst? Was wäre besonders hilfreich?
7. Die Digitaltechnik verwendet Nullen und Einsen. Eine Übertragung auf das Morsealphabet könnte so aussehen: Null entspricht „kurz“, Eins entspricht „lang“. Was meinst du dazu?

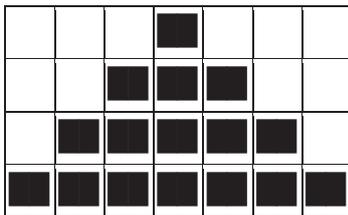
# Malen nach Zahlen

Der Computer, das Tablet und auch das Handy nutzen Zahlen, um Bilder zu speichern und zu zeigen und auch, um Bilder zu versenden. Für den Computermonitor ist alles ein Bild, auch Text, den man schreibt, oder Tabellen. Aber wie kann der Computer Bilder in Zahlen speichern? Computermonitore sind in ein feines Raster gegliedert, deren Punkte man Pixel („picture elements“, deutsch: Bildpunkte) nennt.

Wenn man auf Bildern die einzelnen Bildpunkte erkennen kann, nennt man sie auch manchmal „verpixelt“. Je mehr Bildpunkte ein Bild hat, desto besser ist die Qualität eines Bildes.

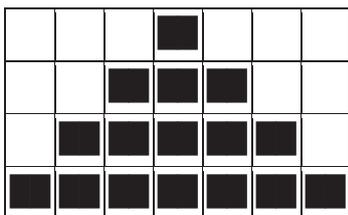


Ein Computermonitor hat meistens über eine Millionen solcher Pixel. Wir beginnen mit Schwarz-Weiß-Bildern. Bei einem schwarz-weißen Bild ist jedes Pixel entweder schwarz oder weiß.



Pixelraster 1

In unserem Beispiel wurde ein Dreieck ▲ vergrößert, um die Pixelstruktur deutlich zu machen. Wenn ein Computer ein Schwarz-Weiß-Bild speichert, muss er nur wissen, welche Punkte schwarz und welche weiß sind.



Pixelraster 2

3, 1, 3  
2, 3, 2  
1, 5, 1  
0, 7

Das ▲ ist in 28 (7x4) kleine Kästchen unterteilt, das sind die Pixel. Neben den Reihen stehen Zahlen, die sagen, ob ein Kästchen schwarz oder weiß ist. Die erste Zahl gibt immer die Anzahl weißer Pixel an. Beginnt die Zeile mit einem schwarzen Pixel, muss die Zeile mit einer Null beginnen. Die erste Zeile besteht aus drei weißen, einem schwarzen und wieder drei weißen Pixel. Die erste Zeile wird folglich als 3, 1, 3 gespeichert.

## Aufgaben

Nun versucht es doch selbst einmal. Malt die richtigen Kästchen in den Rastern mit Bleistift aus.

													13
													1, 2, 10
													0, 3, 10
													0, 3, 9, 1
													2, 1, 8, 1, 1
													2, 10, 1
													2, 9, 2
													2, 9, 2
													2, 2, 4, 2, 3
													2, 2, 4, 2, 3
													1, 3, 3, 3, 3

													2, 1, 2, 1, 6
													2, 4, 6
													2, 4, 6
													2, 4, 6
													3, 2, 3, 2, 2
													2, 4, 3, 1, 2
													2, 4, 3, 1, 2
													2, 4, 3, 2, 1
													1, 6, 3, 1, 1
													1, 6, 3, 1, 1
													1, 6, 3, 1, 1
													1, 6, 3, 1, 1
													1, 6, 1, 3, 1
													2, 7, 3

1. Jetzt werdet ihr zum Künstler. Entwerft ein eigenes Pixel-Bild. Nehmt dafür am besten kariertes Papier und einen Bleistift. Malt euch zuerst ein eigenes Raster auf, vielleicht 12 x 12 Kästchen.
2. Erstellt nun für euren Nachbarn oder eure Nachbarin ein Pixel-Bild. Zeichnet euch das Pixelbild zuerst vor, dann zeichnet ihr ein neues Raster mit den entsprechenden Zahlen dazu und übergebt es an eure Nachbarin oder euren Nachbarn.

# Farbige Bilder

Nun gibt es nicht nur Schwarz-Weiß-Bilder, sondern auch farbige Bilder. Auch die werden genau wie Schwarz-Weiß-Bilder in Pixel gespeichert. Um farbige Bilder darzustellen, wird eine zweite Zahl als Code für die verwendete Farbe genutzt.

**Schwarz = 0**      **Weiß = 1**      **Blau = 2**      **Gelb = 3**

Jedes Pixel wird mit zwei Zahlen gespeichert. Die erste Zahl sagt, wie viele gleiche Pixel in einer Reihe sind. Die zweite Zahl sagt, welche Farbe an der Reihe ist.

## Aufgaben

1. Vervollständigt die fehlenden Reihen:

											1, 0, 9, 1
											3, 1, 4, 3, 3, 1
											2, 1, 6, 3, 2, 1
											1, 1, 2, 3, 1, 2, 2, 3, 1, 2, 2, 3, 1, 1
											1, 1, 8, 3, 1, 1
											_____
											_____
											_____
											_____

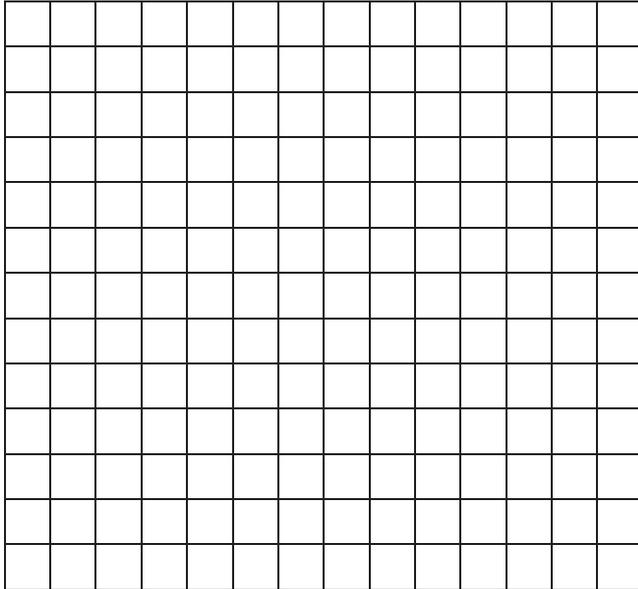
2. Gestaltet ein eigenes farbiges Bild für einen Freund oder eine Freundin. Schreibt zu Beginn auf, welche Zahl für welche Farbe steht:

**Farben:** \_\_\_\_\_

_____
_____
_____
_____
_____
_____
_____
_____
_____

Farben: \_\_\_\_\_



---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Schwarz-Weiß-Bilder und auch Farbbilder bestehen am Computer also immer aus sich wiederholenden Bildpunkten. Je mehr Bildpunkte ein Bild hat, desto hochauflösender ist es und desto besser ist seine Qualität.

# Bilder versenden

Ein Computer kennt nur zwei Zustände, um eine Nachricht oder ein Bild darzustellen:

- Strom an oder aus,
- Licht an oder aus (wie beim Lichtmorsen)
- Ja oder nein
- 0 oder 1

Da es für diesen Code nur zwei Zustände oder Zeichen gibt, nennt man ihn Binärcode.

Bei einem Schwarz-Weiß-Bild wird jeder weiße Bildpunkt mit einer 0 dargestellt und jeder schwarze Bildpunkt mit einer 1.

0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	1	1	0	0	0	0	1	1	0	0	0	0
0	0	1	1	0	0	0	0	1	1	0	0	0	0
0	1	1	1	0	0	0	1	1	1	0	0	0	0

## Aufgabe

Teilt euch in Kleingruppen auf und überlegt euch eine Figur. Diese Figur zeichnet ihr in das Raster „Nachricht zum Senden“. Danach schreibt ihr in alle weißen Kästchen eine „0“ und in alle schwarzen Kästchen eine „1“.

Nun versendet ihr dieses Bild an die nächste Gruppe. Ihr dürft immer nur „1“ oder „0“ sagen, „an“ oder „aus“, „ja“ oder „nein“. Ist das Bild übertragen, darf die nächste Gruppe senden.

### Nachricht zum Senden:


### Empfangene Nachricht:


# Die Cäsar-Verschlüsselung

Julius Cäsar war ein bekannter römischer Feldherr und Politiker. Er hat seine Briefe an Freunde oder Verbündete oft verschlüsselt, damit kein anderer die Nachrichten lesen konnte.

In seiner Verschlüsselung hat er jeden Buchstaben seiner Nachricht durch einen Buchstaben ersetzt, der drei Stellen später im Alphabet kommt. Aus dem Buchstaben A wurde D und aus dem Buchstaben B wurde E und so weiter.

Der Schlüssel sah so aus: Oben ist das Alphabet, man nennt es Klaralphabet. In der unteren Reihe ist das Geheimalphabet.



Nicolas Coustou - Julius Caesar

Klaralphabet																									
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
Geheimalphabet																									

Will man nun eine Nachricht verschlüsseln, ersetzt man einfach den Buchstaben des Klaralphabets durch den Buchstaben des Geheimalphabets. Probiere es einmal aus:

1. Schreibe deinen Namen auf und verschlüssele ihn mit der Cäsar-Verschlüsselung.
2. Kannst du die folgende Nachricht auch entschlüsseln?

XP GUHL LP NLQR

In diesem Beispiel wurden die Buchstaben um drei Stellen verschoben. Man kann natürlich auch fünf oder acht Stellen wählen. Damit man nicht jedes Mal eine neue Tabelle anlegen muss, kann man eine so genannte Chiffrier-Maschine bauen.

Abbildung: N.Coustou - Julius Caesar. Quelle: (Public Domain)  
[https://commons.wikimedia.org/wiki/File:Julius\\_Caesar\\_Coustou\\_Louvre.png](https://commons.wikimedia.org/wiki/File:Julius_Caesar_Coustou_Louvre.png) [17.11.2015]

## Bauanleitung

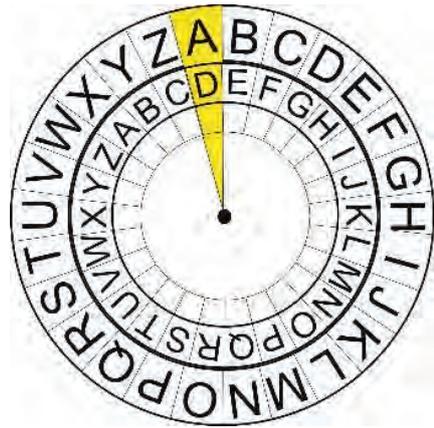
Um eine Verschlüsselungsscheibe zu basteln, benötigst du:

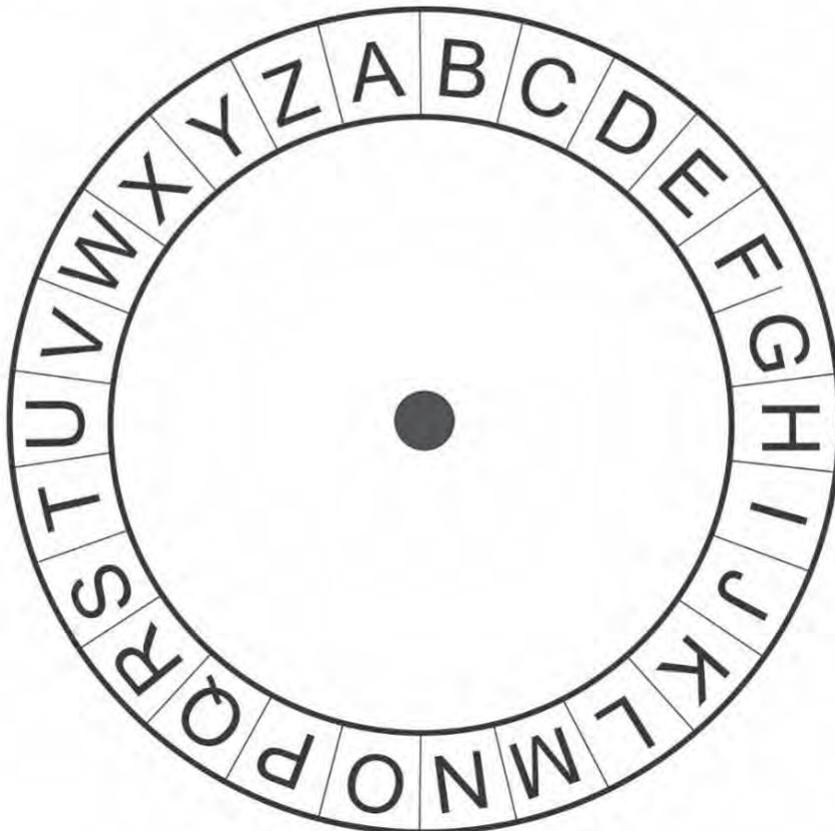
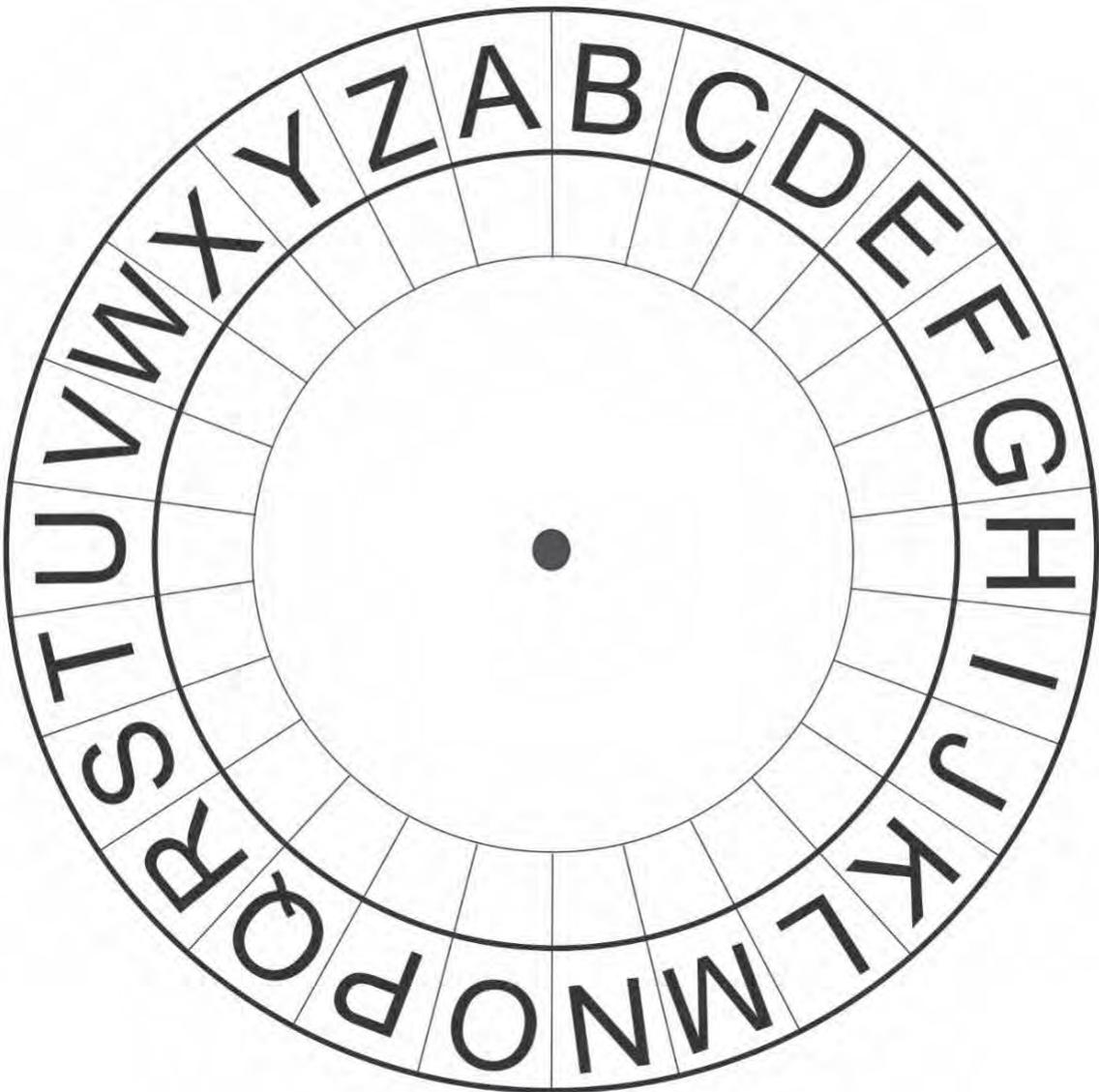
- Dünne Pappe
- Schere
- Kleber
- Musterklammer (benutzt man eigentlich zum Verschließen von Versandtaschen)

Klebe das Blatt mit den Kreisen auf eine dünne Pappe und schneide dann beide Kreise aus.

Lege danach die kleinere Scheibe auf die Größere und verbinde sie mit der Musterklammer. Mithilfe der Scheibe kannst du ganz einfach Texte verschlüsseln.

Wenn du jemandem deine verschlüsselte Nachricht sendest, denke daran, dass du auch den Schlüssel weitergibst, andernfalls kann der Empfänger die Nachricht nicht lesen. Der Empfänger muss wissen, um wie viele Stellen die Buchstaben auf der Scheibe verschoben wurden.





## Aufgaben

1. Verschlüssele folgenden Satz mit einer Verschiebung um vier Stellen:

„Wir treffen uns um vier Uhr vor der Schule“

W	I	R		T	R	E	F	F	E	N		U	N	S		U	M

V	I	E	R		V	O	R		D	E	R		S	C	H	U	L	E

2. Entschlüssele den folgenden Satz mit einer Verschiebung um neun Stellen:

VXAPNW PRKC NB BYJPQNCCR IDV VRCCJPNBBNW

V	X	A	P	N	W		P	R	K	C	

N	B		B	Y	J	P	N	C	C	Q	R

I	D	V		V	R	C	C	J	P	N	B	B	N	W

3. Überlege dir nun eine eigene Botschaft, die du für deinen Nachbarn oder deine Nachbarin verschlüsselst. Denke aber daran, dass du die Verschiebung deinem Nachbarn oder deiner Nachbarin auch übermitteln musst.

# Die Cäsar-Verschlüsselung

Der Versuch Schriften geheim zu halten ist wahrscheinlich so alt wie das Schreiben selbst. Viele Anlässe bewegten Menschen dazu besondere Geheimsprachen zu entwickeln – etwa Krieg, diplomatische Gründe oder die Liebe.

Die Wissenschaft, die sich mit dieser Thematik beschäftigt, nennt man **Kryptologie**.

**Krypto** = geheim      **logos** = Wort/Rede, Sinn  
(-logie bezeichnet die Wissenschaft eines Faches)

Auch der Feldherr und Politiker Julius Cäsar (100 bis 44 v. Chr.) hat sich seinerzeit viel mit der Verschlüsselung von Nachrichten befasst. In seiner Verschlüsselung hat er jeden Buchstaben einer Nachricht durch den Buchstaben ersetzt, der drei Stellen später im Alphabet kommt. Aus dem Buchstaben A wurde D und aus dem Buchstaben B wurde E und so weiter.



Abbildung: Nicolas Coustou - Julius Caesar

Der Schlüssel sah folgendermaßen aus. In der oberen Reihe ist das Alphabet – man nennt es Klaralphabet. In der unteren Reihe ist das Geheimalphabet.

Klaralphabet																									
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
Geheimalphabet																									

Will man nun eine Nachricht verschlüsseln, ersetzt man einfach den Buchstaben des Klaralphabets durch den Buchstaben des Geheimalphabets. Probiere es einmal aus:

- Schreibe deinen Namen auf und verschlüssele ihn mit der Cäsar-Verschlüsselung.
- Kannst du die folgende Nachricht auch entschlüsseln?

XP GUHL LP NLQR

In diesem Beispiel wurden die Buchstaben um drei Stellen verschoben. Man kann natürlich auch fünf oder acht Stellen wählen. Damit man nicht jedes Mal eine neue Tabelle anlegen muss, kann man eine so genannte Chiffrier-Maschine bauen.

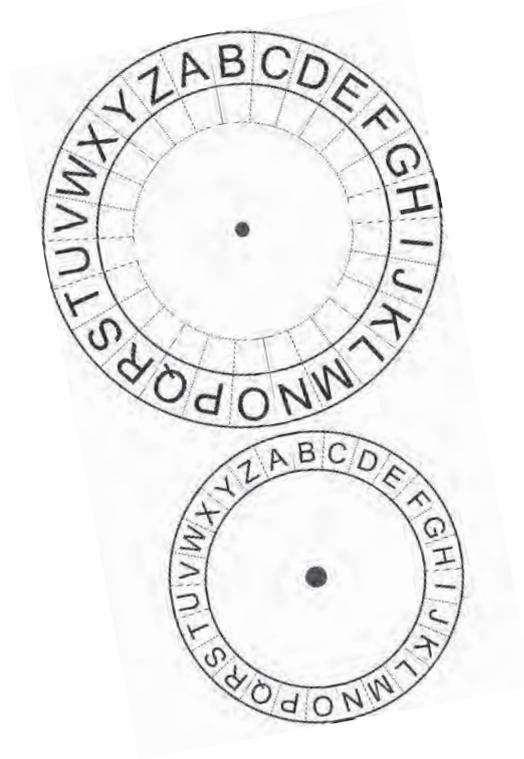


Abbildung: N.Coustou - Julius Caesar. Quelle: (Public Domain) [https://commons.wikimedia.org/wiki/File:Julius\\_Caesar\\_Coustou\\_Louvre.png](https://commons.wikimedia.org/wiki/File:Julius_Caesar_Coustou_Louvre.png) [17.11.2015]

## Bauanleitung

Um eine Verschlüsselungs-Scheibe zu basteln, benötigst du:

- Dünne Pappe
  - Zirkel und Geodreieck
  - Schere
  - Kleber
  - Musterklammer (benutzt man eigentlich zum Verschließen von Versandtaschen)
  - Bleistift, Filzstift
1. Zeichne mit einem Zirkel zwei große Kreise nebeneinander auf eine Pappe. Ein Kreis, sollte einen Durchmesser von etwa 14 cm haben, der andere etwa 10 cm.
  2. Schneide beide Kreise aus.
  3. Nun müssen beide Kreise in 26 Felder unterteilt werden. Dafür ermittelst du mit Hilfe eines Geodreiecks erst einmal den Mittelpunkt. Mit dem Bleistift kannst du den Mittelpunkt aufmalen.
  4. Im nächsten Schritt unterteilst du die beiden Kreise in je zwei Hälften. Der Strich muss dafür immer durch den Mittelpunkt gehen, den du gerade eingezeichnet hast.
  5. Nun muss jede Hälfte in 13 Teile unterteilt werden, dafür setzt du dein Geodreieck an die Mittellinie, schiebst es um 14 Grad weiter und machst erneut einen Strich. Dies machst du, bis du alle nötigen Felder für das Alphabet hast.
  6. Danach schreibst du das Alphabet in die Felder des großen und des kleinen Kreises.
  7. Im letzten Schritt verbindest du beide Kreise mit der Musterklammer, fertig ist die Chiffrier-Maschine.



## Übungsaufgaben

1. Verschlüssele folgenden Satz mit einer Verschiebung um vier Stellen.

„Wir treffen uns um vier Uhr vor der Schule“

\_\_\_\_\_

\_\_\_\_\_.

2. Kannst du die folgende Nachricht entschlüsseln? Die Buchstaben sind um fünf Stellen verschoben:

BNW XHMBFJSEJS MJZYJ INJ XHMZQJ ZSI LJMJS NSX PNST

\_\_\_\_\_

\_\_\_\_\_.

3. Kannst du auch eine Nachricht ohne bekannten Schlüssel knacken?

DOOH PHLQH HQWFKHQ

\_\_\_\_\_.

4. Warum ist das Verschlüsselungsverfahren von Cäsar leicht zu knacken?
5. Welche Möglichkeiten hat man, den Schlüssel der Cäsar-Verschlüsselung herauszubekommen? Gibt es mehrere Möglichkeiten?

# Wie kann man den Code knacken?

Um die Cäsar-Verschlüsselung zu dekodieren, kann man zum einen alle Möglichkeiten der Scheibe ausprobieren. Dafür benötigt man aber sehr viel Zeit.

Eine weitere Möglichkeit, um den Text zu entschlüsseln, ist zu schauen, welcher Buchstabe sehr häufig vorkommt und welche seltener vorkommen. Das liegt an den Eigenschaften einer Sprache. In der deutschen Sprache kommt der Buchstabe E am häufigsten vor und der Buchstabe Q sehr selten. Wenn also in einem Text ein Buchstabe sehr häufig vorkommt, dann ist dieser wahrscheinlich der Buchstabe E.

Probiere es einmal aus:

QV PIUJCZO TMJBMV HEMQ IUMQAMV LQM EWTTBMV VIKP ICABZITQMV  
ZMQAMV.

Häufigster Buchstabe: \_\_\_\_\_

Nutze nun deine Cäsar-Scheibe, um den Satz zu entschlüsseln. Drehe dafür den Buchstaben, der am häufigsten vorkommt, zum E auf der großen Scheibe.

Entschlüsselte Nachricht:

---



---

## Aufgabe

Überlege dir nun selbst eine Nachricht und gib sie **ohne Schlüssel** an deinen Nachbarn oder deine Nachbarin. Kann die Nachricht geknackt werden?

## Häufigkeitsverteilung von Buchstaben in Prozent

<b>E</b>	17,40 %	<b>H</b>	4,76 %	<b>W</b>	1,89 %	<b>Y</b>	0,04 %
<b>N</b>	9,78 %	<b>U</b>	4,35 %	<b>F</b>	1,66 %	<b>X</b>	0,03 %
<b>I</b>	7,55 %	<b>L</b>	3,44 %	<b>K</b>	1,21 %	<b>Q</b>	0,02 %
<b>S</b>	7,27 %	<b>C</b>	3,06 %	<b>Z</b>	1,13 %		
<b>R</b>	7,00 %	<b>G</b>	3,01 %	<b>P</b>	0,79 %		
<b>A</b>	6,51 %	<b>M</b>	2,53 %	<b>V</b>	0,67 %		
<b>T</b>	6,15 %	<b>O</b>	2,51 %	<b>ß</b>	0,31 %		
<b>D</b>	5,08 %	<b>B</b>	1,89 %	<b>J</b>	0,27 %		

# Dosentelefon

Aus einfachen Materialien kann man sich ein eigenes Telefon bauen.

Man braucht dazu:

- 2 leere und saubere Konservendosen ohne scharfe Kanten oder Plastiktrinkbecher
- 1 Nagel
- 1 Hammer
- Ca. 20 m Schnur



Die Konservendosen werden sauber gemacht und abgetrocknet. Nun muss man an der Stelle, an der früher der Deckel war, den inneren Rand mit dem Hammer plattklopfen. Sonst könnte man sich beim Hineinfassen in die offene Dose verletzen.



Danach schlägt man mit Hammer und Nagel je ein Loch in die Mitte der Unterseite beider Dosen. Wenn ihr Plastiktrinkbecher benutzt, solltet ihr einen Nagel über einer Flamme erhitzen und damit Löcher in die Becher brennen.

Durch jedes Loch wird ein Schnurende gesteckt. Dieses wird von innen mit einem Knoten versehen, der so groß sein muss, dass die Schnur nicht mehr aus dem Loch herausrutschen kann.





Nun befindet sich die Schnur als Verbindungskabel zwischen den beiden Dosen beziehungsweise Bechern. Jetzt nimmt sich jede und jeder eine Dose. Die beiden Telefonpartner entfernen sich so weit voneinander, dass die Schnur leicht gespannt ist. Sie darf nicht durchhängen. Sie darf auch nichts berühren oder berührt werden.

Wenn man nun 20 Meter auseinander steht, kann man nicht mehr miteinander sprechen, ohne zu schreien. Jetzt kommt das Dosentelefon ins Spiel: Der Sprechende spricht in seine Dose hinein, der Empfangende hält sich seine Dose mit der offenen Seite ans Ohr. Man hört nun klar und deutlich, wenn auch blechern, sein Gegenüber sprechen!

### Fragen

1. Was hast du beobachtet, wenn die Schnur gespannt ist und wenn die Schnur schlaff ist?
2. Kann man auch um die Ecke telefonieren?
3. Überlegt und versucht herauszufinden, wie das Dosentelefon funktioniert.
4. Stimmt es, dass das Dosentelefon nicht funktioniert, wenn beide Personen gleichzeitig reden? Überlegt euch ein paar Regeln. Wer redet zuerst, wie lange darf jeder reden, wann wird gewechselt, ...?

## Musterlösungen

### B1.1 GS B1.1 Sek. I

#### Aufgabe 2 (GS), Aufgabe 2+3 (Sek I)

Problem	Mögliche Lösung
Aus Versehen geblinzelt, da die Augen z.B. trocken wurden.	Statt blinzeln, das Auge beim entsprechenden Buchstaben öffnen. Den Rest der Zeit bleibt das Auge geschlossen.
Zu spät geblinzelt	Nach einem bestimmten Takt die Buchstaben vorlesen und der Person genug Zeit zum Blinzeln lassen.
Fehlererkennung	Nach jedem Buchstaben bestätigen lassen mit z.B. 1x blinzeln für JA und 2x blinzeln für NEIN, ob der richtige Buchstabe aufgeschrieben wurde.
Übertragung dauert zu lange	Nach ein paar Buchstaben könnte versucht werden, dass Wort zu erraten.
Übertragung dauert zu lange	Tafel auf dem Arbeitsblatt nutzen. Dort sind die Buchstaben nach Häufigkeit sortiert.
Fehlerkorrektur	Nachdem ein Fehler erkannt und mitgeteilt wurde, kann die Fehlerkorrektur beginnen, indem man z.B. den letzten Buchstaben streicht und es nochmal versucht.
Übertragung konnte nicht beendet werden.	Man könnte z.B. 3x blinzeln, wenn man das Wort erfolgreich versendet hat.

#### Aufgabe 3 (GS), Aufgabe 4 (Sek I.)

Besonders schnell können Wörter diktiert werden, wenn die Buchstaben auf der Tafel nach der Häufigkeit ihres Vorkommens (in der jeweiligen Sprache) sortiert sind. Das Bild auf dem Arbeitsblatt zeigt, wie diese Tafel für die deutsche Sprache aussehen würde.

## B1.2 Sek. I

Der römische Kaiser Tiberius nutzte Metallspiegel, welche das Sonnenlicht reflektierten, um sein Reich von der Insel Capri aus zu regieren. (etwa 30 n.Chr.)

Der portugiesische Seefahrer und Ritter Ferdinand Magellan nutzte Kanonen und Flaggen und seine Flotte zu koordinieren. (1520)

Andere Möglichkeiten zur Telekommunikation wären der **Telegraph** (1792 Claude Chappe) oder Morseapparat (1843 Samuel Morse), das **Telefon** (1876 Alexander Graham Bell und Thomas Alva Watson), der **Phonograph** (1877 Thomas Alva Edison), das **Radio** und **Fernsehen** (1920 erste Radioshow in Pittsburgh) (1925 John Logie Baird), das **Fax** (1843 patentiert und 1964 erstmalig kommerziell erhältlich), das **Mobiltelefon** (1947 Douglas H. Ring), **Computer** und das **Internet** (1965 erste eMail versendet), (1969 ARPANET), (1989 Tim Berners-Lee Prototyp WWW)(1994 Internetradio), (2009 Whatsapp)

Ein übersichtlicher Zeitstrahl die Geschichte der Telekommunikation findet man unter folgendem Wikipedia Artikel:

[https://en.wikipedia.org/wiki/History\\_of\\_telecommunication](https://en.wikipedia.org/wiki/History_of_telecommunication)

## B1.3 GS/Sek. I

2.

Man könnte sich überlegen, dass man für jeden Buchstaben eine eigene Armstellung definiert. Dieser Ansatz hat jedoch mehrere Schwächen. Zum einen ist es relativ aufwendig sich für jeden Buchstaben eine eigene Armstellung zu überlegen und zum anderen ist es für den Empfänger schwer, ähnliche Armstellungen voneinander zu unterscheiden bzw. zu interpretieren.

Deutlich einfacher ist es, wenn für die einzelnen Buchstaben Ziffern festgelegt wird. Somit werden nur 10 verschiedene Armstellungen benötigt (Ziffer 0 bis 9). Somit könnte der Buchstabe Z die Zahl 26 dargestellt werden.

Zusätzlich muss man sich noch Regeln überlegen, wie eine exemplarische Übertragung aussehen kann und zusätzliche Armstellungen definieren. Eine mögliche Lösung wäre:

ARMSTELLUNG ZIFFER 2 > EINE SEKUNDE WARTEN > ARMSTELLUNG ZIFFER 6 > EINE SEKUNDE WARTEN > ARMSTELLUNG ÜBERTRAGUNG ABGESCHLOSSEN

Damit wurde z.B. der Buchstabe Z übertragen.

## B1.4 GS

1.

Wenn man mehrere Wörter übertragen möchte, dann kann man die Lesbarkeit durch Trennzeichen verbessern. Je ein Trennzeichen zwischen den Buchstaben und zwei Trennzeichen zwischen den Wörtern positionieren.

|| ..... | .- | .-.. | .-.. | ---- || (HALLO)

## B1.4 Sek. I

1.

Siehe B1.4 GS

5.

Nein, man kann nur Senden oder Empfangen.

6.

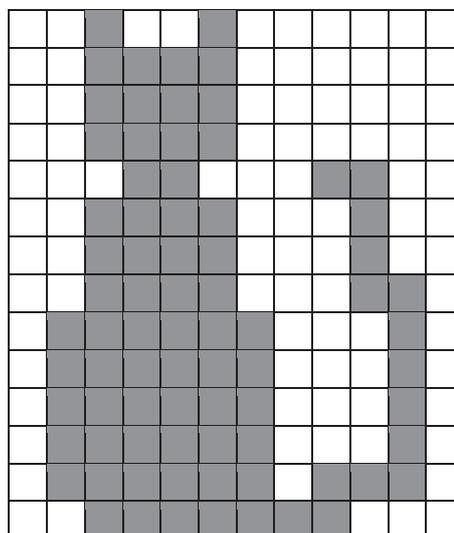
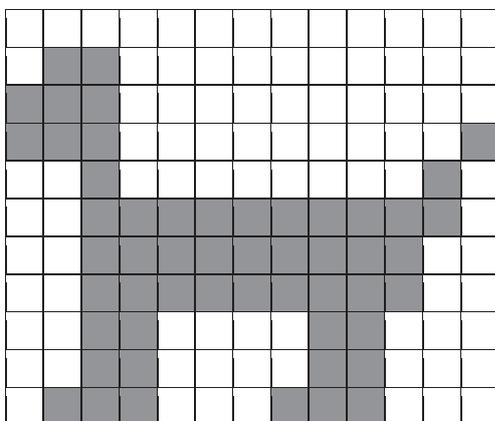
Neben dem Morsealphabet müsste auch ein Zeitraum sichergestellt werden, in dem beide Nachrichten übertragen können. Die Art und Weise (wer zuerst sendet, wie geantwortet wird und so weiter) ist ebenfalls abzusprechen.

7.

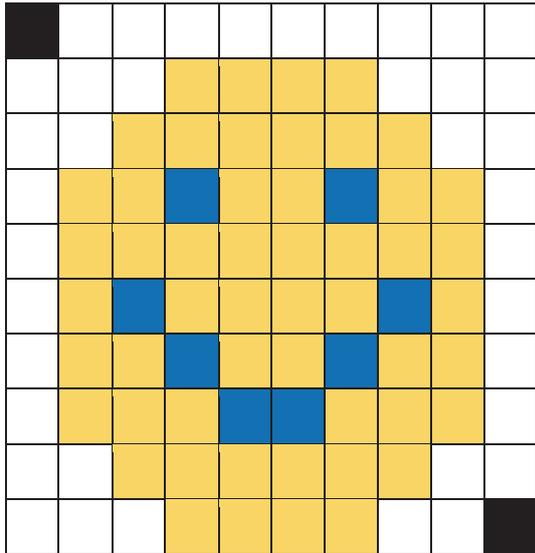
Klingt plausibel die Information so zu kodieren.

## B1.5

### 1.1 Malen nach Zahlen – Aufgabe 1



## 1.2 Farbige Bilder – Aufgabe 1



1, 0, 9, 1

3, 1, 4, 3, 3, 1

2, 1, 6, 3, 2, 1

1, 1, 2, 3, 1, 2, 2, 3, 1, 2, 2, 3, 1, 1

1, 1, 8, 3, 1, 1

1, 1, 1, 3, 1, 2, 4, 3, 1, 2, 1, 3, 1, 1

1, 1, 2, 3, 1, 2, 2, 3, 1, 2, 2, 3, 1, 1

1, 1, 3, 3, 2, 2, 3, 3, 1, 1

2, 1, 6, 3, 2, 1

3, 1, 4, 3, 2, 1, 1, 0



## B1.6 GS und Sek. I

Chiffre: XP GUHL LP NLQR  
Klartext: UM DREI IM KINO

### 1.3 Aufgabe 1

Klartext: WIR TREFFEN UNS UM VIER VOR DER SCHULE  
Chiffre: AMV XVIJJIR YRW YQ ZMIV ZSV HIV WGLYPI

### 1.4 Aufgabe 2 (nur GS)

Chiffre: VXAPNW PRKC NB BYJPQNCCR IDV VRCCJPNBBNW  
Klartext: MORGEN GIBT ES SPAGHETTI ZUM MITTAGESSEN

### 1.5 Aufgabe 2 (nur Sek. I)

Chiffre: BNW XHMBFJSEJS MJZYJ INJ XHMZQJ ZSI LJMJS NSX PNST  
Klartext: WIR SCHWAENZEN HEUTE DIE SCHULE UND GEHEN INS KINO

### 1.6 Aufgabe 3 (nur Sek. I)

Chiffre: DOOH PHLQH HQWFKHQ  
Klartext: ALLE MEINE ENTCHEN

## B1.7 GS

1.

Wenn die Schnur gespannt ist, dann kann man die Person sprechen hören. Bei einer schlaffen Schnur ist dies leider nicht möglich.

2.

Mit einem klassischen Dosentelefon mit Schnur kann man nicht um die Ecke telefonieren; selbst wenn die Schnur gespannt ist, da die Schwingung an der Ecke komplett gedämpft wird. Verbindet man die Dosen allerdings mit einem sehr dünnen Draht kann man auch um die Ecke telefonieren. Der Draht muss dann über dünne Nägel „umgeleitet“ werden.

3.

Töne und Sprache werden in der Luft als Schallwellen übertragen. Spricht man in eine Dose, wird nicht nur die Luft in Schwingung versetzt, sondern auch die Dose. Die Dose gibt die Schwingungen an die Schnur weiter, bis sie an der zweiten Dose ankommen und dort das Trommelfell im Ohr des Zuhörers treffen. Dadurch kann man auch über mehrere Meter Entfernung verstehen, was jemand am anderen Ende sagt.

4.

Eine einfache, aber effektive Regel wäre eine Ankündigung, dass man alles gesagt hat und nun bereit für eine Antwort ist. Ein mögliches Codewort könnte „Over“ sein, sobald die Übertragung abgeschlossen ist und man eine Antwort erwartet. Mit dem Codewort „Over and Out“ kann das Gespräch beendet werden.

# IT2School

Gemeinsam IT entdecken



## Modul B2 – Internet Die Internetversteher

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Inhalt

1	Die Internetversteher .....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Rolle der Unternehmensvertreter*innen.....	4
5	Inhalte des Moduls.....	4
6	Unterrichtliche Umsetzung.....	6
6.1	Einstieg – das Internet als Modell.....	6
6.2	Das Planspiel .....	9
6.3	Entwicklung Sequenz-Diagramm.....	10
6.4	Grober Unterrichtsplan.....	11
6.5	Stundenverlaufsskizzen .....	12
6.5.1	Vorbereitung .....	12
6.5.2	Unterrichtsstunde .....	12
7	Einbettung in verschiedene Fächer und Themen .....	14
8	Anschlussthemen.....	14
9	Literatur und Links .....	15
10	Arbeitsmaterialien .....	15
11	Glossar.....	16
12	FAQs und Feedback.....	16



# 1 Die Internetversteher

Vieles in unserem Alltag wäre ohne das Internet gar nicht denkbar. Das wissen schon die Kinder in der Grundschule, aber wie das virtuelle Netz technisch funktioniert, ist nur den wenigsten bekannt. Dieses Modul erklärt altersgerecht die Funktionsweise des Internets anhand von Pappmodellen und einem Rollenspiel.



Im Rahmen des Rollenspiels übernehmen die Schülerinnen und Schüler selbst die Rolle etwa des Routers, Providers oder des Domain Name System (DNS). Auf diese Weise setzen sie sich aktiv mit den wesentlichen Begriffen auseinander und erfahren, wie Daten von Computern übertragen werden.

<b>Lernfeld/Cluster:</b>	Kommunikation erkunden	
<b>Zielgruppe/Klassenstufe:</b>	X	4. bis 5. Klasse
	X	6. bis 7. Klasse
	X	8. bis 10. Klasse
		11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	2 Stunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Aufbau und Funktion von Informations- und Kommunikationssystemen verstehen</li> <li>• Den Weg einer Internetverbindung kennenlernen</li> <li>• Verstehen, wie Daten von Computern übertragen werden</li> <li>• Hardware, die für den Internetzugang nötig ist, kennenlernen</li> </ul>	
<b>Vorkenntnisse der Schüler*innen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Diagramme und Tabellen lesen und verstehen</li> </ul>	
<b>Vorkenntnisse der Lehrkraft:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Verständnis der Funktionsweise des Internets (ggf. Modul im Vorfeld intensiv durcharbeiten)</li> </ul>	
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Verständnis der Funktionsweise des Internets (ggf. Modul im Vorfeld intensiv durcharbeiten)</li> </ul>	
<b>Sonstige Voraussetzungen:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• zusätzlicher Raum für die Umsetzung, etwa eine Aula oder Sporthalle oder ein weiterer Klassenraum</li> </ul>	

## 2 Warum gibt es das Modul?

Heute ist das Internet fester Bestandteil der Lebenswelt von Schülerinnen und Schülern. Googeln ist seit 2004 als Synonym für „im Internet suchen“ im Duden zu finden. Schon in der Grundschule besitzen 21 Prozent der 6- bis 9-Jährigen ein Smartphone. Grundschulkindern in dieser Altersgruppe sind im Schnitt 15 Minuten täglich im Internet. In der Altersgruppe von 10-12 Jahren sind sie durchschnittlich ca. eine Stunde im Internet. In diesem Alter besitzen schon 86 Prozent ein Smartphone (Statista, Juni 2022).

Aus diesem Grund ist die Vermittlung eines sicheren und kompetenten Umgangs mit dem Internet wesentlicher Bestandteil von Medienbildung, auch schon in der Grundschule. Um Kinder und Jugendliche über die Phänomene des Internets aufzuklären, sollten ihnen auch die Grundlagen und Funktionsweisen des Internets vermittelt werden. Nur auf diese Weise kann ein tieferes Verständnis auch für die Chancen und Risiken digitaler Medien gewonnen werden.

In diesem Modul erfahren die Schülerinnen und Schüler, wie mithilfe einiger Absprachen (Protokolle) die kommunikativen Voraussetzungen für das Internet geschaffen werden und wie es funktioniert. In Form eines Planspiels visualisieren die Kinder und Jugendlichen die entscheidenden Schritte der Kommunikation im Internet. Dadurch verstehen sie, wie das Internet aufgebaut ist und nach welchen Funktionsprinzipien die einzelnen Komponenten zusammenwirken.

## 3 Ziele des Moduls

- frühzeitig das Interesse an Informatik und IT wecken
- Grundverständnis für die Informationsverarbeitung mit Computern schaffen
- den Weg einer Internetverbindung kennenlernen
- verstehen, wie Daten von Computern übertragen werden
- Fachbegriffe kennenlernen und richtig einsetzen (Provider, Cache, URL, DNS,...)
- Aufbau und Funktion von Informations- und Kommunikationssystemen verstehen
- Chancen und Risiken der Vernetzung erkennen und beurteilen

## 4 Rolle der Unternehmensvertreter\*innen

Im *Modul B2 – Internet* kann der\*die Unternehmensvertreter\*in nach vorheriger Absprache mit der Lehrkraft aktiv zum Unterrichtsgeschehen beitragen. So kann er oder sie das Internet anhand des Modells erklären und mit den Schülerinnen und Schülern das Planspiel durchführen. Die Verantwortung und Entscheidungsbefugnis für den Unterricht liegen dabei immer bei der Lehrkraft. Die Unterrichtseinheit sollte von der Lehrkraft und dem\*der Unternehmensvertreter\*in gemeinsam geplant werden, eine vorherige Lektüre der Materialien ist ratsam.

## 5 Inhalte des Moduls

Das Internet ist ein Zusammenschluss mehrerer lokaler Computernetzwerke (wie sie zum Beispiel Zuhause, in der Schule oder in Betrieben existieren) und bildet somit ein globales Com-

puternetzwerk. Generell lässt sich daher von den kleinen lokalen Netzwerken auf das große globale Netzwerk schließen. Die kleineren Netzwerke bestehen aus verschiedenen Komponenten. Eine ist der *Client*, also ein Computer oder allgemeiner ein System, das innerhalb des Netzes kommunizieren möchte. Eine weitere Komponente stellt der *Router* beziehungsweise *Switch* dar, der die Kommunikation zwischen mehreren Clients koordiniert. Zusätzlich gibt es (selbst in kleineren Netzwerken) *Server*, die bestimmte Inhalte und Dienste (wie Webseiten, Dateien, Kalender etc.) innerhalb des Netzwerkes anbieten. Die Kommunikation zum Austausch dieser Dateien oder zur Nutzung der Dienste benötigt das sogenannte *Internet Protocol* (IP), das Regeln und Standards festlegt. Hieraus leitet sich der Begriff *IP-Adresse* ab. Dies ist eine eindeutige vierteilige Zahlenfolge (z. B. 127.0.0.1 oder 192.168.124.2), die jeder beteiligte Computer im Netzwerk (z. B. Client, Router, Switch, Server) besitzt.

Auch im Internet gibt es die bereits erwähnten Komponenten (Client, Router, Server, IP...) und wie im kleinen lokalen Netzwerk steht auch im Internet die Kommunikation an vorderster Stelle. Im Alltag kann diese Kommunikation vieles bedeuten, zum Beispiel den Aufruf einer Internetseite, das Versenden einer E-Mail oder das Streamen von Videos oder Musik. Der Ablauf der Kommunikation im Internet unterscheidet sich jedoch von dem in einem lokalen Netzwerk. So reicht es nicht aus, seinen Computer einfach mit einem Anschluss (LAN, Telefondose etc.) zu verbinden, sondern es wird ein sogenannter *Internet-Service-Provider* (ISP), umgangssprachlich Internetanbieter, etwa die Telekom, 1und1 oder Vodafone, benötigt. Dieser ermöglicht es einem Client (also dem System der Kundin/des Kunden), sich ins Internet einzuwählen (hierfür erhält die Kundin/der Kunde entsprechende Zugangsdaten). Außerdem wird bei Anwendungen und Diensten im Internet in der Regel nicht nur die IP-Adresse verwendet. Zum Einsatz kommen zudem aus Buchstaben und Zeichen bestehende (Internet-)Adressen, *Uniform Resource Locator* (URL) genannt. Da die Kommunikation zwischen Computern jedoch bekanntermaßen im Binärcode, also nur mit Nullen und Einsen funktioniert, benötigt man im Internet das *Domain Name System* (DNS), das URLs in IP-Adressen übersetzt.

Sollten Sie IP-Adressen selbst ermitteln wollen, so ist dies ganz einfach. Es gibt verschiedene Möglichkeiten und Dienste mit deren Hilfe man IP-Adressen herausfinden kann.

Mit Windows selbst lässt sich die IP-Adresse direkt über *Eingabeaufforderung* ermitteln. Man findet die *Eingabeaufforderung* unter *Start > Alle Programme > Zubehör > Eingabeaufforderung*.

Mit einem der beiden Befehle

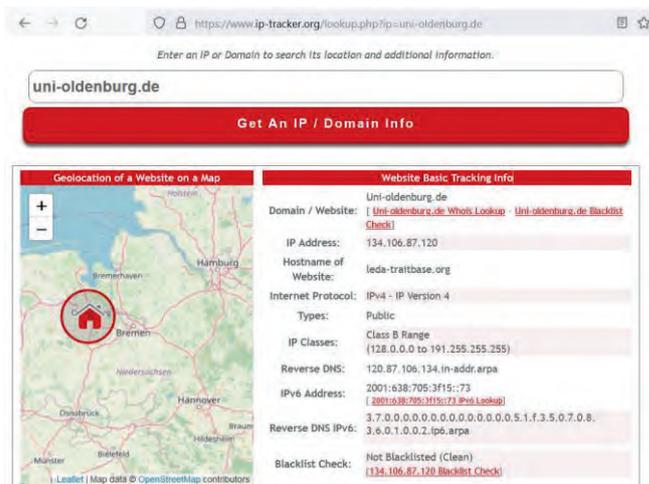
*nslookup*

oder

*ping*

lässt sich die IP-Adresse einer Domain bestimmen.

The image shows two screenshots of a Windows command prompt window. The top screenshot shows the command 'nslookup google.de' being executed. The output displays the server 'schurks.Informatik.Uni-Oldenburg.DE' and the IP address '134.106.11.33'. Below this, it shows the 'Nicht autorisierende Antwort:' for 'google.de' with the IP address '172.217.18.195' highlighted by a red box and an arrow pointing to it. The bottom screenshot shows the command 'ping google.de' being executed. The output shows four successful ping responses from '172.217.18.195' with a response time of 13ms. A red box highlights the IP address '172.217.18.195' in the first line of the ping output, with an arrow pointing to it. The ping statistics at the bottom show 4 packets sent, 4 received, and 0% loss, with a minimum response time of 13ms and a maximum of 13ms.



Eine Alternative bietet die Webseite IP-tracker.org, auf der zusätzlich zu der IP-Adresse auch der Standort des Webservers und viele andere Informationen angezeigt werden.

## 6 Unterrichtsliche Umsetzung

Das Modul ist in drei Teile untergliedert. Zu Beginn wird das Internet als Modell in Form von Pappaufstellern erklärt und aufgebaut. Im zweiten Schritt visualisieren und „erleben“ die Schülerinnen und Schüler die Kommunikation im Internet in Form eines Rollenspiels. Am Ende haben sie die Aufgabe, das Internet schematisch in Form eines Sequenzdiagramms zu beschreiben.

**WICHTIG:** Aus technischen und organisatorischen Gründen ist es nicht möglich, bei den Materialien von IT2School jeweils die eigene Schulhomepage automatisch einzubinden. Daher wird als Platzhalter, die Pseudoadresse: „deine-schule.de“ verwendet. Bei Bedarf kann unter <https://it2school.informatik.uni-oldenburg.de/internetverstehet/index.php> personalisiertes Material für die Schule erstellt und heruntergeladen werden, so dass in den Materialien dann die echte Schulhomepage und die echte IP-Adresse der Schule erscheint.

Außerdem empfehlen wir, die Pappaufsteller in den oberen Ecken leicht einzuschneiden, um die Schnüre leichter an den Aufstellern zu befestigen.

### 6.1 Einstieg – das Internet als Modell

Nach der Begrüßung der Schülerinnen und Schüler fragt die Lehrkraft: „Was glaubt ihr eigentlich wie das Internet funktioniert?“ (Weitere mögliche Fragen: „Wie passt das Internet in euren Computer/Smartphone/Tablet?“ „Wie wird eine E-Mail verschickt?“). Die Antworten werden an der Tafel festgehalten. Als Alternative können die Schülerinnen und Schüler das Pappmodell nach ihren eigenen Vorstellungen aufbauen.

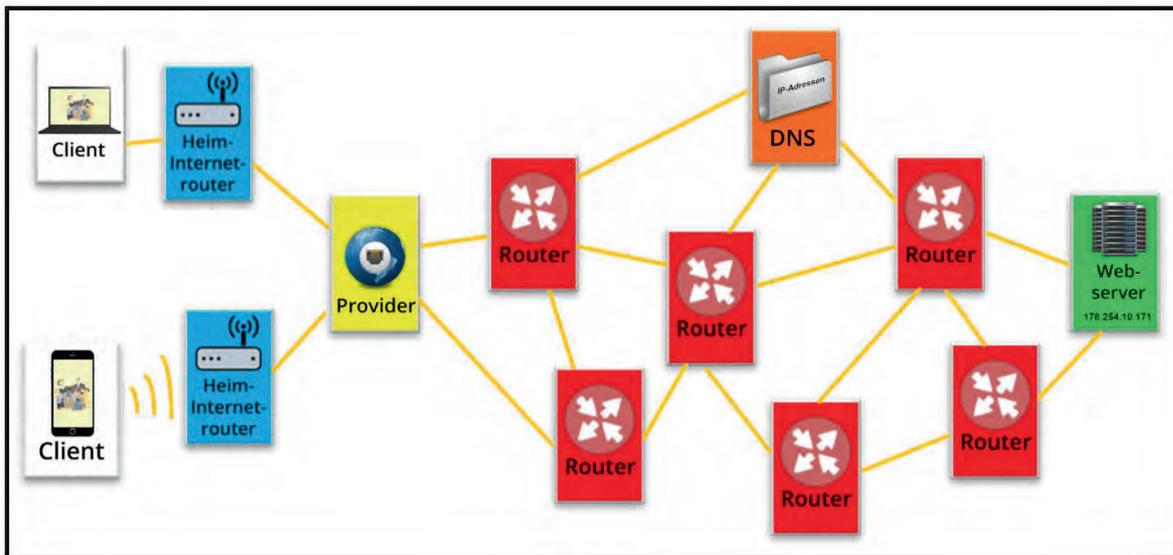
Im Anschluss wird mit dem Pappmodell der idealtypische Aufruf einer Webseite demonstriert. Notwendige Materialien hierfür sind B2.2.1 und B2.2.2. Dazu versammeln sich die Schülerinnen und Schüler in einem Stuhlkreis. Die einzelnen Komponenten werden erklärt, ggf. werden die englischen Begriffe an die Tafel geschrieben.

**Hinweis:** Kursiv gedruckte Begriffe können im Glossar nachgeschlagen werden.

<p><b>Schritt 1:</b> Zu Beginn wird der <b>Client</b> aufgestellt.</p> <p>Lehrkraft: „Der Client bezeichnet (hier vereinfacht dargestellt) den Nutzer eines Computers, Smartphones, Tablets usw.“</p>	 <b>Client</b>
<p><b>Schritt 2:</b> Dazu wird der grüne <b>Webserver</b> mit der IP 178.254.10.171 weit entfernt vom <b>Client</b> aufgestellt.</p> <p>Lehrkraft: „Auf dem <b>Webserver</b> liegt eine Kopie der gewünschten Webseite. Er hat die Aufgabe, Daten (wie beispielsweise Webseiten) zu speichern und zur Verfügung zu stellen.“</p>	 <b>Web-server</b>
<p><b>Schritt 3:</b> „Was benötigt man noch, um eine Webseite vom Server zu bekommen?“</p> <p>Manchmal nennen die Schülerinnen und Schüler schon Router oder WLAN (meinen aber mit beidem den Heim-Internetrouter), da einige das Gerät von zu Hause kennen.</p>	
<p><b>Schritt 4:</b> Im nächsten Schritt wird ein <b>Heim-Internetrouter</b> aufgestellt und mit einer Schnur (Netzwerkkabel) mit dem <b>Client</b> verbunden.</p> <p>Lehrkraft: „Der <b>Heim-Internetrouter</b> stellt für den <b>Client</b> eine Verbindung zum Internet her. Es gibt auch <b>Heim-Internetrouter</b>, die keine Netzwerkkabel brauchen, da sie über WLAN (Wireless Local Area Network) verfügen. Mit WLAN kann man kabellos im Internet surfen“</p>	 <b>Heim-Internet-router</b>
<p><b>Schritt 5:</b> Im Anschluss wird der <b>Provider</b> aufgestellt und mit einer Schnur mit dem <b>Heim-Internetrouter</b> verbunden.</p> <p>Lehrkraft: „Jede*r, der im Internet surfen will, benötigt einen Internet-Anbieter (z. B. Telekom, Kabel Deutschland, Vodafone, 1und1), bei dem man für den Zugang bezahlen muss. Der <b>Heim-Internetrouter</b> meldet sich beim <b>Provider</b> mit persönlichen Zugangsdaten an und erlaubt dann den Zugriff aufs Internet.“</p>	 <b>Provider</b>
<p><b>Schritt 6:</b> Es wird der zweite <b>Heim-Internetrouter</b> aufgestellt, der ebenfalls mit dem <b>Provider</b> verbunden wird.</p> <p>Lehrkraft: „Euer*Eure Nachbar*in, der beim gleichen Provider ist, hat auch einen Heim-Internetrouter.“</p>	 <b>Heim-Internet-router</b>
<p><b>Schritt 7:</b> Es werden die restlichen <b>Client</b>-Aufsteller ins Modell hinzugefügt. Dabei werden aber nicht alle mit einer Schnur mit den Heim-Internetroutern verbunden.</p> <p>Lehrkraft: „Natürlich gibt es auch viele weitere <b>Clients</b>, die aber nicht alle über ein Netzwerkkabel mit dem <b>Heim-Internetrouter</b> verbunden sind.“</p>	 <b>Client</b>

<p>Denn viele Computer und Smartphones nutzen dafür WLAN.“</p>	 <b>Client</b>
<p><b>Schritt 8:</b> Das <b>DNS</b> wird aufgestellt.</p> <p>Lehrkraft: „Die Computer kommunizieren im Internet nicht mit Namen wie <a href="http://www.deine-schule.de">www.deine-schule.de</a> oder <a href="http://www.fragfinn.de">www.fragfinn.de</a>, sondern mit Zahlen, sogenannten IP-Nummern. Diese sind vergleichbar mit einer Postanschrift, und jeder im Internet besitzt eine solche Nummer. Da wir uns aber Namen besser merken können als lange Zahlen, gibt es das DNS (Domain Name System). Dieses System ist vergleichbar mit der Auskunft. Das DNS sagt dem Client, welche Nummer zu welchem Namen gehört.“</p> <p>Als Beispiel können den Schülerinnen und Schüler ein paar Beispiele für IP-Adressen genannt werden (siehe Internetadressliste).</p>	 <b>DNS</b>
<p><b>Schritt 9:</b> Zum Schluss werden alle <b>Router</b> zum Modell hinzugefügt und mit Schnüren verbunden. Ebenso sollten auch der <b>Provider</b>, das <b>DNS</b> und der <b>Webserver</b> mit Schnüren an dieses System aus Routern verbunden werden.</p> <p>Lehrkraft: „Damit eine Nachricht/Anfrage im Internet von einem Ort/Computer zum anderen kommt, benötigt man <b>Router</b>. Sie sind nicht zu vergleichen mit dem <b>Heim-Internetrouter</b>. Die <b>Router</b> dienen als Wegweiser im Internet und reichen eine Nachricht/Anfrage möglichst intelligent von einem zum anderen, bis sie zugestellt werden kann.“</p>	 <b>Router</b>
<p><b>Schritt 10:</b> Nun kann den Schülerinnen und Schüler demonstriert werden, dass man vom <b>Provider</b> über viele Wege zum <b>Webserver</b> kommen kann und dass der Ausfall eines <b>Routers</b> nicht unbedingt dazu führt, dass der <b>Webserver</b> nicht mehr zu erreichen ist. Außerdem kann nun modelliert werden, wie der Client eine Anfrage an den Webserver stellt. Dafür kann beim Webserver ein kleiner Ausdruck der Webseite hinterlegt werden.</p>	

Die folgende Abbildung zeigt den schematischen Aufbau des Modells, das auch auf dem Arbeitsblatt B2.1 dargestellt ist.



## 6.2 Das Planspiel

Nachdem den Schülerinnen und Schülern am Pappmodell der Ablauf der Kommunikation im Internet vorgestellt wurde, stellen sie diese Kommunikation im Rahmen eines Planspiels selber dar. Hierzu werden die Materialien B2.2.3 bis B2.2.6 benötigt. Sie können die einzelnen Stationen in einem zusätzlichen Raum schon vorbereiten oder gemeinsam mit Ihren Schülerinnen und Schülern aufbauen. Jede Station (**Client**, **Heim-Internetrouter**, **DNS**, **Webserver**, **Provider**) wird mit einem farbigen Schild mit der Aufschrift der Komponente versehen. Dort liegen dann auch die benötigten Materialien bereit. So findet sich beim **DNS** zum Beispiel eine Tabelle mit Internetadressen und den IP-Nummern und beim **Webserver** ein Ausdruck der Webseite. Auf dem Boden werden Karten für die **Router** verteilt.

1. Im ersten Schritt überlegen sich die Schülerinnen und Schülern einen Benutzernamen und ein sicheres Passwort für die Provider-Anfrage. In diesem Abschnitt kann auch die Thematik „Sichere Passwörter“ behandelt werden. Weiterführende Informationen dazu finden Sie im Kapitel 9 (Literatur und Links).
2. Die Schülerinnen und Schüler schlüpfen jetzt in die „Rolle“ der einzelnen Stationen **Webserver**, **Client**, **Heim-Internetrouter**, **DNS**, **Provider**. Jede Station wird paarweise besetzt. Für die Zuweisung können die Gruppenkarten ausgeteilt oder von den Schülerinnen und Schülern gezogen werden. Alle Schülerinnen und Schüler, die keiner Station zugeordnet werden, setzen sich als **Router** zwischen die Stationen, so dass sich mindestens zwei (besser mehrere) **Router** bei ausgestreckten Armen mit den Fingerspitzen berühren können.
3. Nachdem die Schülerinnen und Schülern ihre Plätze eingenommen haben, wird ihnen das Protokollheft vorgestellt. Darin stehen die Anfragen und Antworten, die zwischen den Stationen ausgetauscht werden. Wie bei Briefen gibt es immer einen Absender, einen Empfänger und eine Nachricht. Die Absender- und Empfängerfelder sind in den Farben der jeweiligen Stationen hinterlegt.
4. Das Protokollheft beginnt bei der Station **Client**. Die Akteure lesen die Anfrage an den **Heim-Internetrouter** laut vor und reichen das Protokollheft an diese Station weiter.

Daher sollten die Schülerinnen und Schüler, die die Router darstellen, so sitzen, dass sie sich mit den Fingerspitzen berühren und problemlos das Protokollheft weiterreichen können.

5. An jeder neuen Station wird eine Seite im Protokollheft umgeblättert. Die Akteure an dieser Station tragen dort den vereinbarten Benutzernamen und das Passwort ein und lesen die Seite dann laut vor. Im Anschluss wird das Protokollheft an die Zielstation **Provider** weitergereicht.
6. Ist das Protokollheft beim **Provider** angekommen, überprüfen die Akteure den Benutzernamen und das Passwort. Ist die Eingabe korrekt, wird wieder eine Seite umgeblättert und laut vorgelesen. Das OK des **Providers** wird nun an den **Heim-Internetrouter** gesendet. Darüber hinaus wird ihm auch eine IP-Adresse zugeordnet und die IP-Adresse vom **DNS** für den Aufruf von Internetseiten mitgeschickt. (Die verschiedenen IP-Adressen finden zugunsten der didaktischen Reduktion in diesem Planspiel kaum Beachtung und dienen nur zur Unterscheidung der Webserver.)
7. Vom **Heim-Internetrouter** wird das Protokollheft zurück zum **Client** gereicht. Der kann nun seine erste Internet-Suchanfrage starten. Damit ist in diesem Fall das Aufrufen einer bestimmten Homepage gemeint, nicht zu verwechseln mit einer Google-Suche. (Der Client erhält vom **Heim-Internetrouter** auch eine lokale IP-Adresse.)
8. Der **Client** stellt nun eine Anfrage für die Seite „www.deine-schule.de“ beziehungsweise für die eigene Schulhomepage (bitte den Hinweis in Kapitel 5.1 zu Schritt 5 beachten). Das Protokollheft wird über den **Heim-Internetrouter** und über den **Provider** zum **DNS** weitergereicht. Das **DNS** beantwortet die Frage nach der IP-Adresse und schickt die Antwort über den **Provider** zurück zum **Heim-Internetrouter**. Die IP-Nummer zur URL entnehmen die Akteure der Adressliste vom **DNS**, die an der Station bereit liegt.
9. Der **Heim-Internetrouter** fragt nun den **Webserver** mit der IP-Adresse nach dem Inhalt der Website. Die Anfrage sollte über viele Router laufen.
10. Der **Webserver** sendet eine Kopie des Inhalts der Seite an den **Heim-Internetrouter** zurück, hierfür liegen Ausdrucke der angefragten Homepage auf dem Server bereit. Der **Heim-Internetrouter** stellt dann dem **Client** den Inhalt zur Verfügung.
11. Im nächsten Schritt kann eine weitere Anfrage gestartet werden. Diese führt zur Seite FragFinn ([www.fragfinn.de](http://www.fragfinn.de)).

### 6.3 Entwicklung Sequenz-Diagramm

Nachdem die Schülerinnen und Schüler im Modell gesehen haben, wie eine Webseite aufgerufen wird und dies im Planspiel selbst durchgespielt haben, soll gemeinsam mit ihnen ein Sequenzdiagramm entwickelt werden. Hierfür wird das Material B2.3 benötigt. Dadurch soll die Kommunikation im Internet formal dargestellt werden, so dass das Spiel reflektiert und das Gelernte gesichert werden kann.

Mit dem Sequenzdiagramm ist auf einen Blick ersichtlich, wer, wann, was, mit wem im Internet „bespricht“. Zuerst wird das Grundgerüst eines Sequenzdiagramms durch die Lehrkraft an die Tafel gezeichnet. Für jede Komponente wird ein farbiges Rechteck und eine farbige, senkrechte Linie gezeichnet. Die Farben orientieren sich dabei an den Farben des Pappmodells: **Client** = weiß, **Heim-Internetrouter** = blau, **Provider** = gelb, **DNS** = orange, **Webserver** = grün. Zur Komplexitätsreduktion werden **Router** im Sequenzdiagramm nicht berücksichtigt. In das

Rechteck wird später der Name der Komponente geschrieben, die senkrechte Linie bildet die Zeitlinie. Eine Lösung ist den Materialien beigefügt.

#### 6.4 Grober Unterrichtsplan

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Erklärung des Internets anhand eines Modells im Sitzkreis
Planspiel	Durchführung des Planspiels
Erarbeitung	Erarbeitung eines Sequenzdiagramms
Ergebnissicherung	Erstellung eines Sequenzdiagramms in Gruppen

## 6.5 Stundenverlaufsskizzen

### 6.5.1 Vorbereitung

#### Raumgestaltung

Vor dem Unterricht werden verschiedene Stationen im zweiten Raum eingerichtet, die die Komponenten des Internets (**Client**, **Webserver**, **Provider**, **Heim-Internetrouter**, **DNS**) repräsentieren. An diesen Stationen klebt ein farbiges Schild mit der Aufschrift der jeweiligen Komponente. Dort liegen auch die benötigten Materialien bereit. So findet sich beim **DNS** zum Beispiel eine Tabelle mit Internetadressen und den IP-Nummern und beim **Webserver** ein Ausdruck der Homepage. Auf dem Boden werden Karten für die **Router** verteilt.

#### Material

Zur Durchführung des Unterrichts haben wir eine Materialsammlung (siehe B2.2.1 bis B2.2.6) entwickelt, die Pappfiguren für **Client**, **Webserver**, **Provider**, **Heim-Internetrouter**, **DNS** und **Router**, einige Schnüre (als Repräsentation für Netzwerkkabel) und kleine Ausdrücke der Webseiten enthält. Mit diesem Material ist es möglich, ein Modell des Internets aufzubauen. Außerdem enthält dieser Koffer Stationskarten, A4-Ausdrücke der Webseiten, einige Stifte und das sogenannte Protokollheft. Außerdem benötigen wir für den Unterricht noch Arbeitsblätter (siehe B2.3), auf denen ein Sequenzdiagramm gezeichnet werden kann, sowie Karten für die Gruppeneinteilung.

#### Abkürzungen/Legende

AB = Arbeitsblatt/Arbeitsblätter; L = Lehrkraft; MuM = Mitschülerinnen und Mitschüler; SuS = Schülerinnen und Schüler; UV = Unternehmensvertreter\*in

### 6.5.2 Unterrichtsstunde

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
20 Min.	Einstieg	Sitzkreis, frontal	Begrüßung und ggf. Vorstellung der*des UV, Einstiegsfrage: Was glaubt ihr eigentlich wie das Internet funktioniert? Weitere mögliche Fragen: Wie passt das Internet in euren Computer/Smartphone/Tablet? Wie wird eine E-Mail verschickt?. Die Antworten werden an der Tafel festgehalten. Aufbau des Pappmodells und Erklären der Komponenten (Fachbegriffe, Funktionen) und des Zusammenspiels der Internetkomponenten: Client, Heim-Internetrouter, Router, Provider, DNS und Webserver (ggf. Begriffe an die Tafel schreiben).	Figuren/Pappaufsteller, Miniatur-Ausdrücke der Webseiten, Erläuterung „Hinweise zum Pappmodell (Einstieg)“, B2.2.1 bis B2.2.2

5 Min.	Einstieg	Simulation Aufbau	Vorbereitung des Planspiels: Ablauf des Spiels wird erklärt und die Rollen werden verteilt (je Station ein bis zwei SuS als <b>Client</b> , <b>Heim-Internetrouter</b> , <b>Provider</b> , <b>DNS</b> und <b>Webserver</b> , verbleibende SuS verteilen sich als <b>Router</b> ). Die <b>Router</b> stehen so weit auseinander, dass sie sich mit den Fingerspitzen berühren können.	Material laut Materialliste („Rollenkarten“ und „Für die Stationen“), Krepp-Band, 3 Stifte, B2.2.3 bis B2.2.6
30 Min.	Simulation	Planspiel	Planspiel durchführen: 1. Anfrage der Seite „www.deine-schule.de“ bzw. der eigenen Schulhomepage 2. Anfrage der Homepage <a href="http://www.fragfinn.de">www.fragfinn.de</a> 3. Was passiert bei einem Tippfehler? Anfrage der Homepage <a href="http://www.fragfinn.de">www.fragfinn.de</a> . Durch den Tippfehler wird eine Fehlermeldung „Fehler: Server nicht gefunden“ ausgegeben.	Ausdrucke der Webseiten, B2.2.3 bis B2.2.6
10 Min.	Sicherung	Frontal	Besprechung/Reflexion der Simulation: - Welche Komponenten sind beteiligt? - Was macht der <b>Client</b> , <b>Provider</b> ,...? - In welcher Reihenfolge finden die Anfragen statt?	
15 Min.	Sicherung	Gruppenarbeit	Aufteilung in Gruppen, Arbeitsauftrag: Sequenzdiagramm ausfüllen. Im Anschluss wird das Sequenzdiagramm im Plenum besprochen.	Gruppenkärtchen, B2.3, bunte Stifte der SuS
10 Min.	Didaktische Re-serve	Frontal, Plenum	Frage: Oft hört man „Das Internet ist kaputt“. Kann das wirklich sein?	

## 7 Einbettung in verschiedene Fächer und Themen

Dieses Modul ist für alle Unterrichtsfächer geeignet, da bei ihm die Medienbildung im Vordergrund steht, die eine Querschnittsaufgabe aller Fächer ist.

Im Folgenden werden die Kompetenzen aufgeführt, die sich aus den Bildungsstandards der Kultusministerkonferenz oder der einzelnen Rahmenlehrpläne der Länder ergeben.

### Informatik/Medienbildung

Die Schülerinnen und Schüler ...

- kennen die zentralen Komponenten des Internets.
- kennen den Aufbau und die Funktionsweise des Internets.
- können Kommunikationswege im Internet beschreiben.
- beurteilen die Sicherheit der Kommunikation.
- kennen Maßnahmen zur Erstellung sicherer Passwörter.

## 8 Anschlusssthemen

Als Anschlusssthemen im Zusammenhang mit IT2School bieten sich folgende Module an:

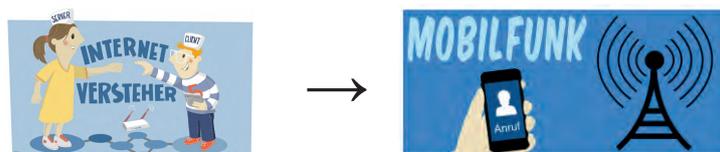
### Beispiel: Leichter Einstieg ohne Technik

Möchten Sie weiter analog arbeiten, dann empfehlen wir das Modul *Codes im Supermarkt und Unternehmen*.



### Beispiel: Mobilfunk

Neben dem Internet ist auch die Kommunikation im Mobilfunk interessant. Zusätzlich zur Funktionsweise des Mobilfunks erfahren die Schülerinnen und Schüler im Modul A1, welche Daten bei der Nutzung des Smartphones und Mobilfunks gesammelt werden und wie diese weiterverarbeitet werden können.



## 9 Literatur und Links

- Film: Sachgeschichten mit der Maus - **Wie funktioniert das Internet?** Online: <http://www.wdrmaus.de/sachgeschichten/sachgeschichten/internet.php5>
- Das **Internet-ABC** bietet Kindern und Erwachsenen Infos, Tipps und Tricks rund um das Thema Internet (Funktionsweise des Internets, Computerspiel „Netzwerkmeister“, Umgang, Surfschein, etc.) Online: <http://www.internet-abc.de>
- **Die Internauten.** Lehrer\*innenmaterial zum sicheren Umgang mit dem Internet und Datensicherheit. Online: <http://www.internauten.de/Lehrermaterial/Lehrerhandbuch.pdf>
- **Soekia**-ein Blick hinter die Kulissen. Didaktische Suchmaschine mit deren Hilfe man die Funktionsweise erklären kann: <http://www.swisseduc.ch/informatik/soekia/index.html>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B2.1	Spielaufbau	Arbeitsblatt zum schematischen Aufbau des Modells.
 B2.2,  B2.2.1,  B2.2.2,  B2.2.3,  B2.2.4,  B2.2.5,  B2.2.6	Bastelmaterial	Bastelmaterialien für das Pappmodell sowie für die einzelnen Stationen.
 B2.3	Sequenzdiagramm	Arbeitsblätter zur Erstellung eines Sequenzdiagramms, in dem der Ablauf beim Aufruf einer Internetseite dargestellt wird. Musterlösung vorhanden.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Browser	Computersoftware, die auf einem Endgerät installiert ist und mit dem Server kommuniziert, z. B. Internet Explorer, Mozilla Firefox.
DNS	Domain Name System – ähnlich wie eine Telefonauskunft; gibt für einen Domainnamen wie <a href="http://www.fragfinn.de">www.fragfinn.de</a> die dazugehörige IP-Nummer aus.
IP-Adresse	Eindeutige vierteilige Zahlenfolge (z.B. 127.0.0.1 oder 192.168.124.2), die jeder beteiligte Computer im Netzwerk (z.B. Client, Router, Switch, Server), aber auch jede Homepage besitzt.
Medienserver	Ermöglicht den Zugriff auf Dateien im Heimnetz, z. B. über eine angeschlossene USB-Festplatte.
Provider	Internetdienstleister, bietet alle Leistungen an, die zur Benutzung oder für den Betrieb des Internets nötig sind.
Router	Vermittelt zwischen den Rechnern im Heimnetz und den Rechnern im Internet.
WLAN-Access Point	Stellt WLAN-Funktionalität zur Verfügung.
WLAN	Wireless Local Area Network, ermöglicht kabelloses Surfen im Internet

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

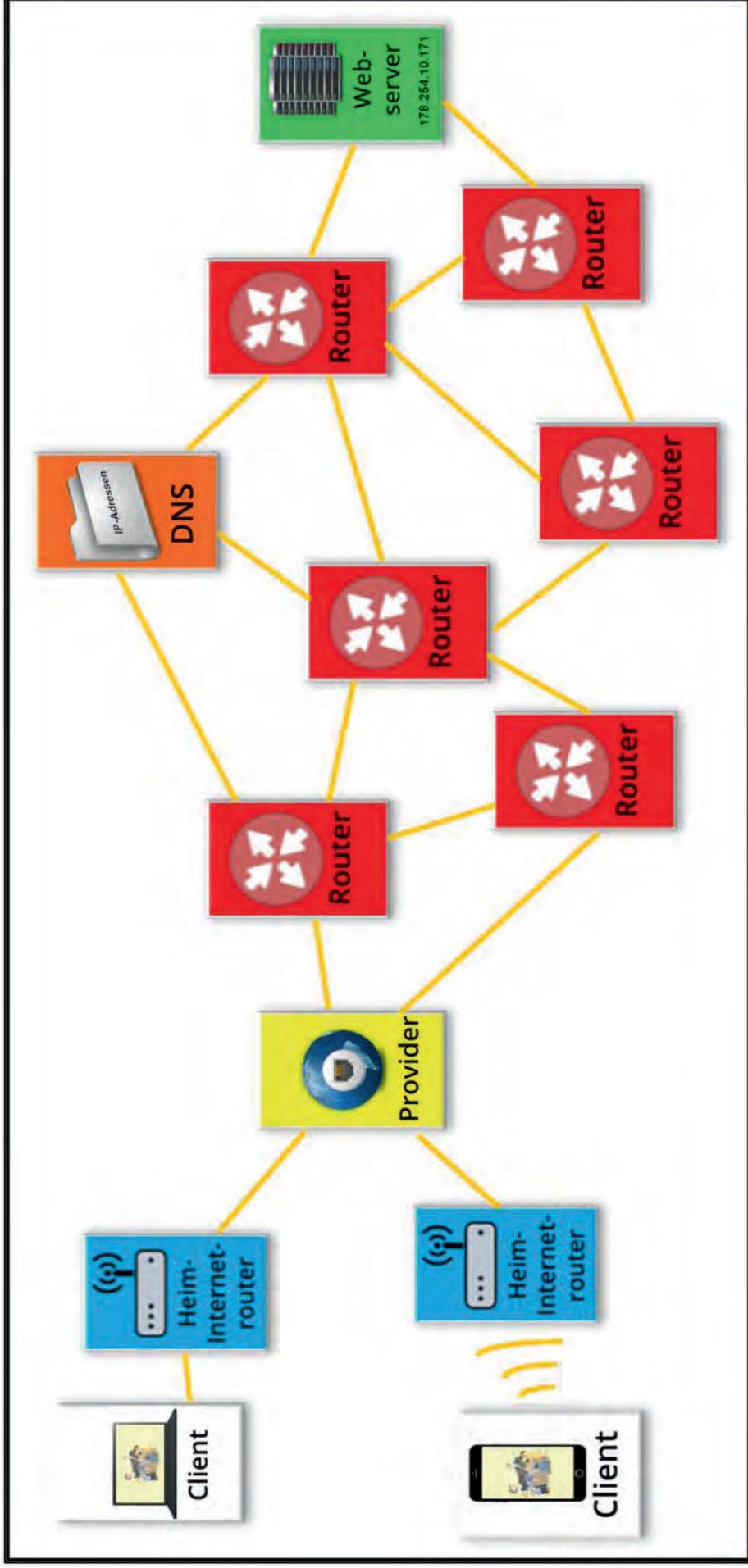
Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Modellaufbau

## Arbeitsmaterial B2.1



# Materialliste

Bitte beachten Sie, dass die **rot gedruckten Materialien** in der vollständigen Materialsammlung der Wissensfabrik, nicht aber in der Print-Version dieses Unterrichtsentwurfs enthalten sind

### Für das Pappmodell:

- Figuren: Client, Heim-Internetrouter, Provider, DNS, 2 Webserver, 9 Router, **Schnüre als Netzwerkabel**
- Kleine Kärtchen: 2x Internetseite der Schule und 2x häufig genutzte Internetseite

### Für die Stationen:

- Stationskarten: Client, Heim-Internetrouter, Provider, DNS, Webserver, 20 Router
- Protokollheft (an der Client-Station hinterlegen)
- Internet-Adressliste von DNS (an der DNS-Station hinterlegen)
- Gruppenkärtchen (für die Einteilung in Gruppen)
- 2x Internetseite der Schule – groß (an der Webserver-Station hinterlegen)
- 2x häufig genutzte Internetseite – groß (an der Webserver-Station hinterlegen)

### Arbeitsblätter:

- Sequenzdiagramm – Aufgabenblatt
- Sequenzdiagramm – Abruf einer Internetseite als Lösungsvorschlag

### Außerdem werden gebraucht:

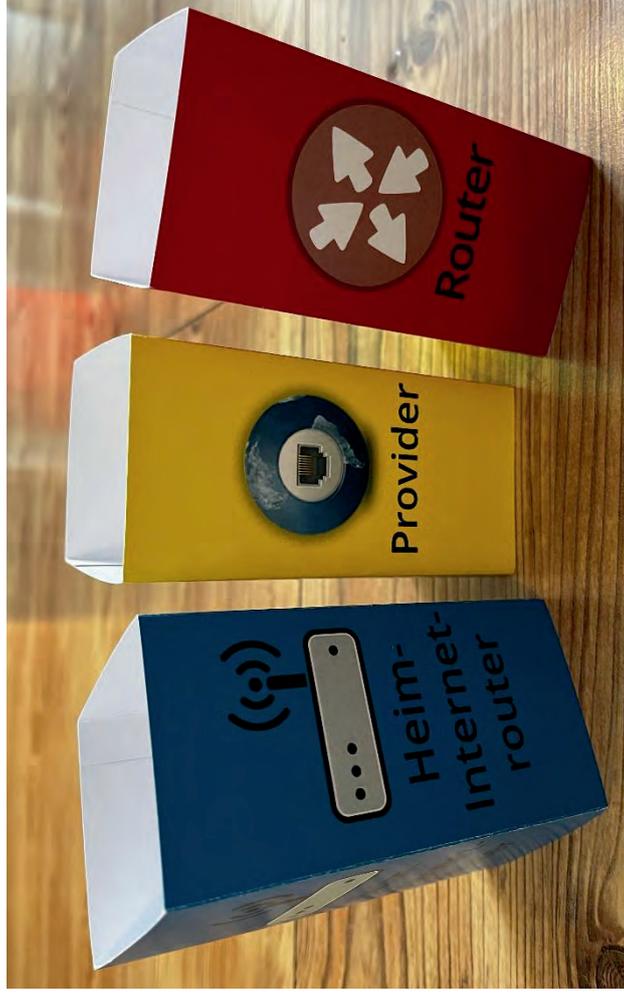
- **Farbige Tafelkreide (weiß, gelb, orange, blau, grün)**
- **3 Stifte (Beschriftung des Protokolls)**
- **Krepp-Klebeband**

# Figuren

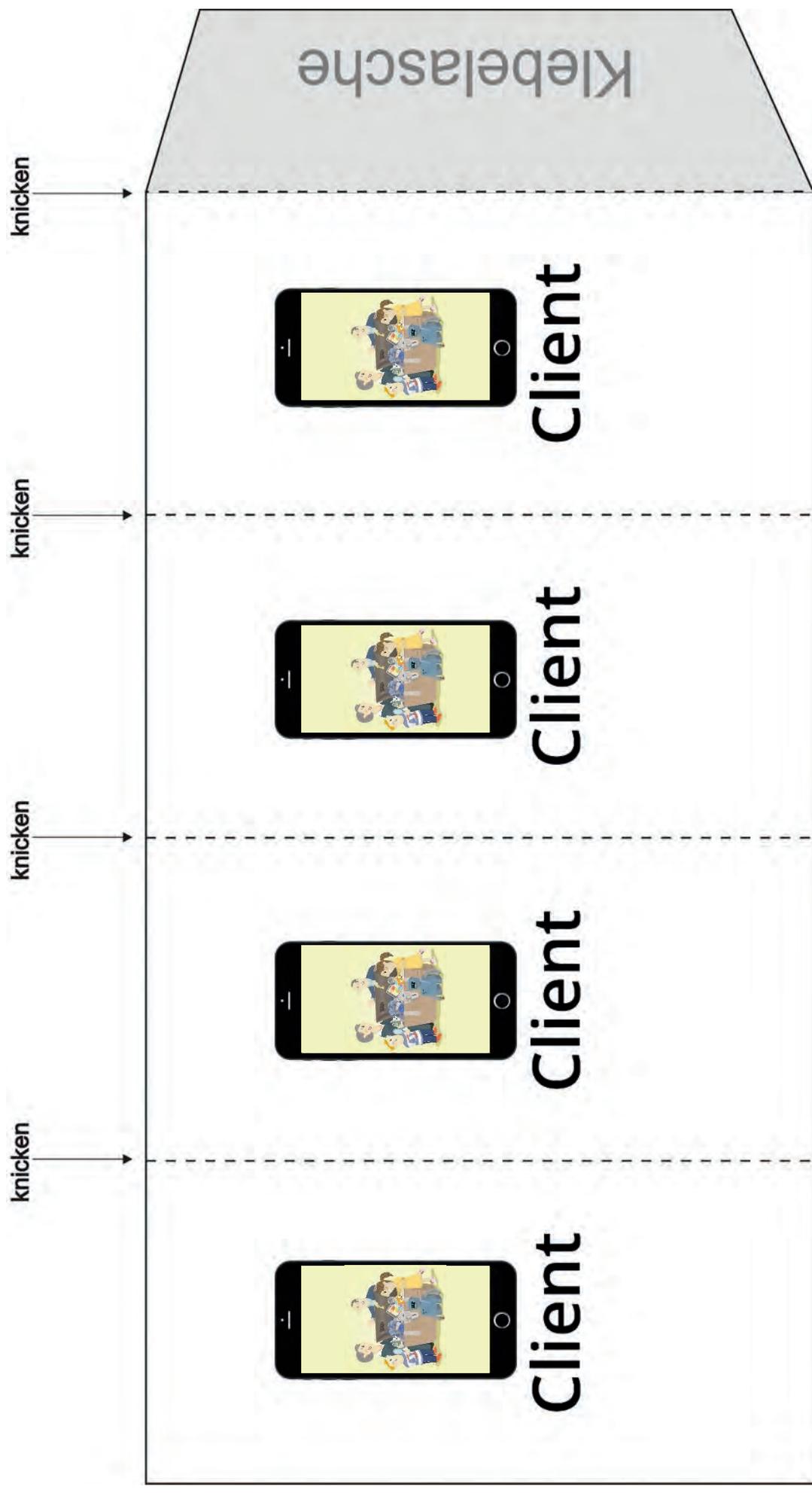
... für das Internet-Model.

Bitte die folgenden Seiten farbig auf festem Papier ausdrucken, zurechtschneiden, knicken und zusammenkleben (siehe untere Abbildung). Es wird empfohlen, die Aufsteller in den oberen Ecken leicht einzuschneiden, um die Schnüre leichter zu befestigen.

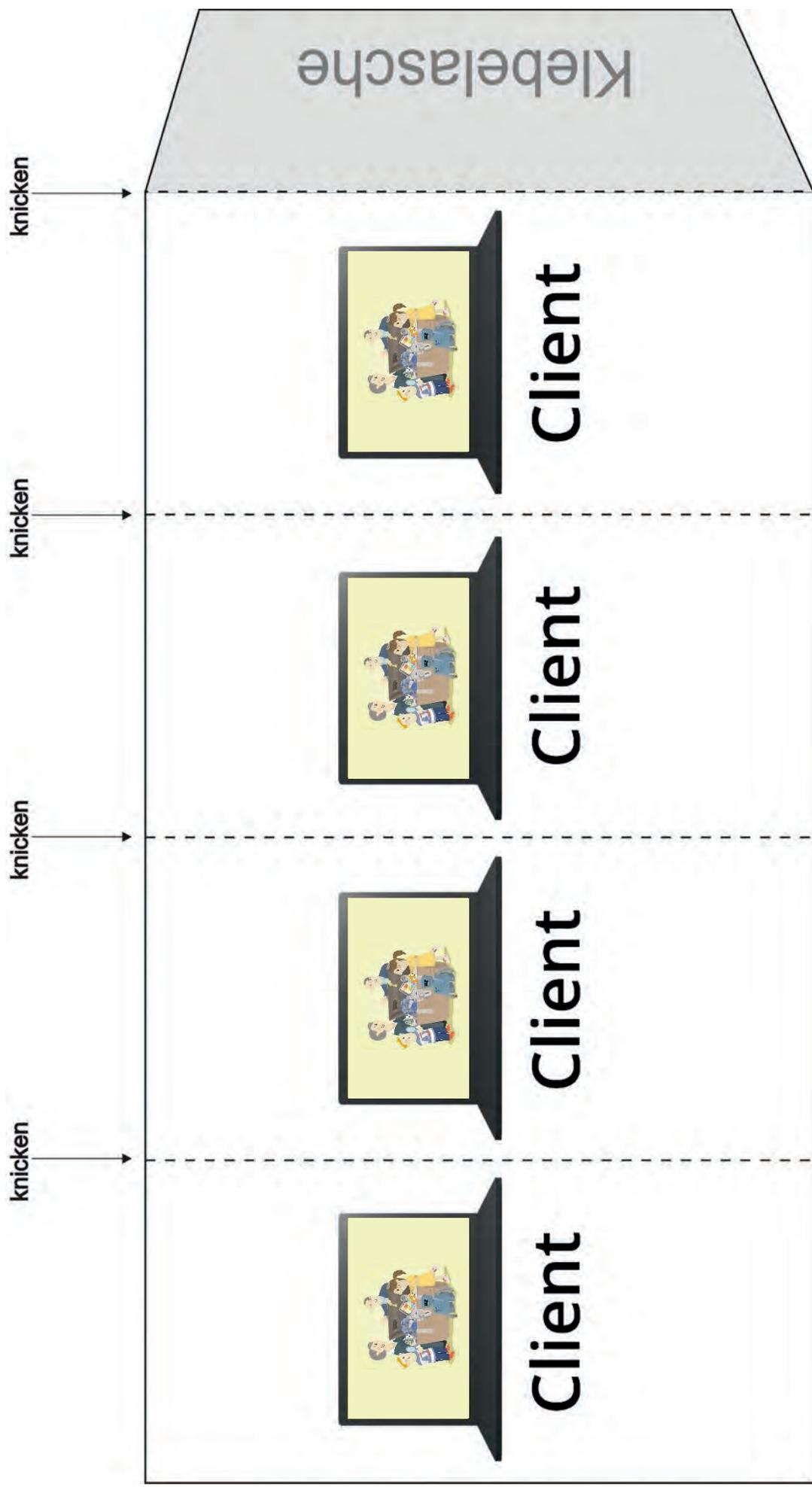
Ecken leicht einzuschneiden, um die Schnüre leichter zu befestigen.



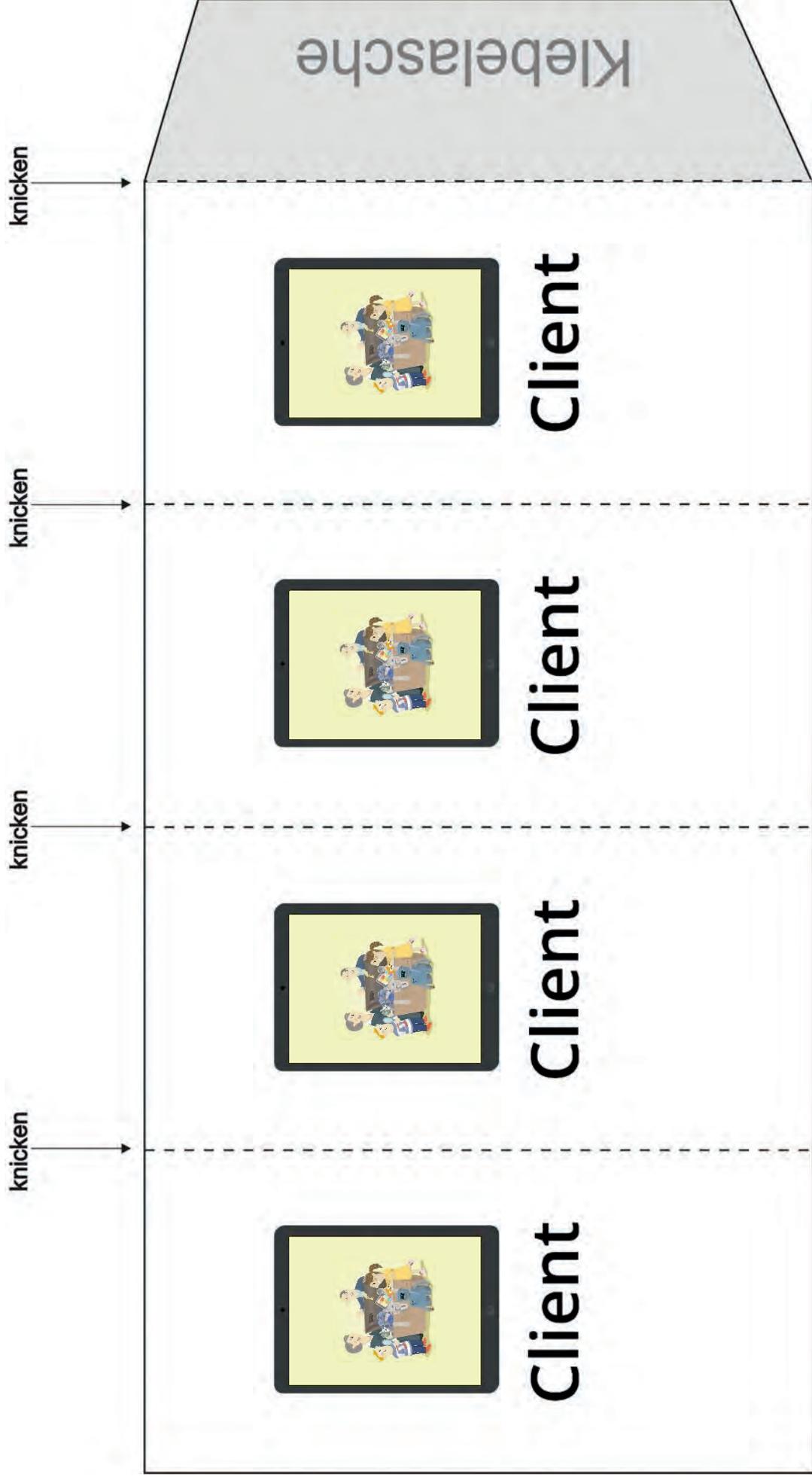
## Arbeitsmaterial B2.2.1



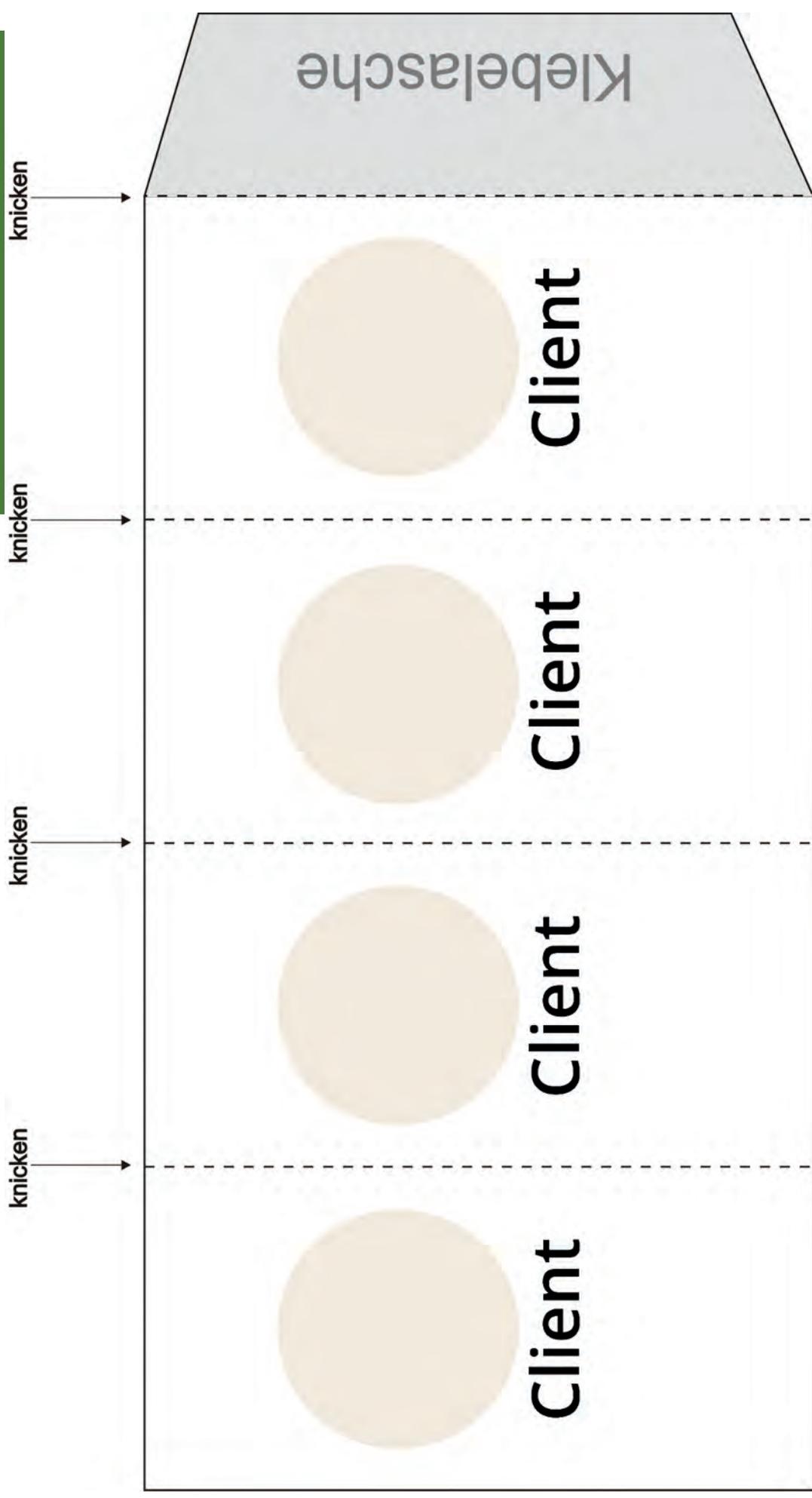
## Arbeitsmaterial B2.2.1



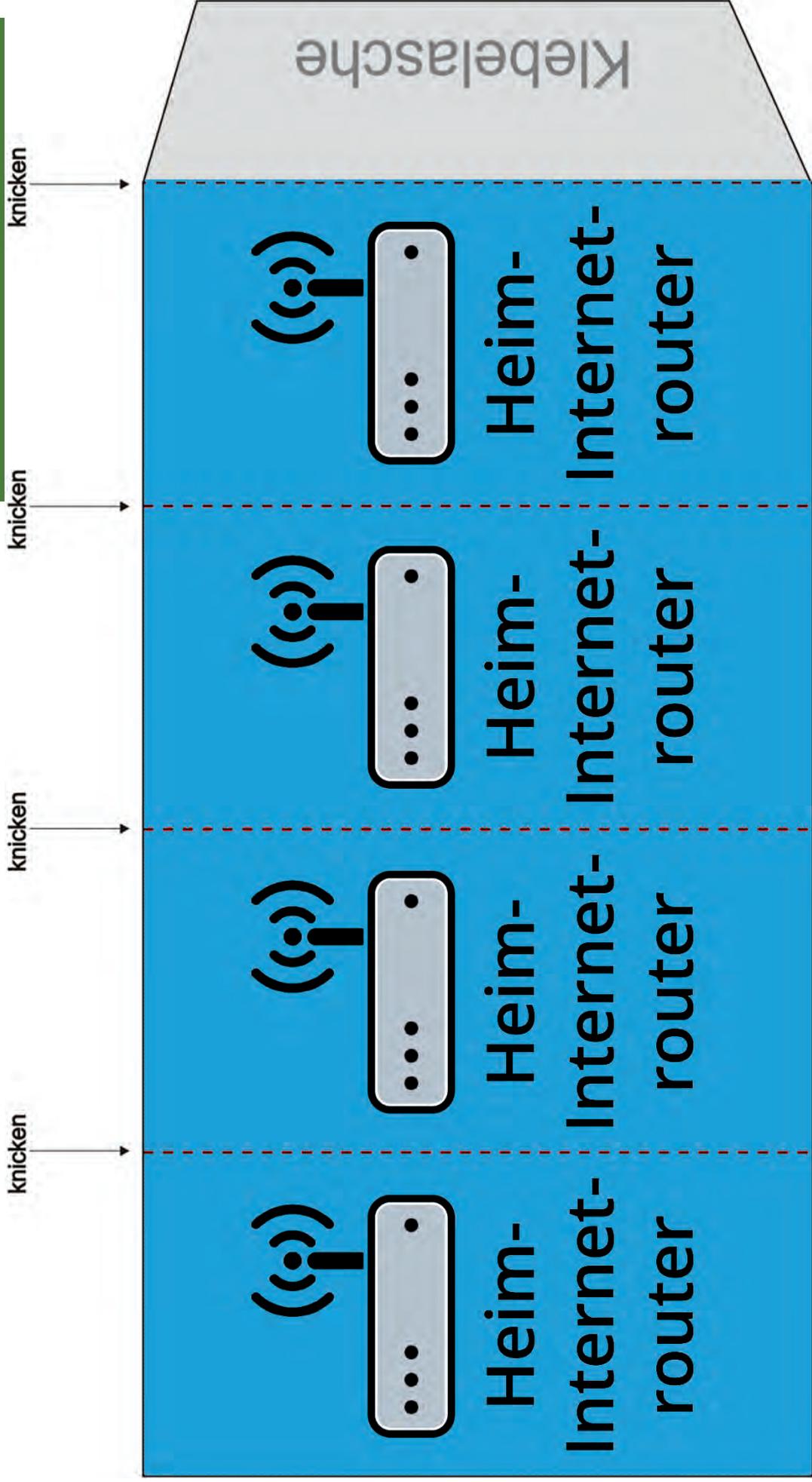
## Arbeitsmaterial B2.2.1



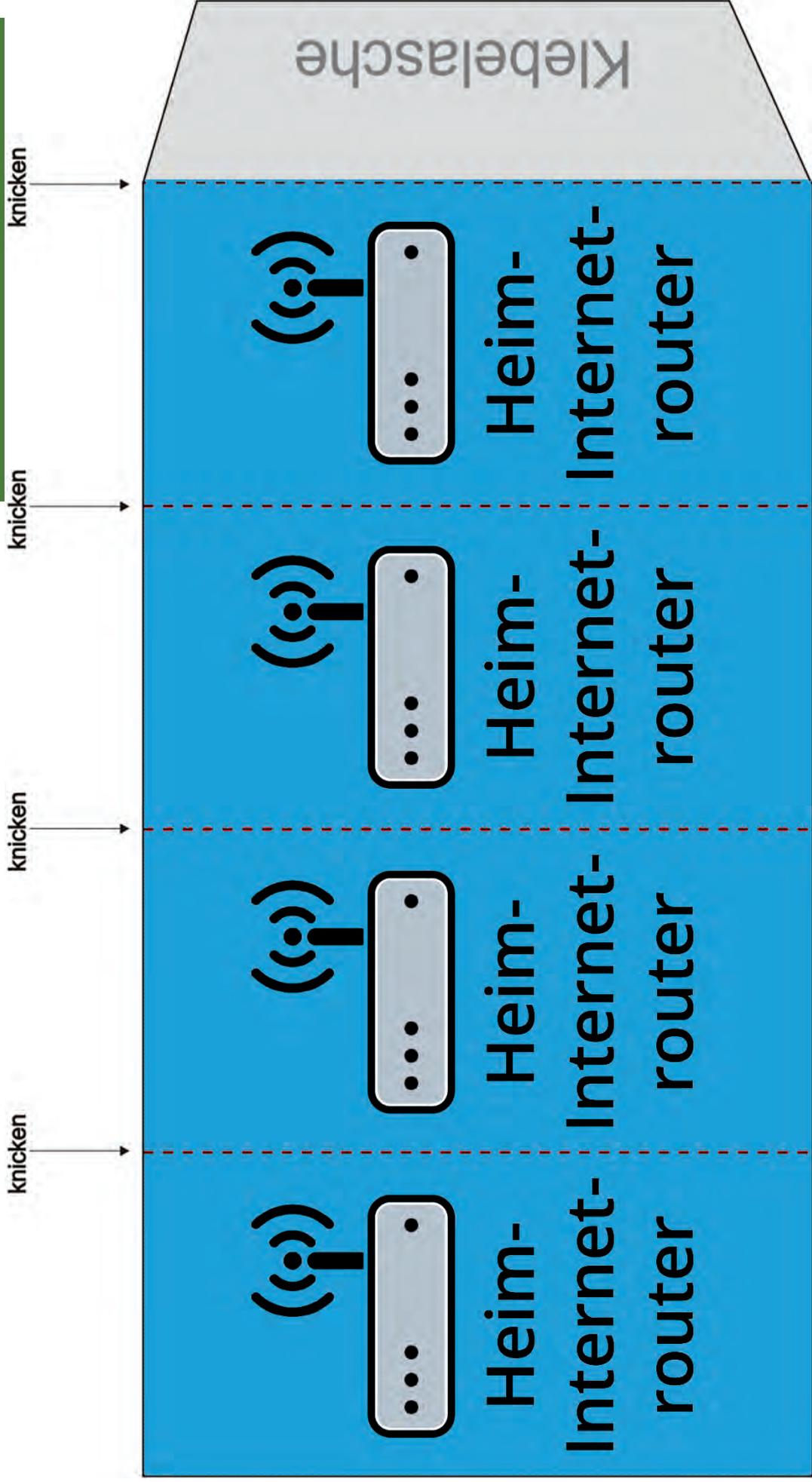
## Arbeitsmaterial B2.2.1



# Arbeitsmaterial B2.2.1



# Arbeitsmaterial B2.2.1



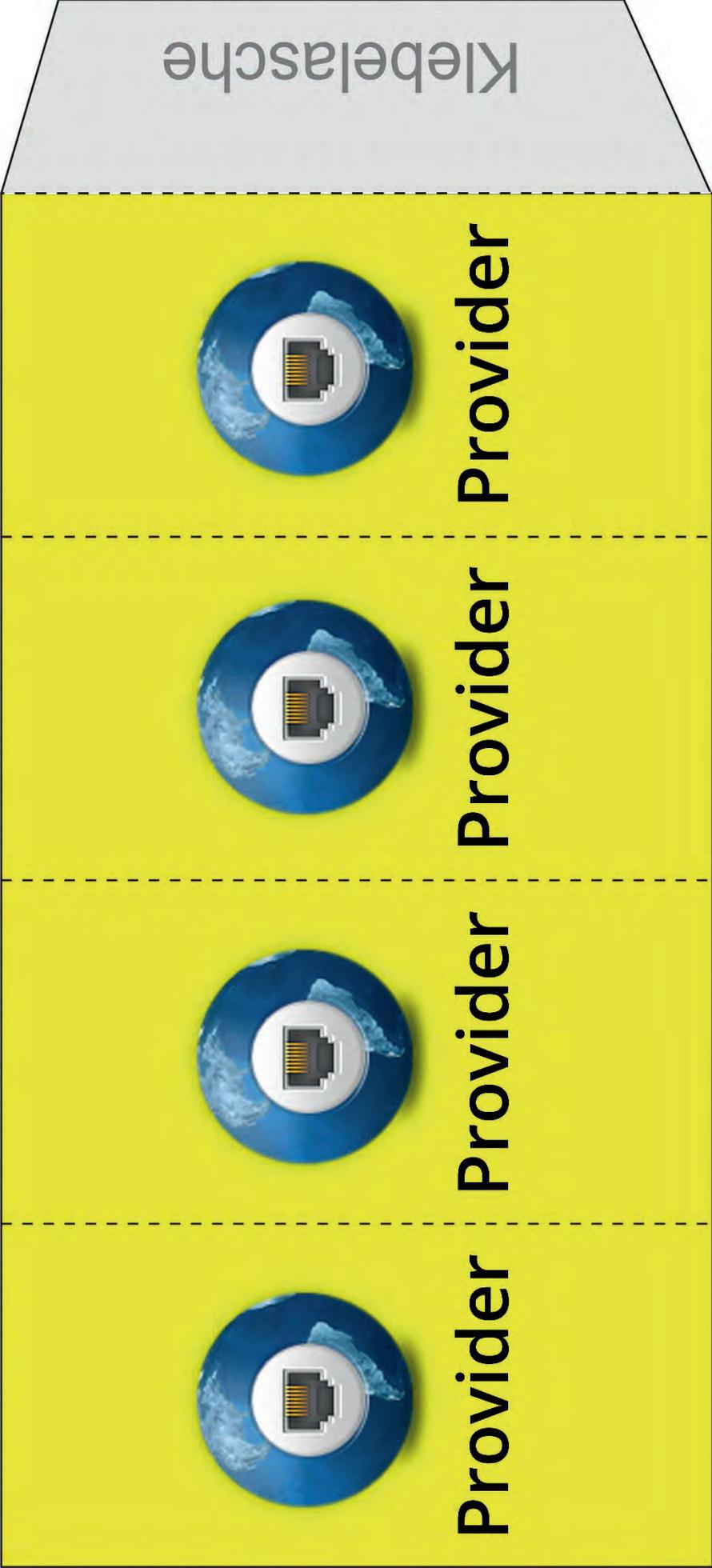
**Arbeitsmaterial B2.2.1**

knicken

knicken

knicken

knicken



Klebelasche

Provider

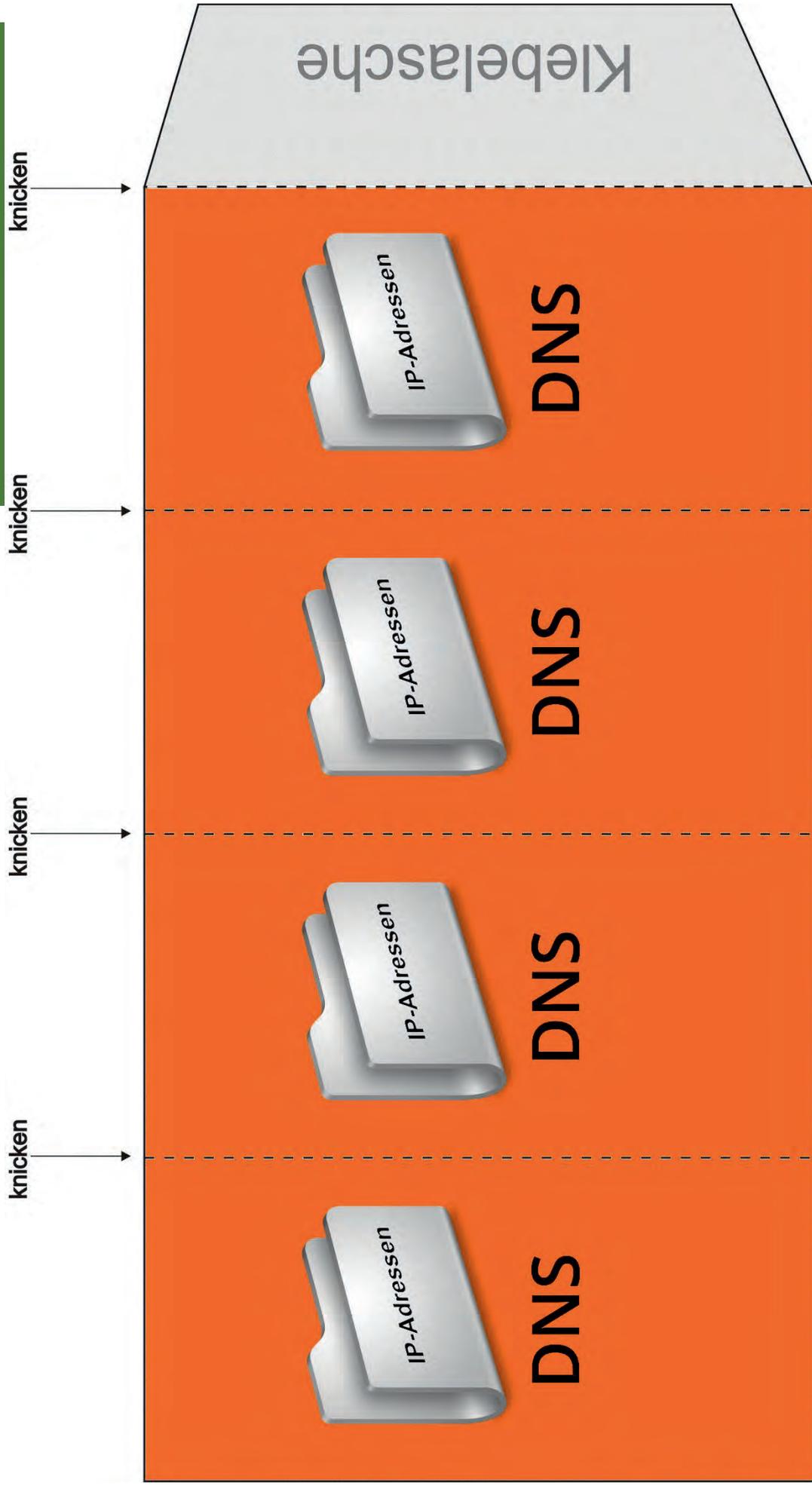
Provider

Provider

Provider



## Arbeitsmaterial B2.2.1



# Arbeitsmaterial B2.2.1

knicken

knicken

knicken

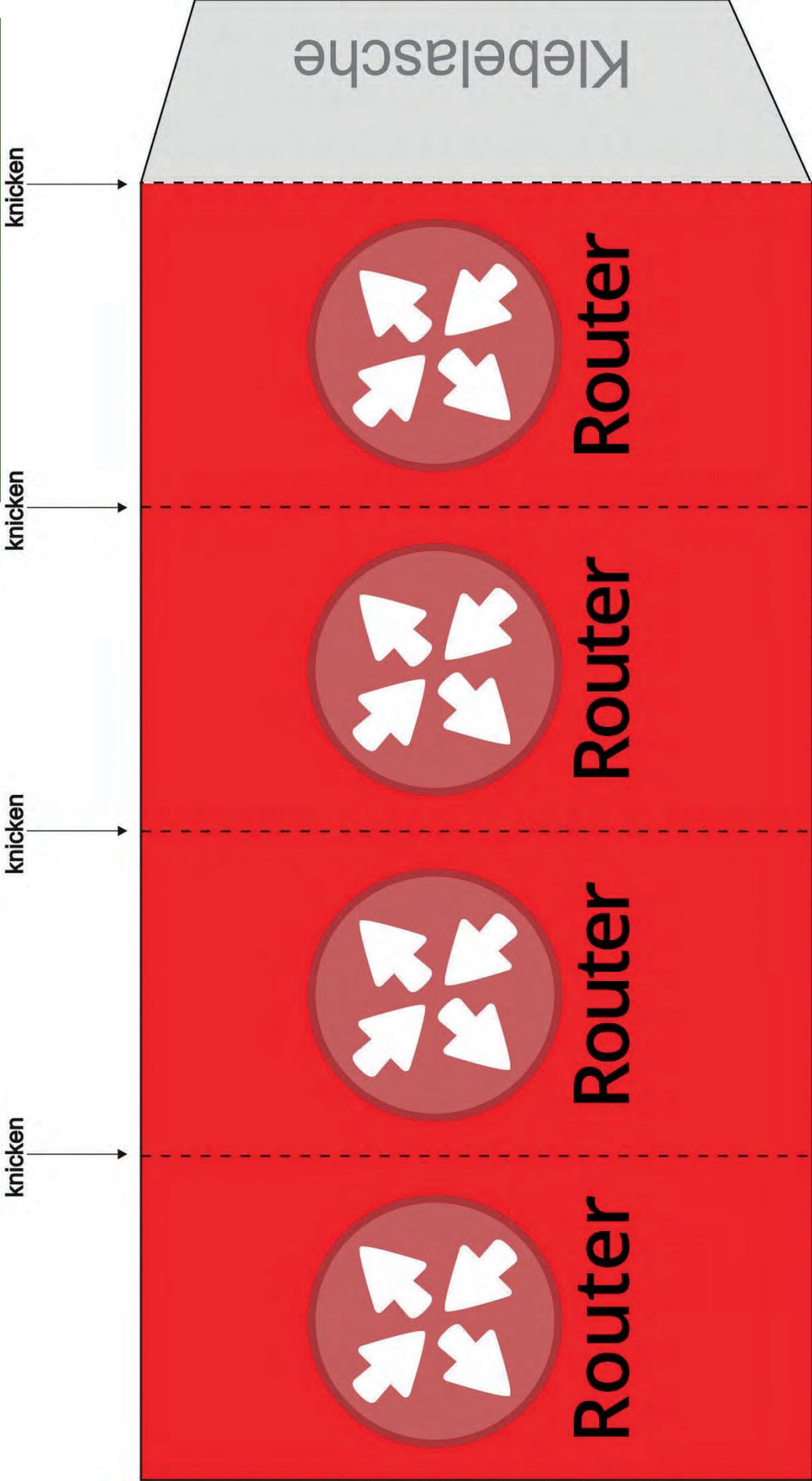
knicken



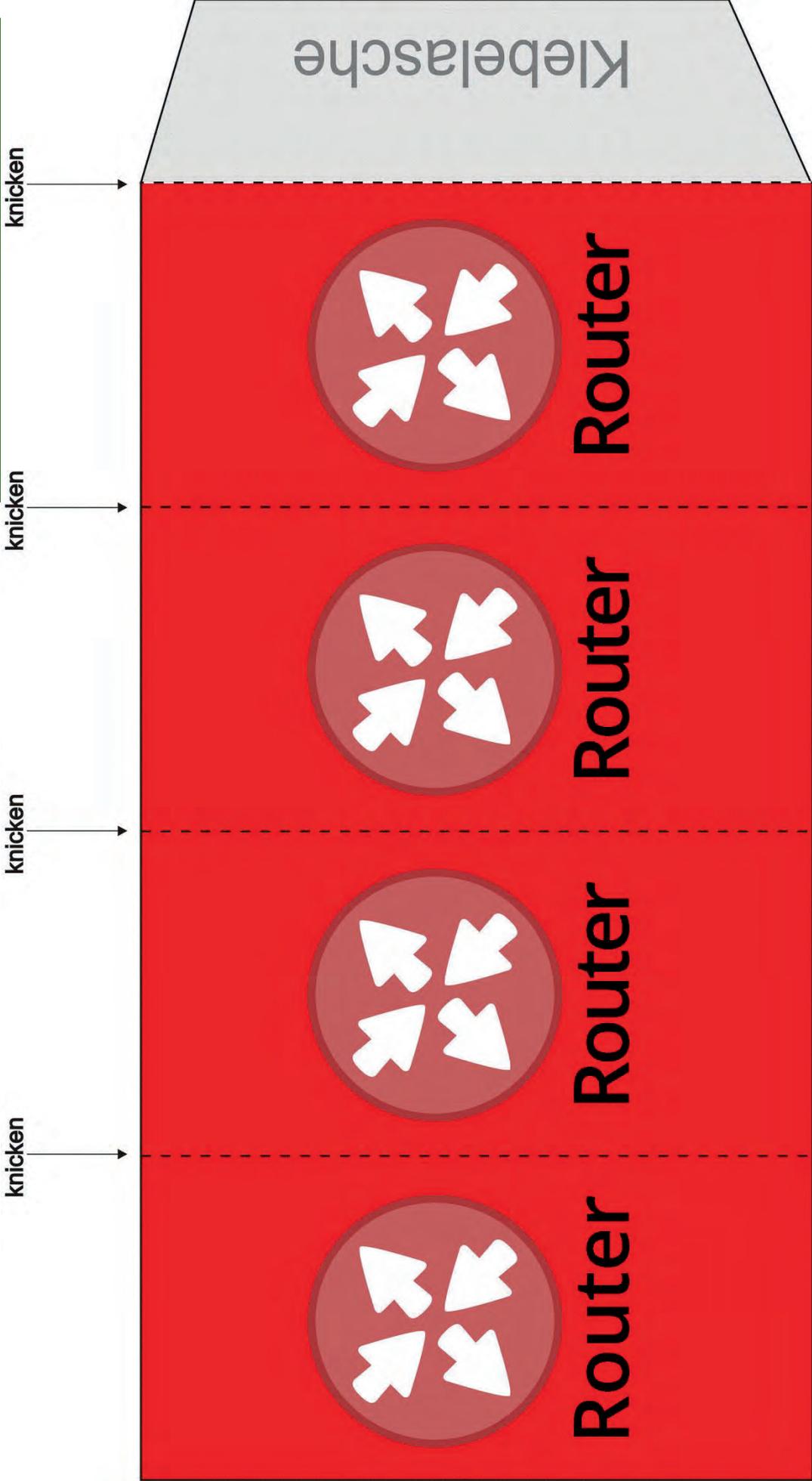
# Arbeitsmaterial B2.2.1



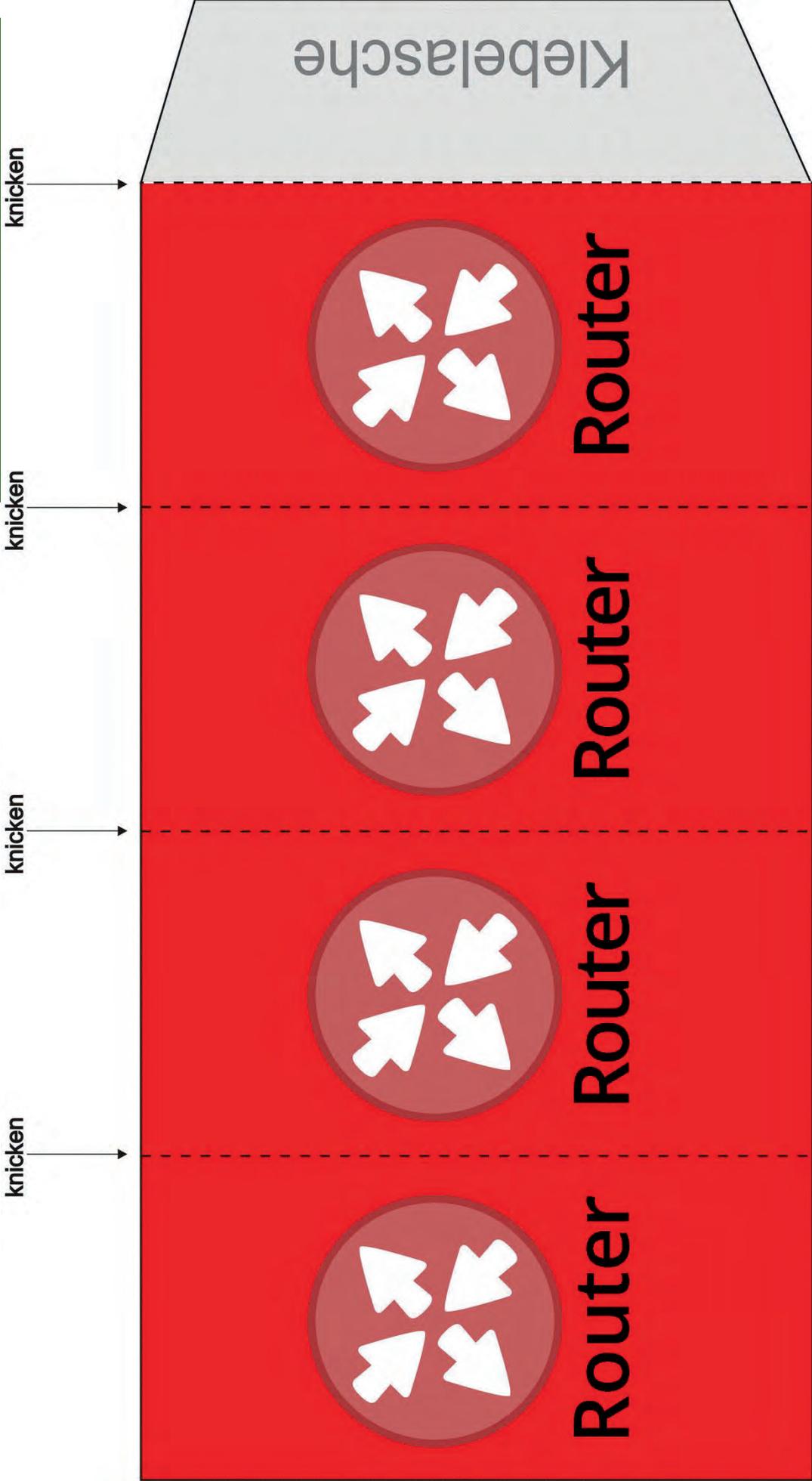
**Arbeitsmaterial B2.2.1**



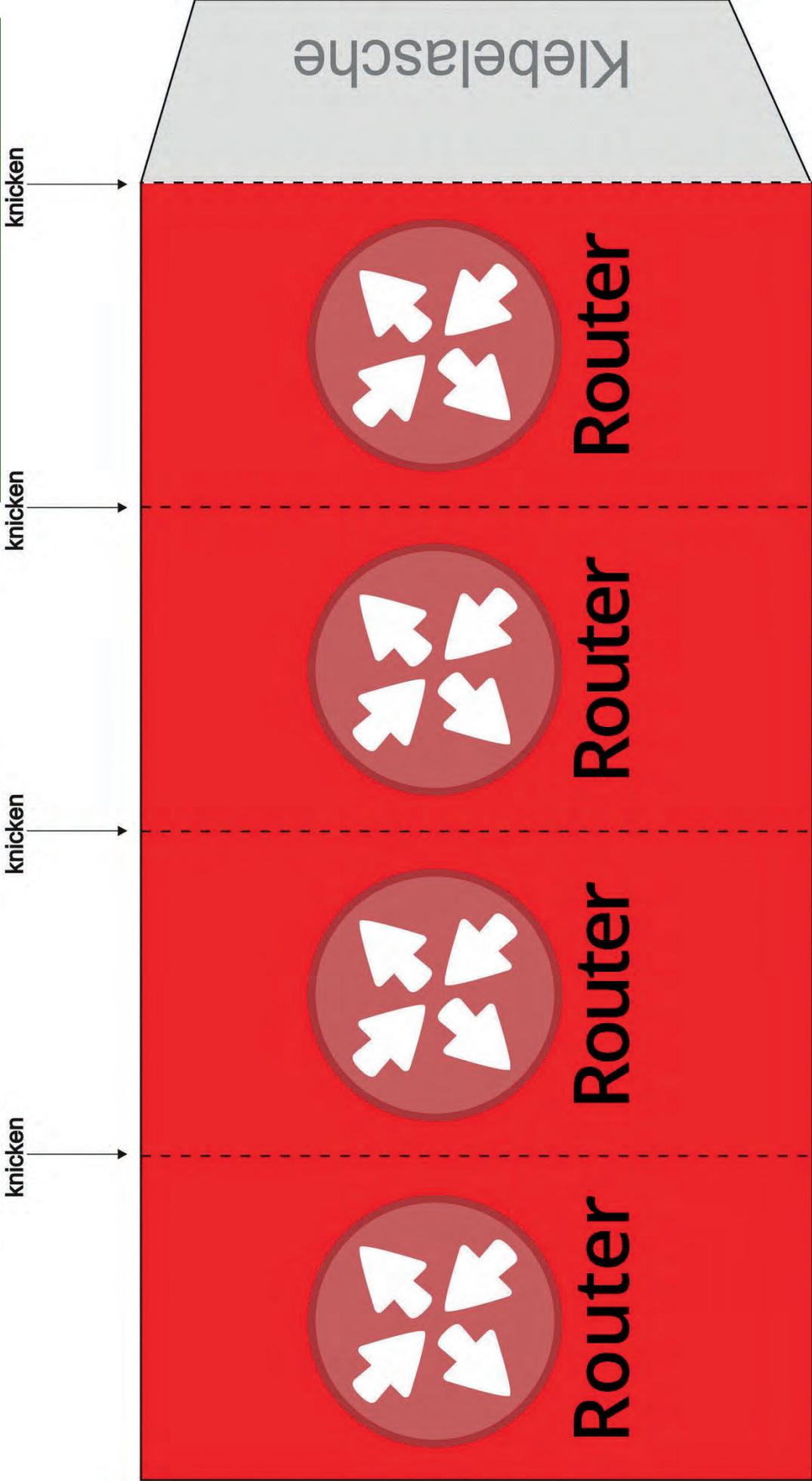
**Arbeitsmaterial B2.2.1**



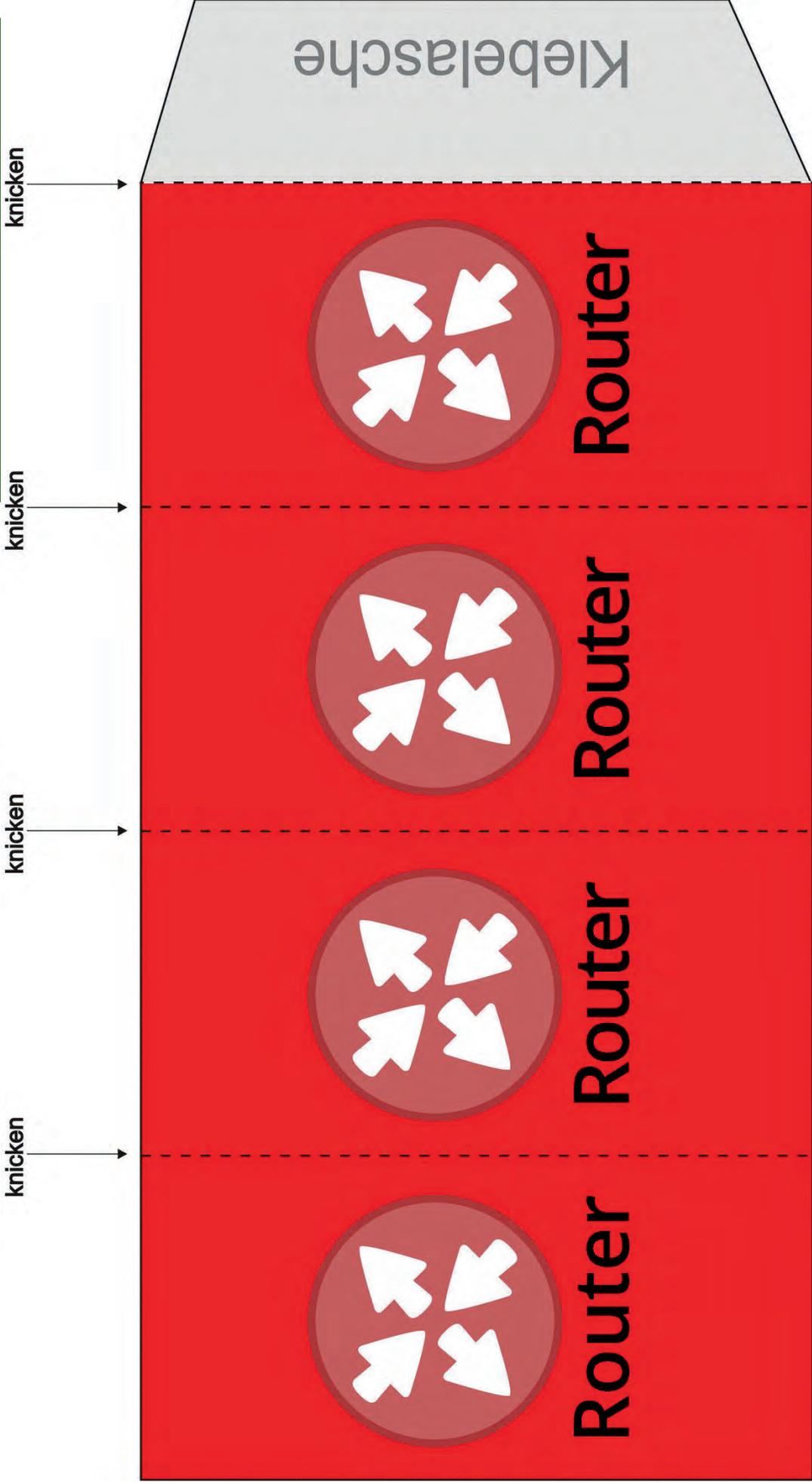
**Arbeitsmaterial B2.2.1**



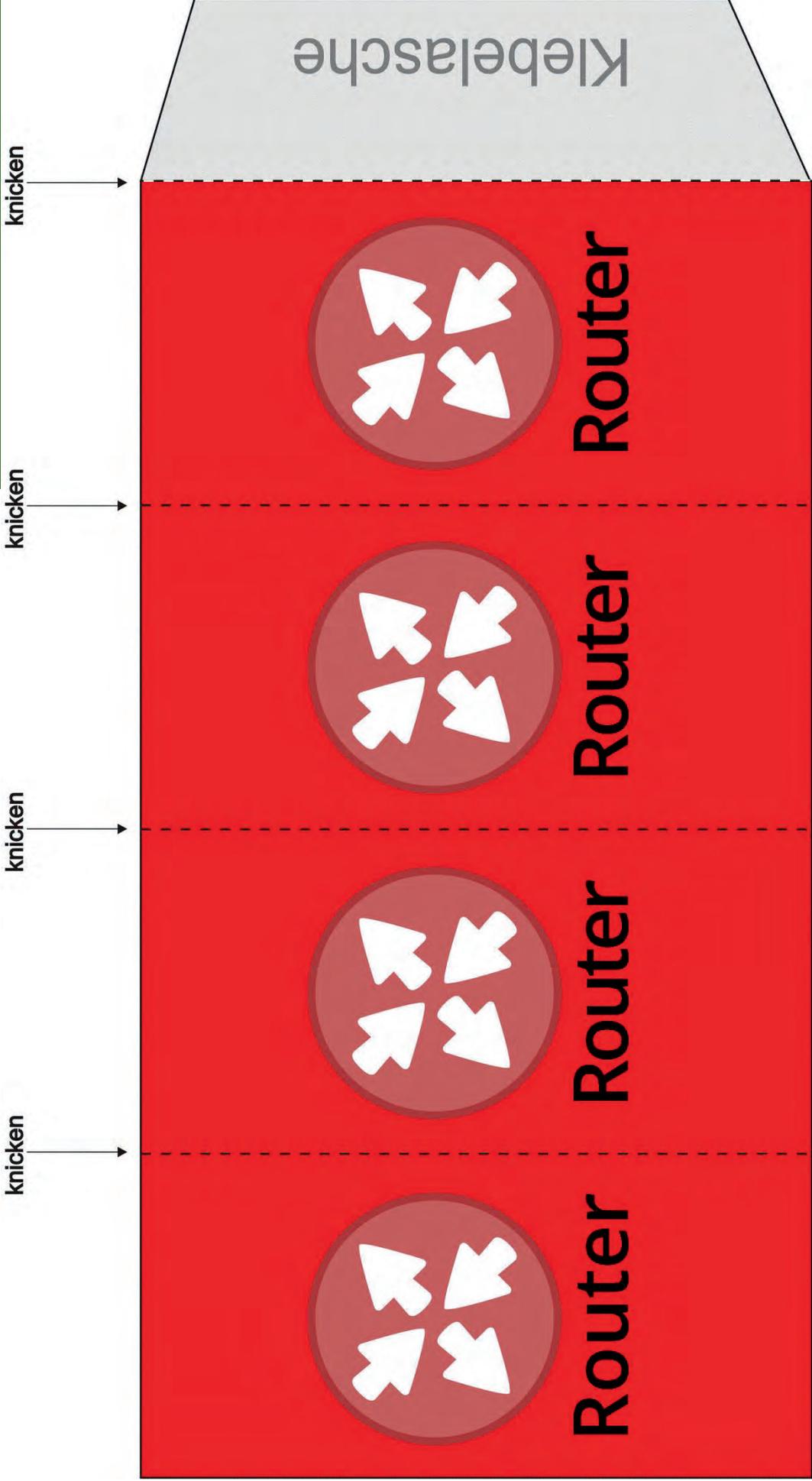
**Arbeitsmaterial B2.2.1**



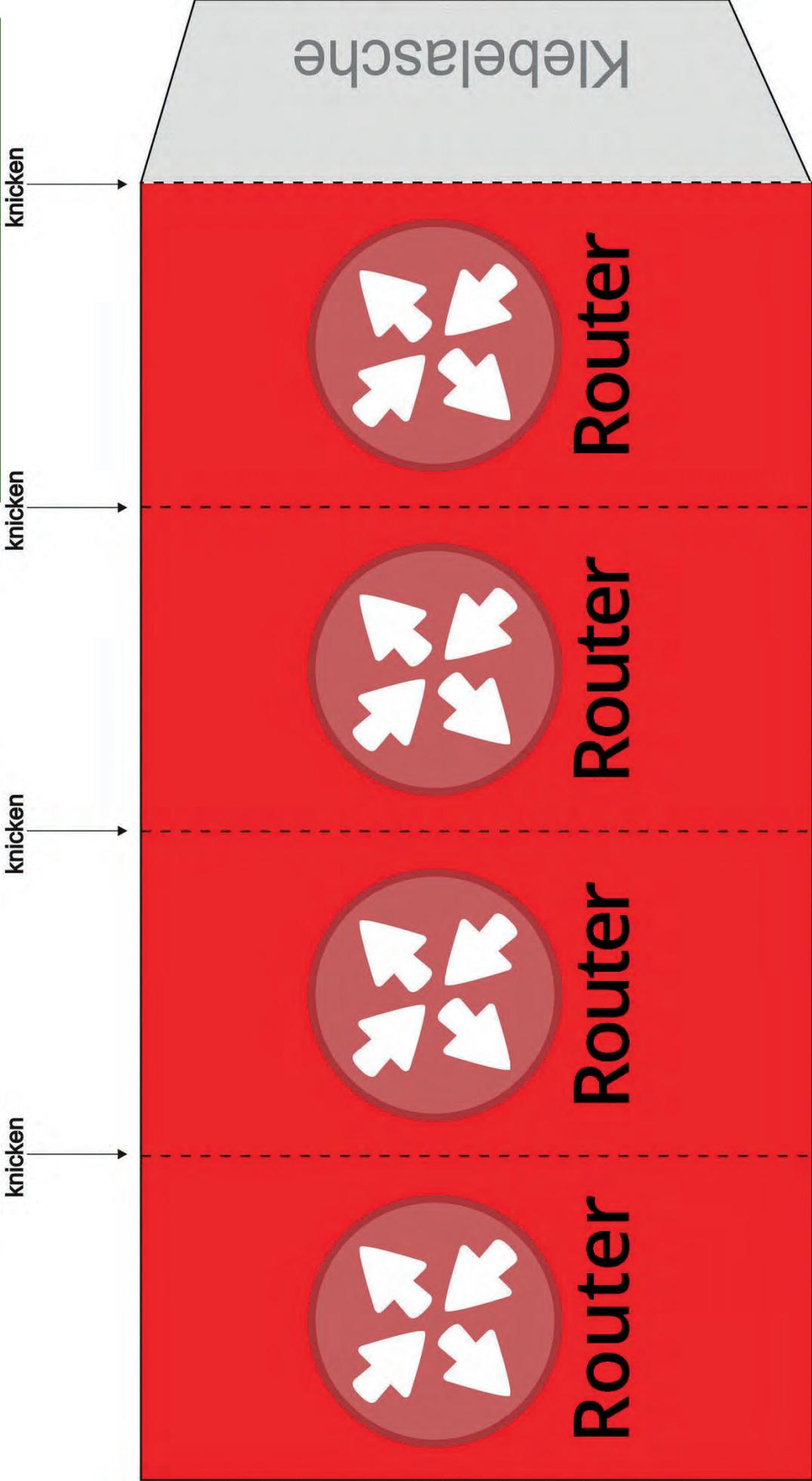
**Arbeitsmaterial B2.2.1**



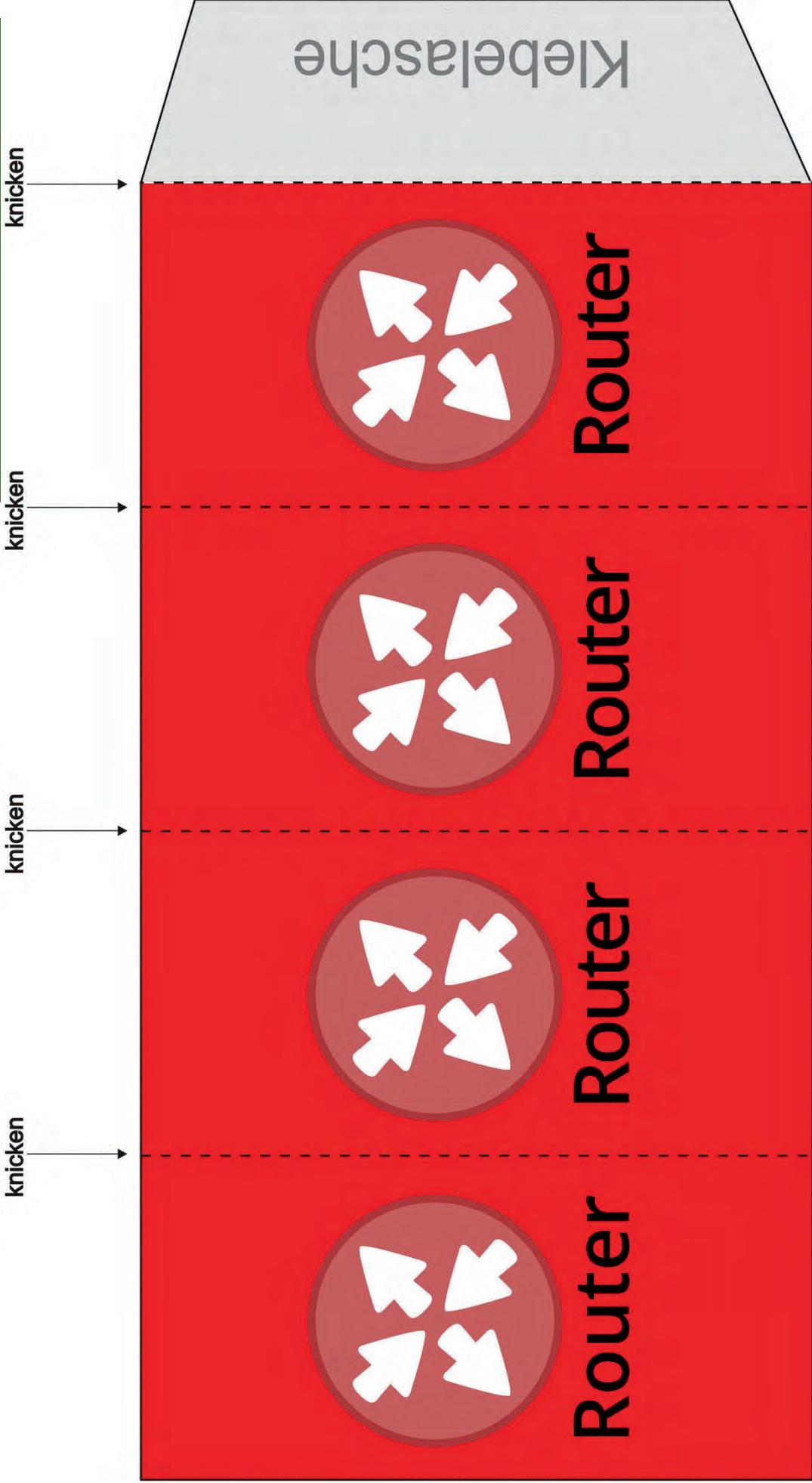
**Arbeitsmaterial B2.2.1**



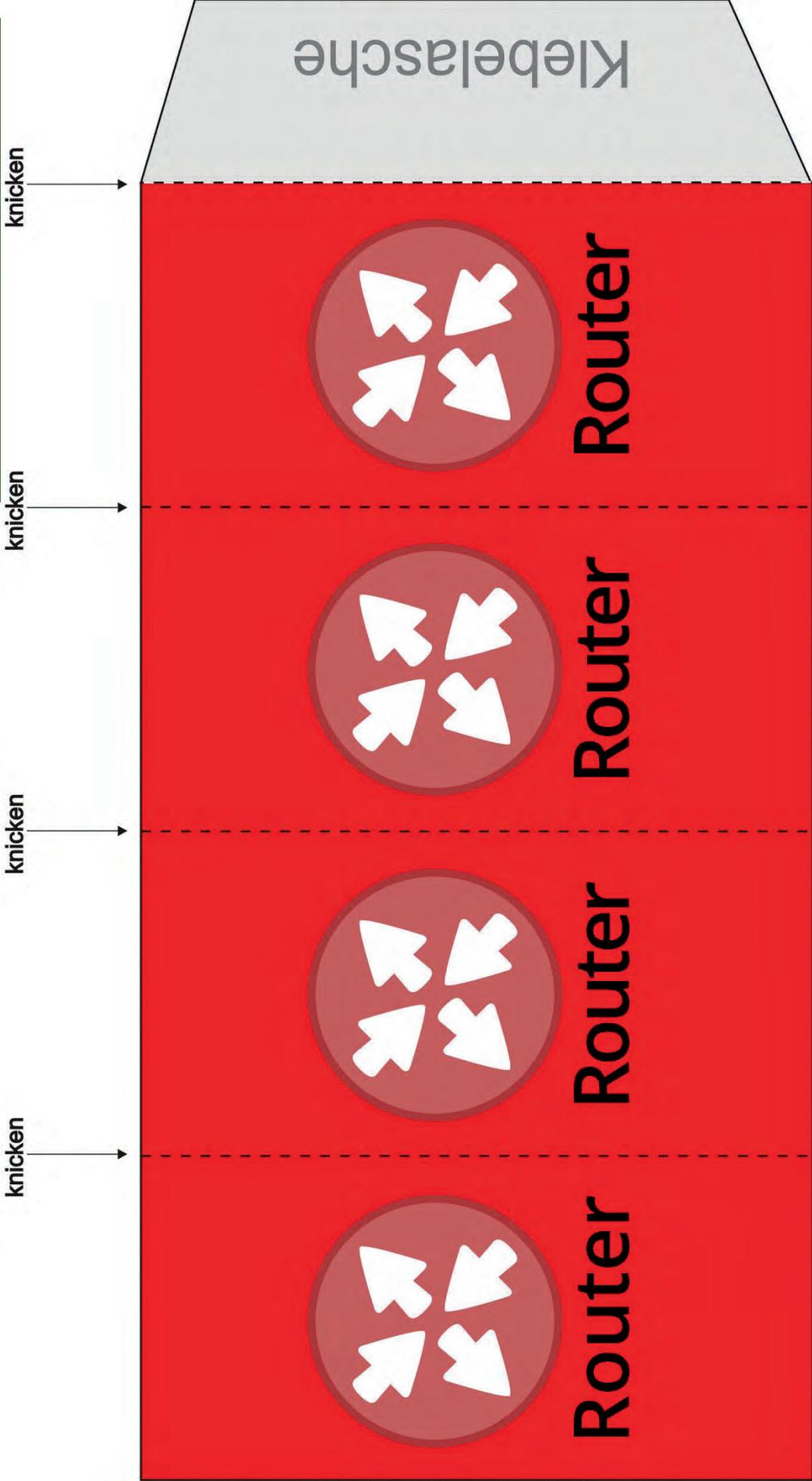
**Arbeitsmaterial B2.2.1**



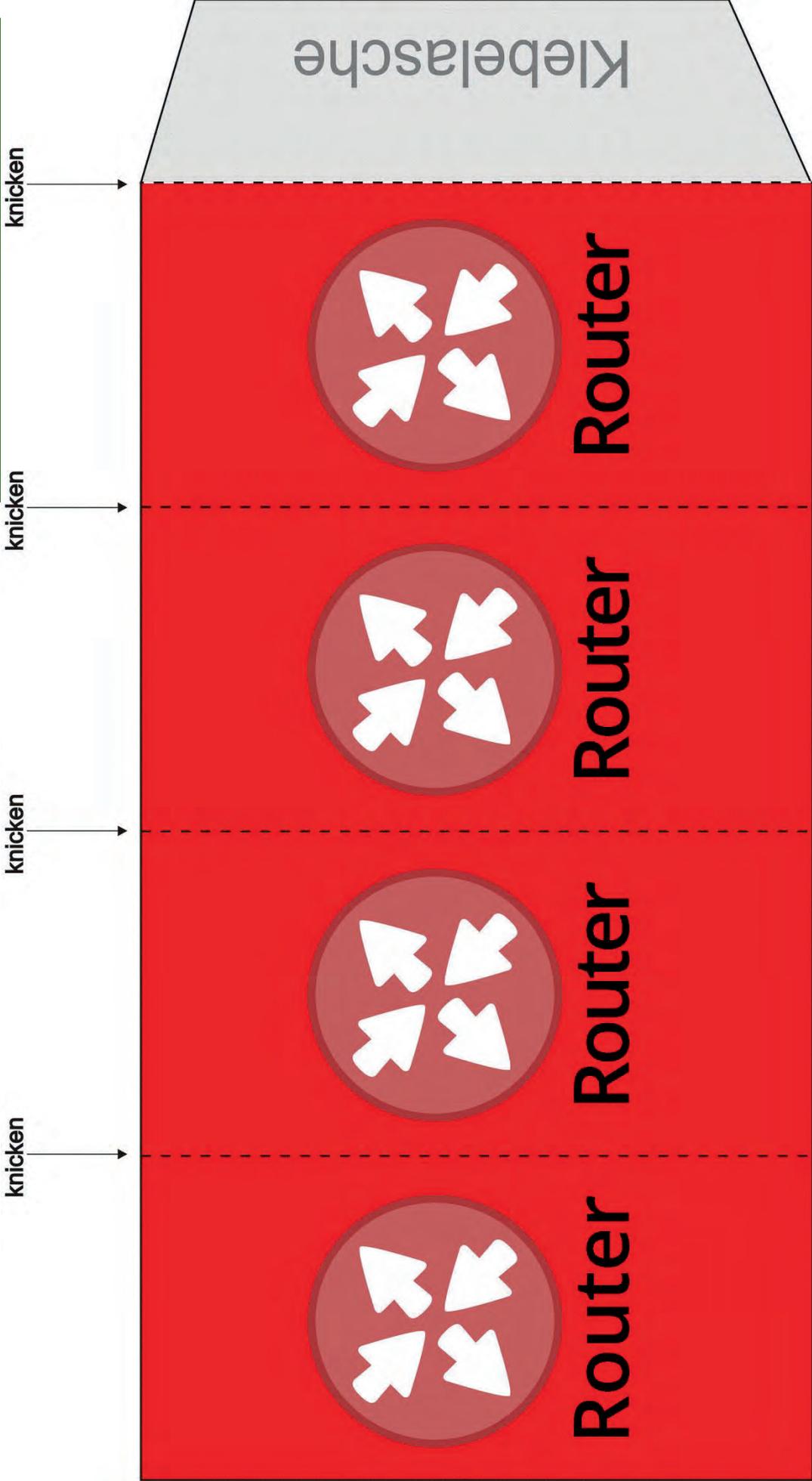
**Arbeitsmaterial B2.2.1**



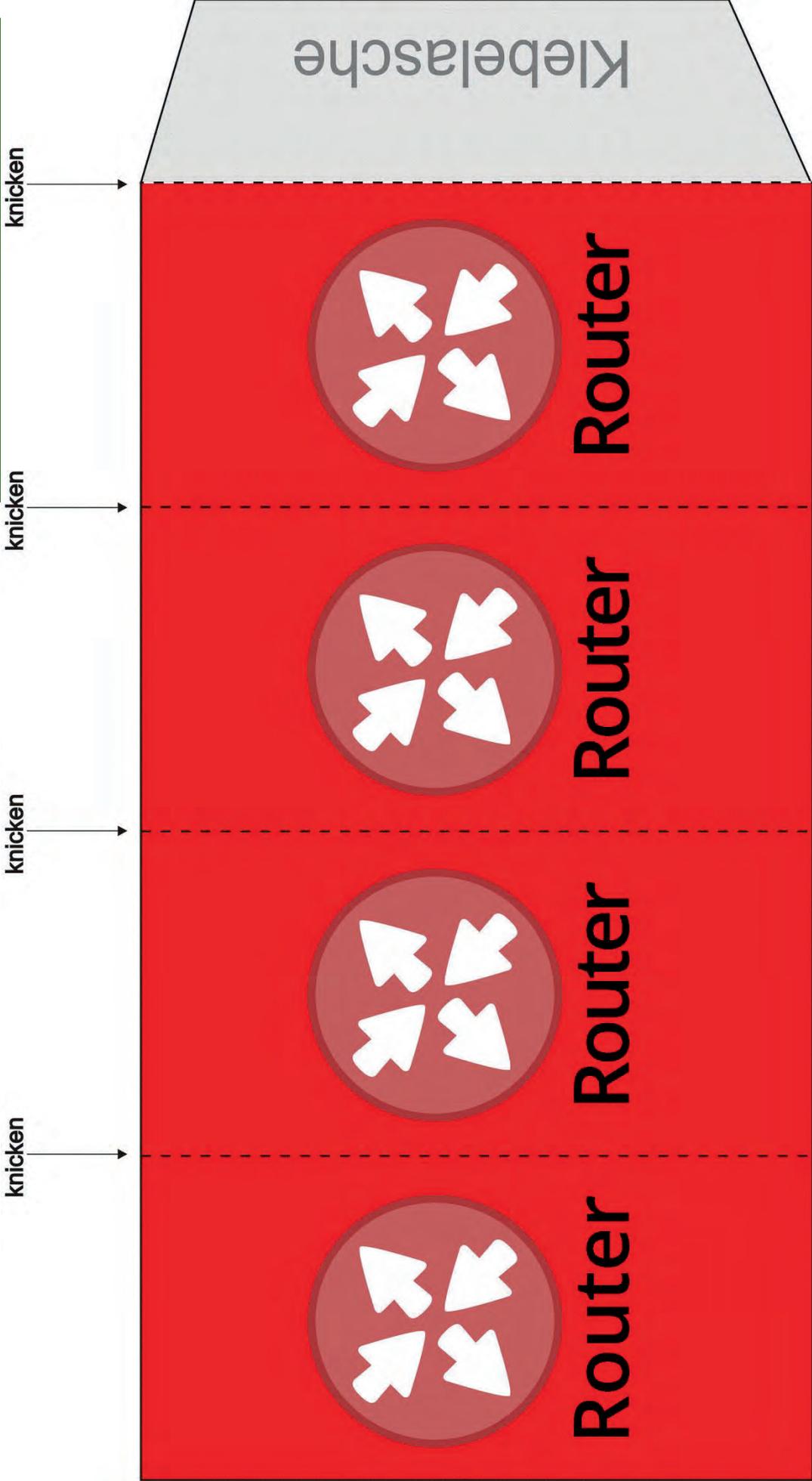
**Arbeitsmaterial B2.2.1**



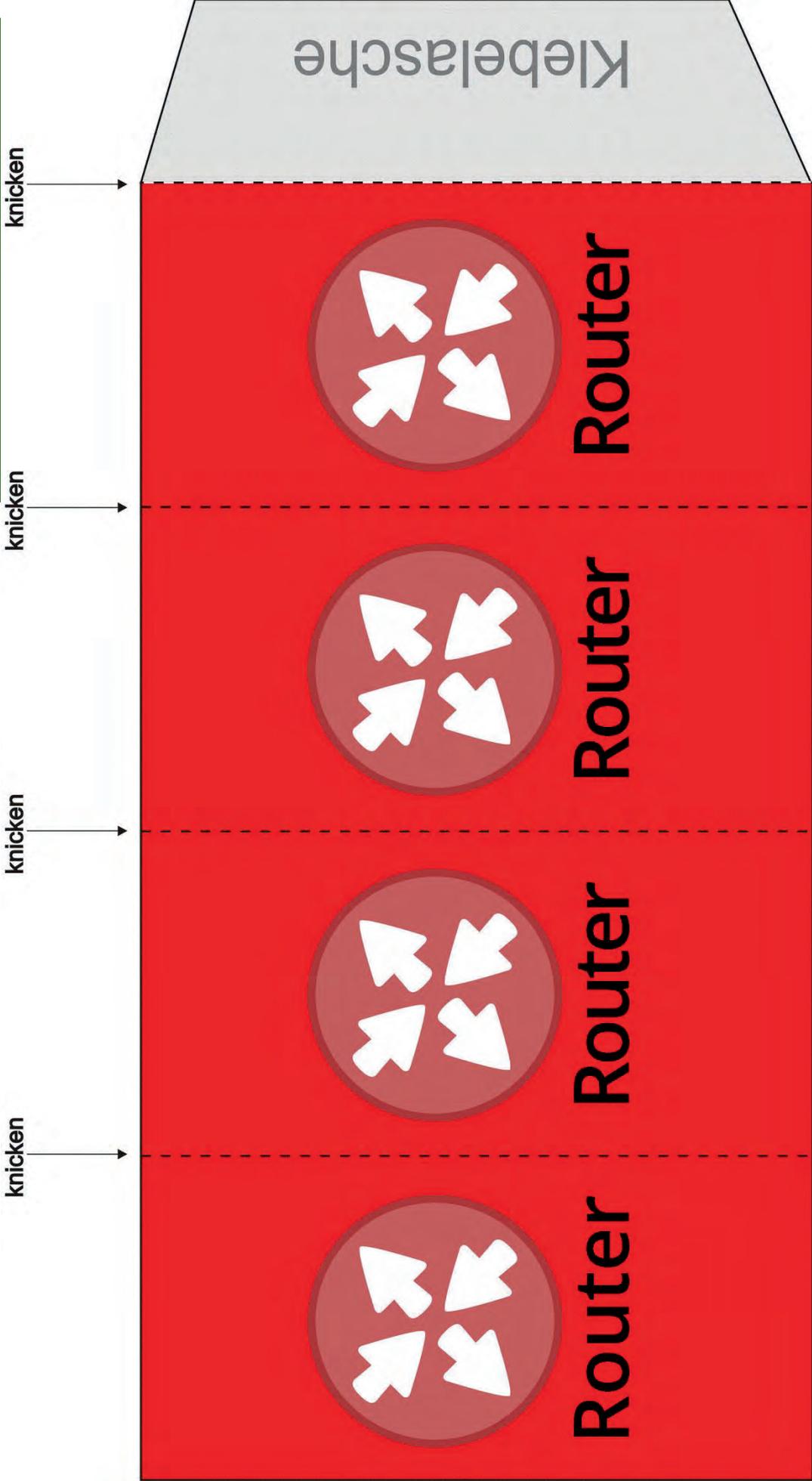
**Arbeitsmaterial B2.2.1**



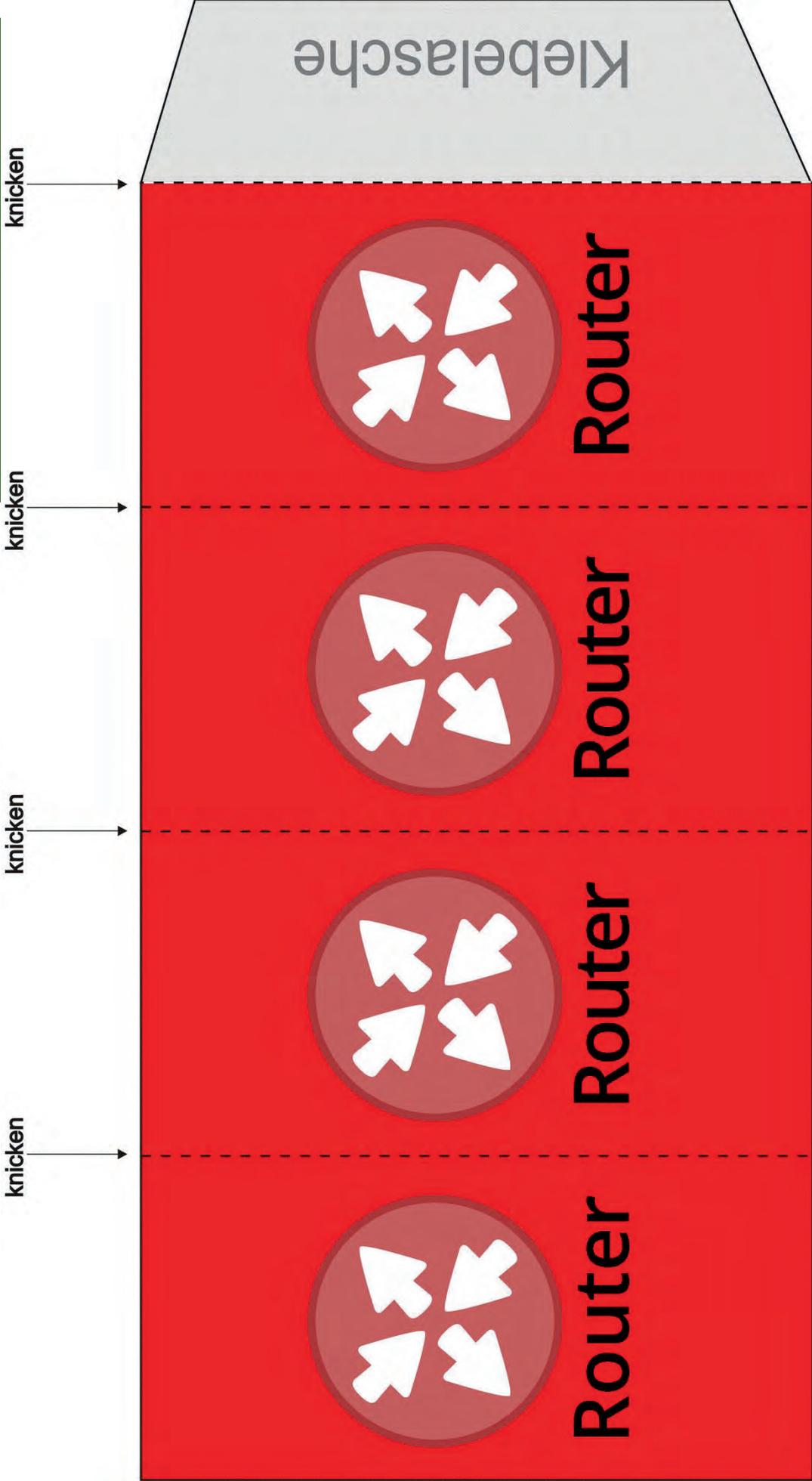
**Arbeitsmaterial B2.2.1**



**Arbeitsmaterial B2.2.1**



**Arbeitsmaterial B2.2.1**



### Webseiten

...für das Model.

Bitte die folgende Seite farbig auf festem Papier ausdrucken  
und die kleinen Abbildungen der Webseiten zurechtschneiden.

Startseite
Über uns
Neuigkeiten
Kontakt

**Verzerrungsplan**  
**Abiturthemen**  
**Mensa Menüplan**  
**Terminkalender**  
**Downloads**

## Fiktive Webseite

**Startseite**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonummy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonummy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.



Startseite
Über uns
Neuigkeiten
Kontakt

**Verzerrungsplan**  
**Abiturthemen**  
**Mensa Menüplan**  
**Terminkalender**  
**Downloads**

## Fiktive Webseite

**Startseite**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonummy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.



Startseite
Über uns
Neuigkeiten
Kontakt

**Verzerrungsplan**  
**Abiturthemen**  
**Mensa Menüplan**  
**Terminkalender**  
**Downloads**

## fragFINNde

Geheckt! Das Netz für Kids.

**Ich suche**

was ist das Problem?  
 Problem:

Frage FINN  
 Ich werde antworten

Media Smart  
 Radio, Paul Pöckel, Anne Anzeile  
 und viele Personen haben dieses  
 Video gesehen. Wie ist die  
 Verbindung? Fräulein, was ist  
 verschicktem Formate in gute. Außerdem  
 werden wir ein neues  
 Thema  
 Tausche dich mit anderen Kindern  
 (Foto: www.berlin.de)

**Los!**

Video des Tages

Spielen

Fräulein, was ist die  
 Verbindung?

Startseite
Über uns
Neuigkeiten
Kontakt

**Verzerrungsplan**  
**Abiturthemen**  
**Mensa Menüplan**  
**Terminkalender**  
**Downloads**

## fragFINNde

Geheckt! Das Netz für Kids.

**Ich suche**

was ist das Problem?  
 Problem:

Frage FINN  
 Ich werde antworten

Media Smart  
 Radio, Paul Pöckel, Anne Anzeile  
 und viele Personen haben dieses  
 Video gesehen. Wie ist die  
 Verbindung? Fräulein, was ist  
 verschicktem Formate in gute. Außerdem  
 werden wir ein neues  
 Thema  
 Tausche dich mit anderen Kindern  
 (Foto: www.berlin.de)

**Los!**

Video des Tages

Spielen

Fräulein, was ist die  
 Verbindung?



## Stationskarten

...für das Planspiel.

Bitte die folgenden Seiten farbig auf festem Papier ausdrucken  
und **nur** die Stationskarten des Routers zurechtschneiden.

# Client

# Provider

# DNS

# Heim- Internetrouter

Webserver  
188.64.58.118

Webserver  
178.254.10.171

Router

## Stationsmaterial

...für das Planspiel.

Bitte die folgenden Seiten farbig auf festem Papier ausdrucken.

Die Adressliste beim DNS hinterlegen und die Abbildungen  
der Webseiten bei den zugehörigen  
Webservern (erkennbar an der IP-Adresse).

Internetadressliste vom DNS	
Internetseite	IP-Adressen
www.spielen.de	46.252.16.49
www.news4kids.de	85.13.140.4
www.seitenstark.de	5.35.256.117
www.oldenburg.de	213.168.206.65
www.sowieso.de	85.214.76.94
www.fragfinn.de	188.64.58.118
www.kidsville.de	85.25.66.34
www.uni-oldenburg.de	134.106.87.120
www.schule.de	80.239.207.207
www.internauten.de	78.47.185.197
www.starke-pfoten.de	62.112.44.148
www.deine-schule.de	178.254.10.171
www.kinderuni-oldenburg.de	94.102.218.116

www.deine-schule.de

178.254.10.171



# DEINE SCHULE

Startseite

Über uns

Neuigkeiten

Kontakt

**Vertretungsplan**

**Abiturthemen**

**Mensa Menüplan**

**Terminkalender**

**Downloads**

## Startseite

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

**Fiktive Webseite**

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.



www.deine-schule.de

178.254.10.171



# DEINE SCHULE

Startseite

Über uns

Neuigkeiten

Kontakt

**Vertretungsplan**

**Abiturthemen**

**Mensa Menüplan**

**Terminkalender**

**Downloads**

## Startseite

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

**Fiktive Webseite**

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.



www.fragfinn.de 188.64.58.118

**fragFINN.de**  
Gecheckt! Das Netz für Kids.

**Ich suche**  **Los!**

**webTIPP** Schick FINN deine Favoriten  
**problem LÖSER** Tipps zum Suchen und Finden

**postFACH** Sag FINN deine Meinung  
**fragFINN** Als Startseite festlegen

**Surftipp des Tages**  
**Media Smart**  
 Hast du schon von **Inge Internet, Ralf Radio, Paul Plakat, Anne Anzeige** und **Felix Fernsehen** gehört? Diese Fünf erklären dir bei [mediasmart.de](http://mediasmart.de) alles Wichtige rund ums Thema Werbung. Finde heraus, wie du Werbung erkennst und weiche **verschiedenen Formen** es gibt. Außerdem wartet wieder ein neues **Thema des Monats** auf dich, zu dem deine Meinung gefragt ist. **Tausche dich mit anderen Kindern** dazu.  
 << Archiv >> zum **Tagesstipp**

**Video des Tages**  
 Sorry, your computer's source and playback technology were found for this video. Try using another browser to download the video.  
 >> **mehr Videos**

**Spielen**  
 FINNs neueste Spieletipps für dich!

**Stöberecke**  
 Mobbing: Nicht mit uns!

**eltern INFO**  
 Der Weg zum sicheren Surfraum

www.fragfinn.de 188.64.58.118

**fragFINN.de**  
Gecheckt! Das Netz für Kids.

**Ich suche**  **Los!**

**webTIPP** Schick FINN deine Favoriten  
**problemLÖSER** Tipps zum Suchen und Finden

**postFACH** Sag FINN deine Meinung  
**fragFINN** Als Startseite festlegen

**Surftipp des Tages**  
**Media Smart**  
 Hast du schon von **Inge Internet, Ralf Radio, Paul Plakat, Anne Anzeige** und **Felix Fernsehen** gehört? Diese Fünf erklären dir bei mediasmart.de alles Wichtige rund ums Thema Werbung. Finde heraus, wie du Werbung erkennst und welche **verschiedenen Formen** es gibt. Außerdem wartet wieder ein neues **Thema des Monats** auf dich, zu dem deine Meinung gefragt ist. **Tausche dich mit anderen Kindern** dazu.  
 << Archiv >> zum Tagestipp

**Video des Tages**  
 Sorry, your computer's source and playback technology were found for this video. Try using another browser to download the video.  
 >> mehr Videos

**Stöberecke**  
 Mobbing: Nicht mit uns!

**Spielen**  
 FINNs neueste Spielertipps für dich!

**eltern INFO**  
 Der Weg zum sicheren Surfraum

# Protokollheft

...für das Planspiel.

Bitte die folgenden Seiten farbig auf festem DIN-A5 Papier ausdrucken oder zwei Seiten auf einer DIN-A4 Seite und dann entsprechend zurechtschneiden.

Anschließend laminieren, lochen und binden (Beispiel siehe Abbildung).





# Protokollheft

Eine Entwicklung von OFFIS e.V. in Kooperation mit der Universität Oldenburg  
im Auftrag der Wissensfabrik – Unternehmen für Deutschland e.V.



**Von:**  
Client

**An:**  
Heim-Internetrouter

**Nachricht:**

Hallo! Ich möchte mich im lokalen Netzwerk anmelden, damit ich ins Internet kann. Bitte melde mich wenn nötig auch beim Provider an.

1

**Von:**  
Heim-Internetrouter

**An:**  
Provider

**Nachricht:**

Ich bin

\_\_\_\_\_ .  
(Name eintragen)

Mein Passwort lautet

\_\_\_\_\_ .  
(Passwort eintragen)

Ich möchte eine Verbindung zum  
Internet.

2

**Von:**  
Provider

**An:**  
Heim-Internetrouter

**Nachricht:**

OK, du bist Kunde bei mir!

Deine öffentliche IP-Adresse lautet:

134.106.27.170

Die IP-Adresse des DNS lautet:

8.8.8.8

3

**Von:**  
Heim-Internetrouter

**An:**  
Client

**Nachricht:**

OK, du bist im lokalen Netzwerk angemeldet und kannst nun ins Internet.

Deine lokale IP-Adresse lautet:

192.168.1.4

Die IP-Adresse des DNS lautet:

8.8.8.8

4

**Von:**  
Client

**Über:**  
Heim-Internetrouter

**Über:**  
Provider

**An:**  
DNS

**Nachricht:**

Wie lautet die IP-Adresse vom  
Webserver mit der Internetseite:

`www.deine-schule.de`

5

**Von:**  
DNS

**Über:**  
Provider

**Über:**  
Heim-Internetrouter

**An:**  
Client

**Nachricht:**

Die IP-Adresse des Webserver mit  
der Internetseite `www.deine-  
schule.de` lautet:

---

(IP-Adresse eintragen)

6

**Von:**  
Client

**Über:**  
Heim-Internetrouter

**Über:**  
Provider

**An:**  
Webserver 178.254.10.171

**Nachricht:**

Bitte gib mir eine Kopie der  
Internetseite:

[www.deine-schule.de](http://www.deine-schule.de)

7

**Von:**

Webserver 178.254.10.171

**Über:**

Provider

**Über:**

Heim-Internetrouter

**An:**

Client

**Nachricht:**

Bitte schön, hier ist der Inhalt der  
Internetseite:

[www.deine-schule.de](http://www.deine-schule.de)

(Internetseite bitte mitgeben!)

8

**Von:**  
Client

**Über:**  
Heim-Internetrouter

**Über:**  
Provider

**An:**  
DNS

**Nachricht:**

Wie lautet die IP-Adresse vom  
Webserver mit der Internetseite:

[www.fragfinn.de](http://www.fragfinn.de)

9

**Von:**  
DNS

**Über:**  
Provider

**Über:**  
Heim-Internetrouter

**An:**  
Client

**Nachricht:**

Die IP-Adresse des Webserver mit  
der Internetseite [www.fragfinn.de](http://www.fragfinn.de)  
lautet:

---

(IP-Adresse eintragen)

10

**Von:**  
Client

**Über:**  
Heim-Internetrouter

**Über:**  
Provider

**An:**  
Webserver 188.64.58.118

**Nachricht:**

Bitte gib mir eine Kopie der  
Internetseite:

[www.fragfinn.de](http://www.fragfinn.de)

11

**Von:**  
Webserver 188.64.58.118

**Über:**  
Provider

**Über:**  
Heim-Internetrouter

**An:**  
Client

**Nachricht:**

Bitte schön, hier ist der Inhalt der  
Internetseite:

[www.fragfinn.de](http://www.fragfinn.de)

(Internetseite bitte mitgeben!)

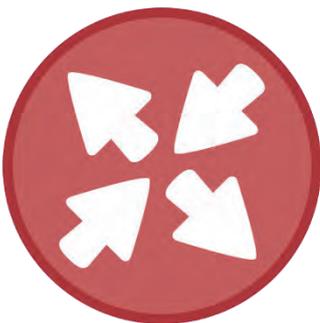
12

## Gruppenkarten

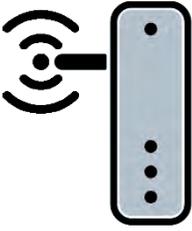
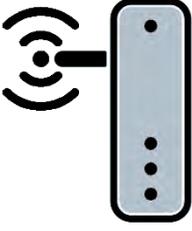
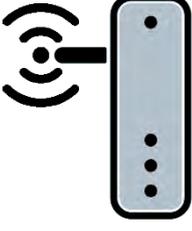
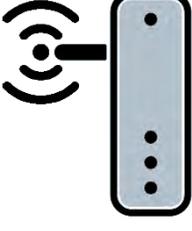
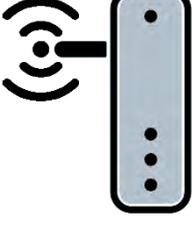
...für das Planspiel.

Bitte die folgenden Seiten farbig auf festem Papier  
ausdrucken und zurechtschneiden.

Arbeitsmaterial B2.2.6

 <b>Router</b>	 <b>Router</b>	 <b>Router</b>	 <b>Router</b>	 <b>Router</b>
 <b>Provider</b>	 <b>Provider</b>	 <b>Provider</b>	 <b>Provider</b>	 <b>Provider</b>

## Arbeitsmaterial B2.2.6

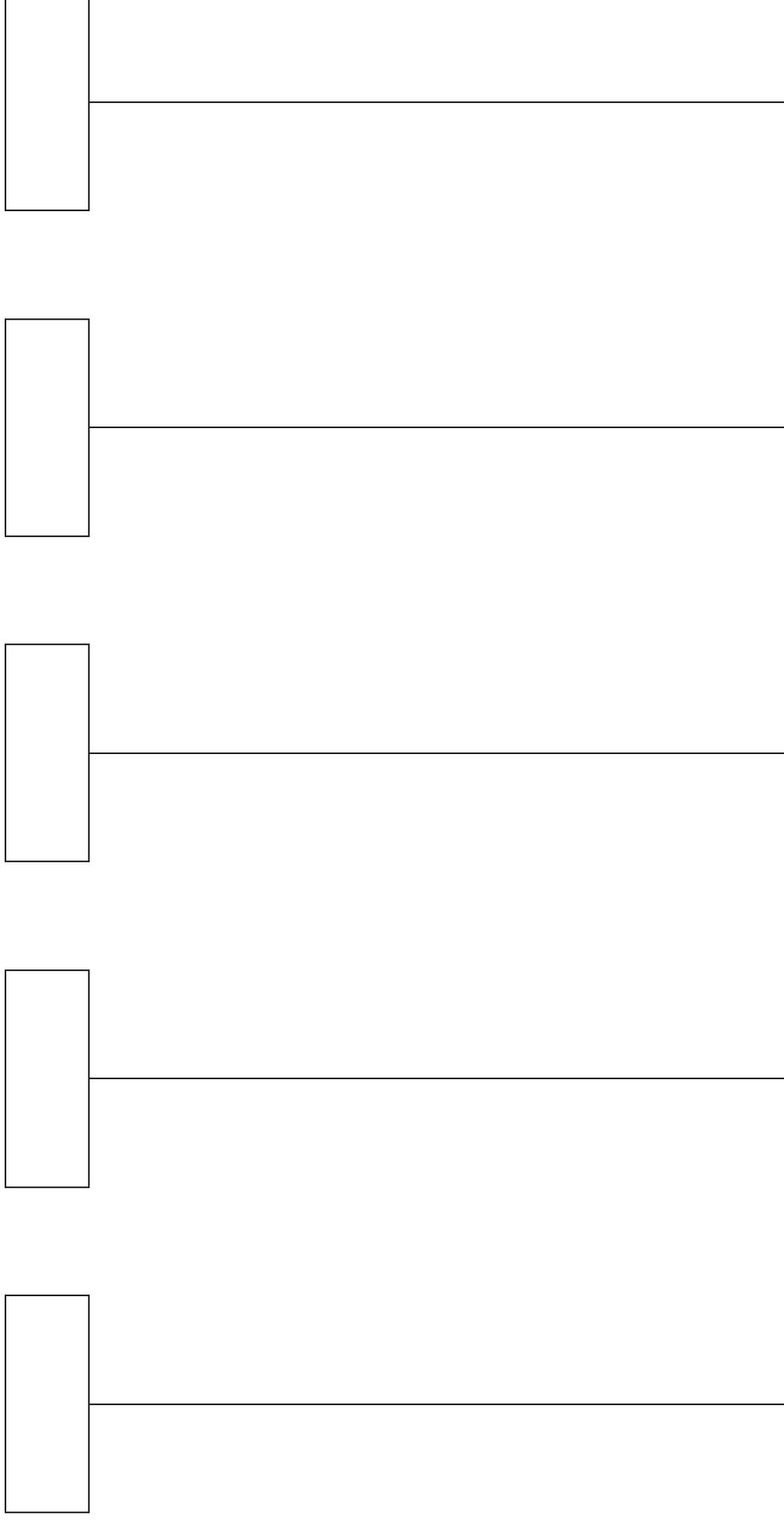
 <p><b>DNS</b></p>	 <p><b>DNS</b></p>	 <p><b>DNS</b></p>	 <p><b>DNS</b></p>	 <p><b>DNS</b></p>
 <p>Heim- Internet- router</p>	 <p>Heim- Internet- router</p>	 <p>Heim- Internet- router</p>	 <p>Heim- Internet- router</p>	 <p>Heim- Internet- router</p>

## Arbeitsmaterial B2.2.6

 <b>Web- server</b>	 <b>Web- server</b>	 <b>Web- server</b>	 <b>Web- server</b>	 <b>Web- server</b>
 <b>Client</b>	 <b>Client</b>	 <b>Client</b>	 <b>Client</b>	 <b>Client</b>

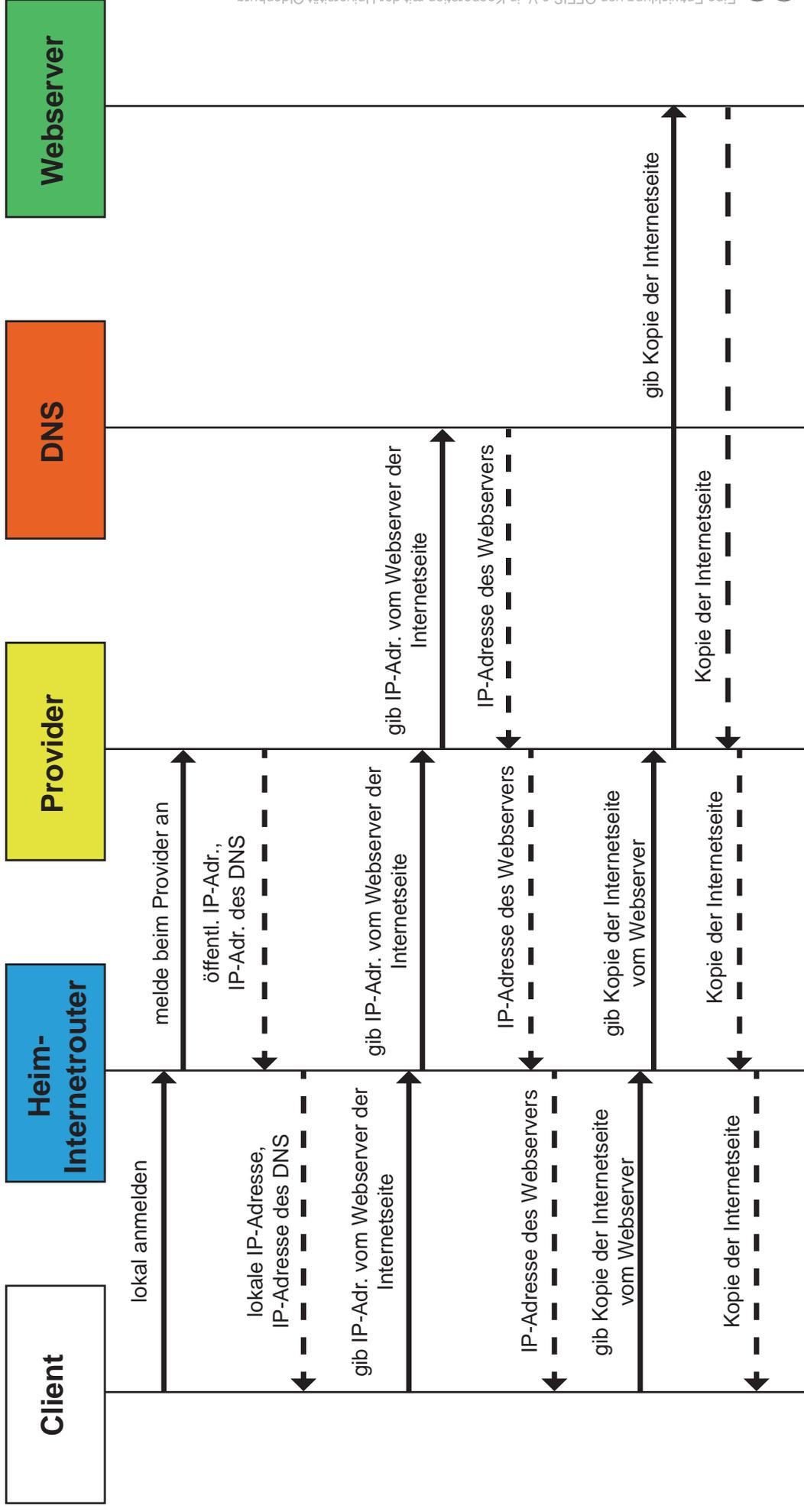
## Arbeitsmaterial B2.3

### Arbeitsblatt: Sequenzdiagramm – Abruf einer Internetseite



# Musterlösung B2

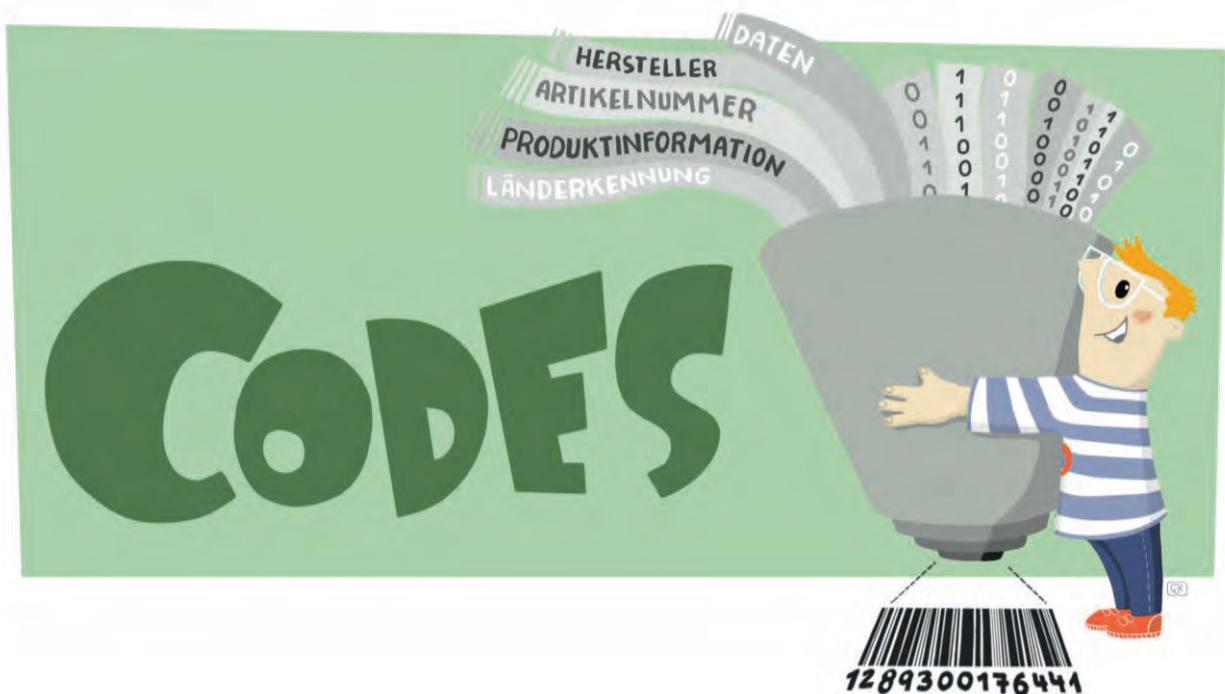
## Musterlösungen



© CC BY-NC-SA  
 Eine Entwicklung von OFFIS e.V. in Kooperation mit der Universität Oldenburg  
 im Auftrag der Wissensfabrik – Unternehmen für Deutschland e.V.

# IT2School

Gemeinsam IT entdecken



## Modul B3 – Codes

Codes im Supermarkt und Unternehmen

Eine Entwicklung von



In Kooperation mit



Im Auftrag der

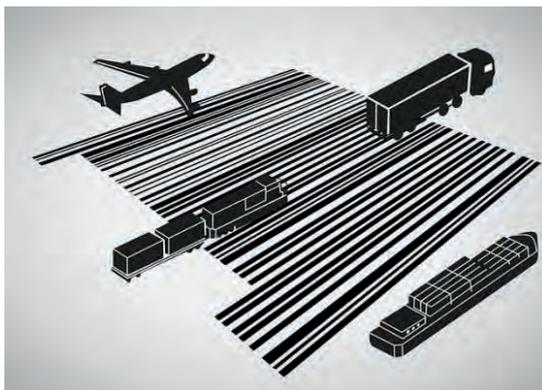


# Inhalt

1	Codes im Supermarkt und Unternehmen .....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Rolle der Unternehmensvertreter*innen.....	4
5	Inhalte des Moduls.....	4
5.1	Wofür benötigt man Barcodes? .....	7
5.2	Der QR-Code .....	7
6	Unterrichtliche Umsetzung.....	8
6.1	Grober Unterrichtsplan.....	8
6.1.1	Variante 1 .....	9
6.1.2	Variante 2 .....	9
6.2	Stundenverlaufsskizzen .....	10
6.2.1	Variante 1 .....	10
6.2.2	Variante 2 .....	13
7	Einbettung in verschiedene Fächer und Themen .....	17
8	Anschlussthemen.....	18
9	Literatur und Links .....	18
10	Arbeitsmaterialien .....	18
11	Glossar.....	19
12	FAQs und Feedback.....	20

# 1 Codes im Supermarkt und Unternehmen

Optische Codes wie Barcodes oder QR-Codes begegnen uns mittlerweile überall. Egal ob auf den Produkten im Supermarkt, Tickets (Veranstaltungen, Bahn-/Flugzeugtickets) oder in der Werbung. Aber wie funktionieren eigentlich solche Codes und welche Informationen beinhalten sie?



Dieses Modul befasst sich mit der Funktionsweise und den Einsatzmöglichkeiten von optischen Codes. Die Schülerinnen und Schüler erfahren, wo sie überall zu finden sind, wie man sie erstellt, welche Informationen sie bereitstellen und welche Bedeutung sie für die Wirtschaft haben. Für diesen Zweck kann auch eine Exkursion zu einem Unternehmen oder einem ortsansässigen Supermarkt unternommen werden.

<b>Lernfeld/Cluster:</b>	Daten erforschen	
<b>Zielgruppe/Klassenstufe:</b>		4. bis 5. Klasse
	X	6. bis 7. Klasse
	X	8. bis 10. Klasse
	X	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	Ca. 5 bis 8 Doppelstunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• computergestützte Codierungs- und Decodierungssysteme kennenlernen</li> <li>• Funktionsweise von Codes (EAN-Codes) kennenlernen</li> <li>• IT im Alltag entdecken</li> <li>• versteckte Informationen im Supermarkt mithilfe von Barcodes erforschen</li> <li>• QR-Codes selbst für eigene Zwecke (Rallye) erstellen</li> </ul>	
<b>Vorkenntnisse der Schüler*innen:</b>	Keine	
<b>Vorkenntnisse der Lehrkraft:</b>	Keine	
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Keine	
<b>Sonstige Voraussetzungen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Internet- und WLAN-Zugang (für die Installation und Verwendung von Apps)</li> <li>• Mehrere Schüler*innen-Smartphones</li> </ul>	



## 2 Warum gibt es das Modul?

Sowohl Barcodes als auch QR-Codes begegnen uns überall im Alltag. Im Supermarkt ist der auch „Strichcode“ genannte Barcode auf nahezu jedem Produkt zu finden. Mussten früher Kassiererinnen und Kassierer die Preise manuell in die Kasse eintippen, so können heute die meisten Artikel in Sekundenschnelle eingescannt werden. Auch die Lagerung sowie die Logistik wurden mithilfe von Barcodes wesentlich einfacher. Die Einführung dieses Codes hat vor 40 Jahren die Handelsbranche revolutioniert.

In diesem Modul erforschen und entdecken die Schülerinnen und Schüler, wie Barcodes und QR-Codes funktionieren, welche Informationen mit ihrer Hilfe gespeichert werden und welche Bedeutung ihre Entwicklung für den Handel und die Wirtschaft hat.

## 3 Ziele des Moduls

- computergestützte Codierungs- und Decodierungssysteme kennenlernen
- Funktionsweise von Codes (EAN-Codes) kennenlernen
- Informationstechnologien (IT) im Alltag entdecken
- versteckte Informationen im Supermarkt mithilfe von Barcodes erforschen
- verschiedene Codes selbst erzeugen
- die Bedeutung von Codes für die Industrie und Wirtschaft verstehen

## 4 Rolle der Unternehmensvertreter\*innen

Im *Modul B3 – Codes* hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten aktiv mitzuwirken. Hier einige Anregungen:

- Sie oder er kann als Special-Guest eingeladen werden, um über die Bedeutung von Codes in der Wirtschaft und insbesondere im eigenen Unternehmen zu berichten.
- Sie oder er kann den Schülerinnen und Schülern eine Exkursion in das eigene Unternehmen ermöglichen und zeigen, wie Codes eingesetzt werden.

## 5 Inhalte des Moduls

Es gibt viele verschiedene Typen von Strichcodes. Die **GTIN-13** (Global Trade Item Number) oder, wie bis 2009 genannt, Europäische Artikelnummer (EAN), findet sich auf einer Vielzahl von Artikeln und Verpackungen im Einzelhandel. Sind Verpackungen allerdings zu klein für einen 13-stelligen Code, besteht auch die Möglichkeit, den Barcode in kleinerer Form als sogenannten **GTIN-8** zu drucken. Sowohl die GTIN-13 als auch die GTIN-8-Codes können von Barcode-Scannern, wie sie an Supermarktkassen zur Verfügung stehen, ausgelesen werden. Der Code



8-stelliger-Code

besteht aus breiten und schmalen schwarzen Strichen sowie Lücken dazwischen. Das Wort „Code“ lässt an eine geheime Verschlüsselung denken, gemeint ist hier aber die Abbildung von Daten im binären Code.

Durch das Scannen von Barcodes können alle wichtigen Informationen eines Produkts, wie die Herkunft oder der Preis, abgelesen werden. Der GTIN-13-Code besitzt weltweite Gültigkeit und ist für jedes Produkt einmalig und wie folgt aufgebaut:



Die ersten sieben Ziffern entsprechen der GS1 Basisnummer, welche sich aus der Länderkennung und der Hersteller-Nr. zusammensetzt. Die GS1 (Global Standards One) ist dabei die Organisation, die für die weltweite Vergabe der GTIN verantwortlich ist. Je nachdem wo ein Unternehmen oder eine Organisation die GS1 Basisnummer lizenziert, unterscheiden sich die ersten zwei oder drei Ziffern. So vergibt die GS1 in Deutschland die Länderkennungen 400 bis 440. Das bedeutet aber nicht, dass das Produkt mit diesem Code in Deutschland produziert wird. Es bedeutet nämlich nur, dass die Basisnummer von diesem GS1-Standort erworben wurde.

Danach folgt die Hersteller- bzw. Betriebsnummer, welche ebenfalls durch die GS1 vergeben wird und jedes Unternehmen eindeutig identifiziert. Es folgt die individuelle Artikelnummer für das entsprechende Produkt. Am Ende gibt es noch eine Prüfziffer, die es ermöglicht, Fehler beim Einlesen der Codes zu erkennen.

Die Prüfziffer kann selbst berechnet werden: Alle Ziffern werden in Leserichtung abwechselnd mit 1 und 3 multipliziert und dann aufsummiert:  $1 \times \text{Ziffer } 1 + 3 \times \text{Ziffer } 2 + 1 \times \text{Ziffer } 3 + 3 \times \text{Ziffer } 4 + \dots$ . Die Prüfziffer ist dann die Zahl, die zu einer vollen Zehnerzahl (10, 20, 30, 40, ...) fehlt.

Die **GTIN-8** besteht aus lediglich acht Zeichen, die sich auch aus einer Länderkennung, einer Artikelnummer sowie einer Prüfziffer zusammensetzen; die Betriebsnummer wird bei diesem Code weggelassen.

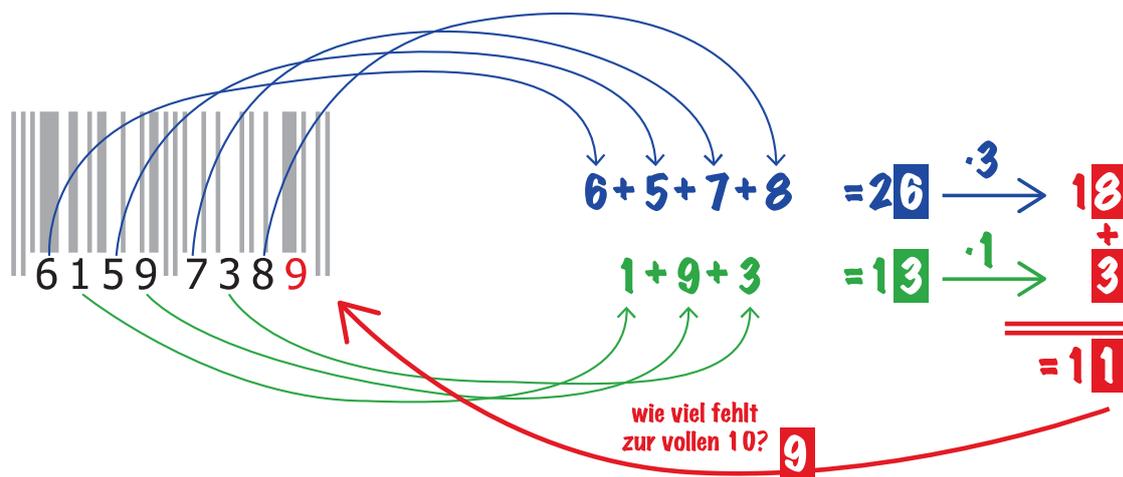
Auch bei den GTIN-8-Codes kann man die Prüfziffer berechnen. Dies funktioniert genauso wie oben für die GTIN-13-Codes beschrieben. Allerdings beginnt man in diesem Fall die Multiplikation mit 3 und wechselt dann mit der 1. Im Anschluss wird wieder aufsummiert und die Zahl, die zur vollen Zehnerzahl fehlt, ist die Prüfziffer.

**Beispiel:**  $3 \times \text{Ziffer } 1 + 1 \times \text{Ziffer } 2 + 3 \times \text{Ziffer } 3 + 1 \times \text{Ziffer } 4 + 3 \times \text{Ziffer } 5 + 1 \times \text{Ziffer } 6 + 3 \times \text{Ziffer } 7$ .

Um die Berechnung zu vereinfachen, kann das Distributivgesetz verwendet werden. So können die Ziffern an den ungeraden bzw. geraden Stellen erst addiert und im nächsten Schritt mit 1 bzw. 3 multipliziert werden.

Um den Rechenaufwand noch weiter zu reduzieren, können die Rechenregeln der Kongruenz angewendet werden.

Dies verdeutlicht die folgende Abbildung:



Da  $26 \cdot 3 \bmod 10$  kongruent zu  $6 \cdot 3 \bmod 10$  ist, können auch die Zwischenergebnisse der Addition, also  $6 + 5 + 7 + 8$ , als auch die Multiplikation, also  $26 \cdot 3$ , im Vorfeld Modulo 10 gerechnet werden, um die Berechnungen möglichst einfach zu halten. Im letzten Schritt wird bestimmt wie viele Einer zum nächsten vollen Zehner fehlen und die Berechnung der Prüfziffer ist abgeschlossen.

Häufig stellen sich im Anschluss die Fragen, warum ausgerechnet Modulo 10 und wieso die Gewichtung 1 und 3 zur Berechnung der Prüfziffer berechnet wird.

Die Antwort auf die Frage, weshalb Modulo 10 gerechnet wird, steckt bereits im Begriff **Prüfziffer** selbst. Die Berechnung soll am Ende nur eine einzelne Zahl ausgeben und kann durch die Division mit Rest mit der Zahl 10 erreicht werden.

Prüfzifferberechnungsalgorithmen sind so entworfen, dass diese selbstständig Fehler bei der manuellen Eingabe von Codes oder bei der automatischen Datenerfassung durch beispielsweise Barcodescannern erkennen können. In beiden Fällen wird die eingegebene Prüfziffer mit der nach dem obigen Verfahren berechneten Prüfziffer verglichen. Sollten beide Ziffern übereinstimmen, so würde der Code mit einer hohen Wahrscheinlichkeit richtig erkannt.

Typische Fehler beim manuellen Eintippen sind beispielsweise Tippfehler (z.B. 1 statt 2 eingegeben) oder Vertauschungen (12 statt 21). Der Tippfehler (1 statt 2) kann auch von Algorithmen ohne Gewichtung der einzelnen Stellen erkannt werden, indem einfach alle Ziffern aufaddiert und Modulo 10 gerechnet werden. Jedoch ist dieses Verfahren nicht in der Lage Vertauschungen (12 statt 21) zu erkennen, da  $1+2 \bmod 10$  auch  $2+1 \bmod 10$  entspricht.

Wählt man aber eine Gewichtung der einzelnen Stellen, z.B. 1 und 3, so können diese Vertauschungen meistens erkannt werden, da  $(1 \cdot 1 + 2 \cdot 3) \bmod 10$  nicht  $(2 \cdot 1 + 1 \cdot 3) \bmod 10$  entspricht. Die Ziffern 1 und 3 wurden gewählt, da alle Vielfachen dieser Ziffern teilerfremd zu 10 sind und sich so abhängig von den eingelesenen Ziffern immer eine andere Prüfziffer ergibt. Selbiges gibt für die Ziffern 7 und 9.

Ein gutes Gegenbeispiel sind die Ziffern 2 und 5, da Vielfache dieser Zahlen nicht mehr teilerfremd zu 10 sind. So sind z.B.  $2 \cdot 5 \bmod 10$  und  $4 \cdot 5 \bmod 10$  identisch und folglich würde ein einfacher Tippfehler (2 statt 4) nicht mehr mit Hilfe der Prüfziffer erkannt werden können.

Die oben dargestellte Prüfzifferberechnung hat aber leider eine Schwäche, da nur 90% der Vertauschungsfehler erkannt werden können. Sollten nämlich zwei Zahlen, die sich nebeneinander befinden vertauscht worden sein und der Abstand dieser Ziffern beträgt 5 (z.B. 0 und 5,

1 und 6 usw.), kann eine Transposition nicht durch die Prüfziffer festgestellt werden, da  $(1*0 + 5*3) \bmod 10$  auch  $(1*5 + 3*0) \bmod 10$  entspricht.

Die Prüfziffer des ISBN-10 Codes löst dieses Problem, indem diese durch die Division mit Rest mit der Zahl 11 bestimmt wird. Da alle Vielfachen der Ziffern im Code teilerfremd zu 11 sind, können dadurch nicht nur alle Tippfehler, sondern auch Vertauschungsfehler erkannt werden. Jedoch kann die Prüfziffer nun aber auch zweistellig sein. Um dies zu vermeiden, wird für den Fall, dass die 10 als Prüfziffer berechnet wurde, die Zahl durch ein X ersetzt.

## 5.1 Wofür benötigt man Barcodes?

Die Einführung von Barcodes hat uns viele Vorteile gebracht:

- Alle weltweit gehandelten Artikel können identifiziert werden.
- Das Erfassen von Waren an der Supermarktkasse geht wesentlich schneller.
- Einzelne Produkte in einem Laden müssen nicht mehr mit Preisen ausgezeichnet werden, es reicht aus, wenn der Preis am Regal steht.
- Auch als Kunde kann man heute mit Hilfe des Smartphones Informationen von Waren auslesen.
- Tipp- und Übertragungsfehler an der Kasse werden vermieden.
- Der internationale Handel wird erleichtert.
- Die automatische Lagerhaltung hat sich vereinfacht, indem sowohl Produkte als auch Regalplätze mit Codes ausgestattet wurden. Bei jeder Bewegung wird beides gescannt, so ist der aktuelle Lagerplatz eines Produkts immer bekannt.
- Auch in medizinischen Labors wird mittlerweile auf Barcodes zurückgegriffen: Blutproben etwa werden damit versehen, dadurch ist eine eindeutige aber trotzdem anonyme Zuordnung möglich.
- Post- und Warensendungen können sogar von Kunden ohne Probleme über das Internet verfolgt werden.

## 5.2 Der QR-Code

Der QR-Code (Quick Response) wurde Anfang der 90er Jahre des 20. Jahrhunderts in der japanischen Automobilindustrie entwickelt, um viele Informationen auf kleinem Platz unterzubringen.

Der Code besteht aus einer quadratischen Matrix aus schwarzen und weißen Punkten, die in horizontaler und vertikaler Richtung angeordnet sind – daher bezeichnet man QR-Codes auch als 2D-Codes. Genau wie beim Barcode stellt auch hier eine bestimmte Abfolge einen definierten Wert, beispielsweise eine Zahl oder einen Buchstaben, dar. In drei Ecken befindet sich ein bestimmtes Muster, wodurch Lesegeräte erkennen, wie der quadratische Code entschlüsselt werden muss.

Es gibt mittlerweile vielfältige Einsatzmöglichkeiten:



- Insbesondere in der Werbung werden QR-Codes genutzt. Auf Plakaten, in Zeitschriften oder direkt auf Produkten sind sie zu finden. Sie führen meisten auf Internetseiten, auf denen weitere Produktinformationen verfügbar sind.
- Hinterlegter Text: In Form von Eintrittskarten oder Fahrscheinen bei der Bahn werden Textinformationen in Form von QR-Codes gespeichert.
- Auch Geo-Daten lassen sich speichern und auslesen – zum Beispiel über Orte auf Google Maps.
- Immer häufiger sieht man QR-Codes auch auf Visitenkarten – das Scannen erleichtert die Übernahme der Kontaktdaten in das eigene Smartphone.
- Mittlerweile gibt es sogar schon QR-Codes in Form von Tattoos und Schmuckstücken.

Für das Auslesen werden zahlreiche Apps für Smartphone und Tablets angeboten. Mit Hilfe von QR-Code-Generatoren können ganz einfach selbst Codes erzeugt und Informationen hinterlegt werden.

## 6 Unterrichtliche Umsetzung

Innerhalb dieses Moduls werden in der zweiten Doppelstunde Apps (Applikationen) für das Smartphone verwendet. Diese müssen unter Umständen erst noch installiert werden. Für die Installation, aber auch die Verwendung der Apps benötigt man Internetzugriff, etwa via WLAN. Daher sollte sichergestellt werden, dass entweder alle Smartphones der Schülerinnen und Schüler in das Schulnetz aufgenommen werden oder (dies ist eher zu empfehlen) ein separates WLAN für den Zeitraum dieses Moduls zugänglich gemacht wird<sup>1</sup>.

Dieses Modul behandelt den Supermarkt als beispielhaftes Unternehmen ausführlich. Denkbar ist auch, ein anderes Unternehmen, in dem Codes verwendet werden, als Beispiel heranzuziehen und zu besuchen. Je nachdem, welche Codes (Bar- oder QR-Codes) in dem gewählten Unternehmen gebräuchlich sind, ist es empfehlenswert die Exkursion passend zum Unterrichtsverlauf zeitlich zu planen. Hierfür ist es wichtig, sich im Vorfeld im gewählten Unternehmen zu informieren (siehe auch Variante 2).

Zum Abschluss dieses Moduls kann eine QR-Code-Rallye durchgeführt werden. Es empfiehlt sich, dass die Schülerinnen und Schüler einen Lageplan der Schule erhalten, um dort die Standorte ihrer QR-Codes zu markieren. Dies vereinfacht zum einen die Gesprächsführung, da alle Beteiligten sich so gezielt über einzelne QR-Codes austauschen können, zum anderen auch das Entfernen der QR-Codes nach Beendigung der Rallye.

### 6.1 Grober Unterrichtsplan

In diesem Modul werden zwei Varianten für Unterrichtsverläufe vorgeschlagen. Die erste Variante ist durch klare Arbeitsaufträge gekennzeichnet, die zweite Variante ist besonders durch selbstgesteuertes Lernen geprägt.

---

<sup>1</sup> Da die WLAN-Verfügbarkeit in den einzelnen Schulen sehr unterschiedlich ist, sollten im Vorfeld die Möglichkeiten mit dem Systemadministrator geklärt werden.

### 6.1.1 Variante 1

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Aufbau von GTIN 13 – Codes (EAN-Codes) kennenlernen und nachvollziehen.
Vertiefung	Was nützt uns das als Kunde? Welche Informationen können wir auslesen und wie?
Erarbeitung	Supermarkterkundung, falls möglich mit Führung und Fragerunde, ggf. alternativ in geeignetem Unternehmen.
Vertiefung	Wie funktionieren QR-Codes?
Erarbeitung	Rallye mit QR-Codes selber planen und durchführen.

### 6.1.2 Variante 2

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Aufbau von GTIN 13 – Codes (EAN-Codes) kennenlernen und nachvollziehen.
Erarbeitung	Exkursion in ein Unternehmen oder den ortsansässigen Supermarkt.
Erarbeitung	Ausblick: Schülerinnen und Schüler entwickeln mit der Methode Design Thinking neue Ideen dazu, welche Informationen in Zukunft auf Chips gespeichert werden könnten, um den Handel zu optimieren.

## 6.2 Stundenverlaufsskizzen

### Abkürzungen/Legende

AB = Arbeitsblatt; L = Lehrkraft; MuM = Mitschüler\*innen; SuS = Schüler\*innen; UV = Unternehmensvertreter\*in

#### 6.2.1 Variante 1

### Erste Unterrichtsstunde: Strichcodes

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
10–15 Min.	Einstieg	Lehrkraftvortrag Plenum	<p>L stellt verschiedene Produkte mit Barcodes bereit (Miniprodukte, Bücher, Spielzeug, Lebensmittel etc.) und SuS untersuchen die Produkte, indem sie diese selbst scannen.</p> <p>Frage: Was haben alle Produkte gemeinsam? Was passiert eigentlich, wenn man sie über die Scannerkasse (im Supermarkt) zieht? Hilfestellung durch gezielte Fragen, z. B. Wo werden sie gekauft? SuS berichten über ihre Erfahrung mit Scannerkassen und überlegen, was passiert, wenn man die Miniprodukte über eine echte Scannerkasse zieht.</p> <p><i>Alternative:</i> L kündigt einen kleinen Zaubertrick an: Hierfür lässt er sich den Code eines beliebigen Produkts bis auf die letzte Stelle diktieren und schreibt ihn an die Tafel. Dann bestimmt L die Prüfziffer ohne den Rechenweg preiszugeben. Dies kann ein zweites Mal wiederholt werden. SuS sind verwundert, dass L die letzte Ziffer einfach so bestimmen kann und werden neugierig auf das Thema.</p>	Versch. Materialien mit Strichcodes (Miniprodukte, Lebensmittel, Zeitschriften ...), Smartphones mit Scanner-App
20–30 Min.	Erarbeitung		AB „Strichcodes kennenlernen“ bearbeiten.	AB B3.1
5–10 Min.	Sicherung	Einzel-/Gruppenarbeit	Die Aufgaben auf dem Arbeitsblatt werden besprochen, Teile können auch als Hausaufgabe gegeben und in der nächsten Stunde besprochen werden.	AB B3.1

## Zweite bis vierte Unterrichtsstunde: Strichcodes im Supermarkt

Zeit	Phase	Sozialform/ Lehrerimpuls	Inhalt/Unterrichtsgeschehen	Materialien
10-20 Min.	Einstieg	Lehrerkraftvortrag Plenum	L erzählt, ein*e Freund*in von ihm* ihr (oder er*sie selbst/seine* ihre Mutter etc.) habe eine Lebensmittelunverträglichkeit. Er*sie ernähre sich glutenfrei/laktosefrei o. ä.  Frage: Worauf muss er*sie achten? Wie kann er*sie sich informieren? L sammelt die SuS-Ideen an der Tafel; SuS notieren diese.	Tafel
10 Min.	Hinführung	Lehrerkraftvortrag	L stellt verschiedene Apps vor, die auf das gegebene Szenario passen (Liste mit Apps im Anhang).	Beamer, Smartphone mit versch. Apps
40-90 Min. (plus Weg zum Supermarkt)	Erarbeitung	Gruppenarbeit und Exkursion	SuS bilden Kleingruppen, installieren eine App pro Gruppe und untersuchen verschiedene Lebensmittel (Produkte können von zu Hause mitgebracht oder im nächsten Supermarkt getestet werden). SuS füllen dabei AB aus.  Falls möglich, Gespräch mit dem*der Marktleiter*in und Führung „hinter die Kulissen“ des Supermarktes (vorher von der Lehrkraft oder dem*der Unternehmensvertreter*in mit dem*der Marktleiter*in zu vereinbaren). Gespräch und Führung mit dem*der Filialeiter*in durch das Lager und den Anlieferungsbereich des Supermarktes, mögliche Themen oder Fragen: Wann werden die Waren gescannt und registriert? Wie funktioniert das Barcode-System aus Firmensicht? Welche Veränderungen hat die Einführung der Scannerkassen mit sich gebracht? Wo sind die Preise gespeichert? Kann der Supermarkt automatisch Waren nachbestellen und wie ist das möglich? Welche Geräte im Markt (außer der Kasse) arbeiten mit Barcodes (z. B. Pfandautomat) und welche Probleme treten dabei auf? Vielleicht dürfen die Schüler*innen auch selbst etwas scannen und vielleicht sogar die Miniprodukte auf der echten Kasse ausprobieren und damit ihre Vermutungen aus der ersten Stunde überprüfen.	Smartphones mit App, AB B3.2, AB B3.3

			Falls geeignet, kann der Supermarkt auch durch das Partnerunternehmen ersetzt werden.	
35 Min.	Vertiefung	Gruppenarbeit	SuS recherchieren selbstständig im Internet zu ihren Produkten, vergleichen ihre Ergebnisse mit denen der Apps, schreiben eine Bewertung unter Berücksichtigung verschiedener Aspekte (siehe AB) und erstellen eine Präsentation aus den Ergebnissen.	PC mit Internetzugang, AB B3.3
20 Min.	Sicherung	Plenum Präsentation	Kleingruppen präsentieren ihre Ergebnisse (Powerpoint, Poster o. ä.), ggf. Reflexion des Gesprächs und der Führung im Supermarkt, schriftliche Zusammenfassung der Einsatzzwecke und der gespeicherten Informationen in Barcodes im Supermarkt bzw. Unternehmen.	Beamer, PC

### Fünfte Unterrichtsstunde: QR-Codes kennenlernen

Zeit	Phase	Sozialform/ Lehrerimpuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Lehrerkraftvortrag	L zeigt QR-Codes und fragt, wo SuS diese bereits gesehen haben und für welchen Zweck sie eingesetzt werden (Werbung, Handyapps, Ticketsysteme etc.). L kann gezeigte QR-Codes scannen und das Ergebnis SuS zeigen. Evtl. kurze Erläuterung zu QR-Scanner-Apps für das Smartphone. AB wird ausgeteilt, SuS sollen die Fragen auf dem AB mithilfe von Internetrecherche/Materialsichtung beantworten (Geschichte, Einsatzmöglichkeiten, Verschlüsselung, Aufbau).	QR-Code-Abbildungen, B3.4, AB B3.5 SuS
25 Min.	Erarbeitung	Gruppenarbeit	SuS recherchieren einzeln oder in Zweiertteams die Antworten zu den Fragen des AB.	Smartphone mit QR-Scanner, AB B3.5 (falls kein Internet), Computer mit Internetzugang
10 Min.	Sicherung	Plenum	Die gefundenen Antworten werden verglichen.	AB B3.5 SuS

## Sechste bis achte Unterrichtsstunde: QR-Code-Rallye

Zeit	Phase	Sozialform/ Lehrerimpuls	Inhalt/Unterrichtsgeschehen	Material
10–15 Min.	Einstieg	Lehrerkraftvortrag	L erläutert Aufgabe: Konzeption einer Schulrallye mit dem QR-Code-Generator (allgemein oder thematische Eingliederung in andere Fächer möglich). Erarbeiten von Kriterien einer „guten“ Rallye (z. B. interessante Fragen und Aufgaben, gut platzierte QR-Codes etc.). L gibt einen Überblick über den Ablauf der nächsten Stunden, erläutert ggf. die Spielregeln und teilt Gruppen ein. Als kurze Erläuterung zu Generierung von eigenen QR-Codes kann das Arbeitsmaterial B3.6 verteilt werden.	AB B3.6, AB B3.7
80 Min.	Praxisphase I	Gruppenarbeit	SuS planen eine Rallye durch die Schule zu einem selbstgewählten Thema mithilfe eines QR-Code-Generators (Gruppenlabel).	AB B3.7, Computer mit Internetzugang, Drucker, Smartphones mit QR-Scanner
15 Min.	Praxisphase II	Gruppendurchlauf	SuS-Gruppen tauschen die vorbereiteten Rallyes untereinander, sodass jede Gruppe die Rallye einer anderen ausführt.	Smartphones mit QR-Scanner, vorbereitete Rallyes
25 Min.	Evaluierung	Plenum	Bewertung der einzelnen Rallyes anhand vorgegebener Kriterien, Vorstellung im Plenum.	Tafel

### 6.2.2 Variante 2

#### Erste bis vierte Unterrichtsstunde: Einführung

Die folgende Unterrichtseinheit ist vom selbstbestimmten Lernen der Schülerinnen und Schüler geprägt. Als Lehrkraft geben Sie den äußeren, insbesondere den zeitlichen Rahmen vor. Daher kann für den folgenden Ablauf eine Doppelstunde, aber auch bei Bedarf mehr Zeit eingeplant werden.

Zeit	Phase	Sozialform/ Lehrerimpuls	Inhalt/Unterrichtsgeschehen	Material
	Vorbereitung	Einzelaufgabe	SuS werden gebeten, Codes für die nächste Stunde mitzubringen – ausgeschnitten oder ganze Verpackungen (z.B. QR-Codes aus Werbung in Zeitschriften, Barcodes von Lebensmittelverpackungen, Büchern, Paketsendungen, ...).	
10–15 Min.	Einstieg	Lehrerkraftvortrag Plenum	SuS präsentieren ihre Codes und scannen sie selbst, ggf. kann L noch Miniprodukte mit Barcodes zur Verfügung stellen.  Frage: Was haben alle Produkte gemeinsam? Was passiert eigentlich, wenn man sie über die Scannerkasse im Supermarkt zieht? Wie funktionieren Codes?	Versch. Materialien mit Strichcodes (Miniprodukte, Lebensmittel, Zeitschriften...), Smartphones mit Scanner-App
60–180 Min.	Erarbeitung	Einzel- oder Gruppenarbeit	SuS recherchieren selbstbestimmt zu den Themen Barcode und QR-Code und erstellen eine Präsentation dazu. Je nach Zeit kann die Präsentation als Powerpoint, als einfaches Wandplakat oder auch als Erklärvideo, beispielsweise mit der kostenlosen Software Powtoon ( <a href="http://www.powtoon.de">www.powtoon.de</a> ), erstellt werden.	AB B3.5, ggf. PCs, Internet
30 Min.	Sicherung	Plenum	SuS präsentieren die Ergebnisse.	Ggf. Beamer, Laptop, Boxen

#### Fünfte bis siebte Unterrichtsstunde: Exkursion in ein Unternehmen

Zeit	Phase	Sozialform/ Lehrerimpuls	Inhalt/Unterrichtsgeschehen	Materialien
15 Min.	Einstieg/Empfang	Vortrag des Unternehmens	Begrüßung der Schulklasse, Vorstellung des Unternehmens, ggf. mit Präsentation	AB B4.8
20–50 Min.	Hinführung/ Betriebsbesichtigung	Vortrag	SuS erhalten eine Führung durch das Unternehmen, dabei wird insbesondere auf die Bedeutung von Codes (Barcodes oder QR-Codes) eingegangen: Wo	

			kommen sie im Unternehmen überall zum Einsatz? Welche Bedeutung hatte die Einführung von Codes? ...	
20–40 Min.	Erarbeitung/ Praxisphase	Gruppenarbeit	SuS bekommen die Gelegenheit, selbst auszuprobieren, wie die Arbeit mit Codes funktioniert.	
20 Min.	Sicherung	Plenum/ Abschlussphase	SuS haben die Möglichkeit, offene Fragen zu klären, die Erkundung Revue passieren zu lassen, Feedback zu geben.	

### Achte bis neunte Unterrichtsstunde: Design Thinking

Weitere Informationen zum Thema Design Thinking erhalten Sie im Methodenmodul M1 von IT2School.

Zeit	Phase	Sozialform/ Lehrererimpuls	Inhalt/Unterrichtsgeschehen	Materialien
15 Min	Einstieg/ Empfang	Plenum	Feedback zur Exkursion: Was hat die Klasse mitgenommen? Anschließend wird der Frage nachgegangen, wie die Entwicklung in Zukunft weitergehen kann: Haben wir dann noch Barcodes oder QR-Codes oder gibt es andere Möglichkeiten? Erklärung der folgenden Aufgabe.	
10 Min.	Vertiefung	Plenum	<b>Verstehensphase/Aufgabe:</b> Ein neuer Chip soll entwickelt werden. Welche Informationen könnten darauf gespeichert sein, um die Wirtschaft, aber auch unser Leben zu Hause zu revolutionieren? Standpunkte werden definiert und Gruppen aufgeteilt; Gruppeneinteilung der Klasse, ggf. jedes Thema doppelt verteilen = 6 Gruppen à etwa 5 Pers.; Personas werden vorgegeben.	AB B3.10
5 Min.	Vertiefung	Gruppenarbeit	<b>Beobachten + Synthese:</b> Die Gruppe überlegt, welche besonderen Wünsche, Bedürfnisse oder Probleme die Personas haben könnten (hier fließen die Erkenntnisse aus der Exkursion mit ein).	
10–20 Min.	Praxisphase	Gruppenarbeit	Ideen sammeln: <b>1. Schritt:</b> Ideenfindung – jedes Gruppenmitglied soll so viele Ideen wie	Haftnotizen („Post-its“), Papier, Klebe-

			möglich skizzieren (10 Min.). <b>2. Schritt:</b> Vorstellung der Ideen in den Gruppen, Ideen sortieren (5 Min.). <b>3. Schritt:</b> Abstimmen mit Klebepunkten, an der besten Idee wird weitergearbeitet (5 min.).	band, Stifte, Klebepunkte
10 Min.	Praxisphase	Gruppenarbeit	<b>Prototyp bauen:</b> Mit Bastelmaterialien wird ein erster Prototyp gebaut, um die Idee <i>begreifbar</i> zu machen.	Papier, Schere, Pappe, Lego, Knete, Pfeifenreiniger, Klebeband, Stifte etc.
10 Min.	Testen	Gruppenarbeit	<b>Testen:</b> Zwei Gruppen stellen sich gegenseitig ihren Prototypen vor, Feedback wird gegeben. Ggf. nacharbeiten und Korrekturen vornehmen.	
20 Min.	Präsentation	Plenum	SuS stellen ihren neuen Chip in der Klasse vor.	
15–20 Min.	Sicherung	Plenum	Diskussion: Welche Vor- aber auch Nachteile hat der Einsatz von kontaktloser Übertragungstechnik, insbesondere für uns Verbraucher?	



## 7 Einbettung in verschiedene Fächer und Themen

Bei diesem Modul lassen sich viele Bezüge zu anderen Fächern herstellen. Deswegen kann das gesamte Modul oder können einzelne Teile in unterschiedlichen Fächern eingesetzt werden.

Die folgenden Kompetenzen finden sich entweder in den Bildungsstandards der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

### **Informatik**

Die Schülerinnen und Schüler ...

- können den Einfluss von Veränderungen in der Informationstechnologie auf Individuum und Gesellschaft, sowie Arbeitswelt reflektieren.
- kennen den Zusammenhang von Information und Daten erfassen sowie verschiedene Darstellungsformen für Daten.
- setzen sich mit der Vielfalt von Informatiksystemen im Alltag auseinander.

### **Verbraucherbildung**

Die Schülerinnen und Schüler ...

- kennen Produktionskennzeichnungen sowie Prüf- und Qualitätssiegel.
- beschaffen und erfassen Produktinformationen und werten die Informationen aus und beurteilen diese.

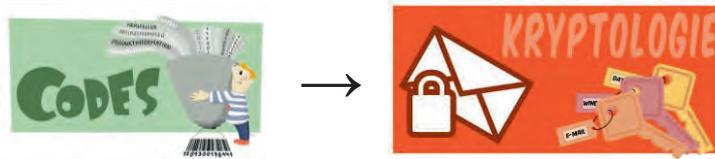
### **Wirtschaftslehre/Wirtschaft-Arbeit-Technik**

Die Schülerinnen und Schüler ...

- setzen sich mit den ständig verändernden Strukturen der Berufs- und Arbeitswelt auseinander.
- erschließen sich selbstständig und in Kooperation mit anderen (mithilfe verschiedener alter und neuer Medien sowie elementarer Lern- und Arbeitstechniken) politische, gesellschaftliche und wirtschaftliche Sachverhalte.
- können im Bereich Produktion und Unternehmen Technisierungsstufen an Beispielen unterschiedlicher Epochen darstellen.
- können historische und gegenwärtige Entwicklungslinien technischer Systeme analysieren und bewerten.

## 8 Anschluss Themen

Als Anschluss Themen im Zusammenhang mit IT2School bietet sich folgendes Modul an:



Die Kryptologie ist ein wichtiger Bestandteil der Datensicherheit in verschiedensten Unternehmen. Zusätzlich bietet es ein großes Motivationspotenzial bei Schülerinnen und Schülern und besitzt viele Anknüpfungspunkte an den Alltag. Im Aufbaumodul A2 befassen sich die Schülerinnen und Schüler sowohl mit einigen historischen Verschlüsselungsverfahren, aber auch mit dem Knacken von solchen Verfahren und der praktischen Anwendung in Form von Datei- und E-Mail-Verschlüsselung.

## 9 Literatur und Links

- Wikipedia: [https://en.wikipedia.org/wiki/International\\_Article\\_Number](https://en.wikipedia.org/wiki/International_Article_Number)
- Wikipedia: [https://en.wikipedia.org/wiki/Check\\_digit](https://en.wikipedia.org/wiki/Check_digit)
- EAN-Suche: **Welches Produkt verbirgt sich hinter der Nummer.** Online: <http://ean-suche.org>
- GS1 Germany GmbH: **Lehrfilm Strichcodes:** Das Einmaleins des Barcodes (Ausschnitt) Online: <https://www.youtube.com/watch?v=2b1Txpgj-r8>
- **QR-Code-Generator.** Online: <http://goqr.me/de/>
- Harzt, Wilko (2013): **Basiswissen QR-Code.** Online: <http://qrcode.wilkohartz.de>
- Simmetsberger; Ursula (2013): **QR-Codes im Unterricht.** Online: <https://www.schule.at/tools/detail/-d371ffe399.html>
- **Individuelle Gestaltung von QR-Codes.** Online: <http://www.wonderqrcode.de/>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
😊 B3.1	Strichcodes kennenlernen	Arbeitsblatt, das den Aufbau von Strichcodes und die Bestimmung der Prüfziffer erklärt.
😊 B3.2	App-Liste	Liste mit Beispiel-Apps zum Scannen von Lebensmitteln.
😊 B3.3	App-Guide SuS	Arbeitsblatt zur Untersuchung und Bewertung von Lebensmittel-Apps.
😊 B3.4	QR-Codes	Dokument mit Beispiel-QR-Codes.

 B3.5 L	QR-Recherche	Dokument mit abgedrucktem Arbeitsauftrag.
 B3.5 SuS	QR-Recherche	Arbeitsblatt, das dazu dient, den Aufbau und die Funktionsweise von QR-Codes kennenzulernen. Arbeitsauftrag steckt im QR-Code.
 B3.6	QR-Code selber machen	Kurze Erklärung, wie QR-Codes selber generiert werden können.
 B3.7	QR-Rallye	Arbeitsblatt mit Hinweisen zur Erstellung einer Schulrallye.
 B3.8	Codes im Supermarkt und im Unternehmen	Arbeitsblatt, das dazu anleiten soll, sich selbst-gesteuert mit Barcodes und QR-Codes auseinanderzusetzen.
 B3.9	Betriebserkundungen planen und durchführen	Leitfaden und Empfehlungen für Lehrerinnen und Lehrer, sowie für Unternehmensvertreterinnen und -vertreter zur Umsetzung von Betriebs-erkundungen.
 B3.10	Die Zukunft der Codes	Arbeitsblatt zur Entwicklung eigener Ideen für die Zukunft.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Algorithmus	Als Algorithmus wird eine Handlungsvorschrift bezeichnet, deren einzelnen Handlungsanweisungen eindeutig und deterministisch (endlich; zeitlich begrenzt) sind. Algorithmen beschreiben meist, wie gegebene Problemstellungen gelöst werden oder bestimmte Tätigkeiten durchzuführen sind.
Barcode	1D-Code; wird auch Strichcode genannt, besteht aus parallelen Strichen in verschiedenen Breiten. Mithilfe des Barcodes können Informationen in binären Symbolen dargestellt werden, häufig dargestellte Information ist die GTIN-13-Nummer.
EAN-Code	Europäische Artikelnummer und unverwechselbare Produktkennzeichnung, wird durch Strichcode dargestellt.
GTIN-13	Global Trade Item Number, hat 2009 den → EAN-Code abgelöst, stellt Produktnummer durch einen Strichcode dar.

QR-Code

2D-Barcode, zweidimensionaler Strichcode, bestehend aus schwarzen Vierecken, Die Information kann nicht aus einer einzelnen Zeile gelesen werden, sondern der Code muss als Ganzes erfasst werden.

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Strichcodes kennenlernen

## Aufgabe 1

- a) Holt euch mindestens drei der bereitgestellten Artikel an euren Platz und beschreibt sie mit folgenden Fragen:
- Woraus bestehen die Codes? Ist ein Muster/Raster erkennbar?
  - Tauscht eure Aufzeichnungen mit eurem Sitznachbarn und vergleicht eure Beobachtungen.

## Berechnung der Prüfziffern von Barcodes

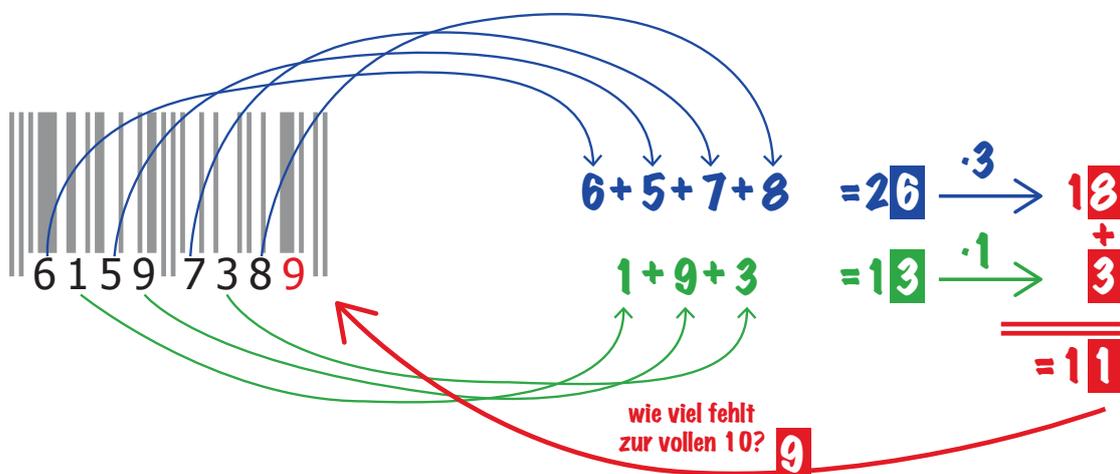
Um die fehlerfreie Funktionsweise der Codes zu gewährleisten, steht am Ende der Codefolge eine sogenannte Prüfziffer (siehe Pfeil). Der Algorithmus für die Prüfziffer lautet:

Alle Ziffern werden in Leserichtung abwechselnd mit 3 und 1 multipliziert und dann aufsummiert ( $3 \times \text{Ziffer 1} + 1 \times \text{Ziffer 2} + 3 \times \text{Ziffer 3} + 1 \times \text{Ziffer 4} + \dots$ ), die Prüfziffer ist dann die Zahl, die zu einer vollen Zehnerzahl (10, 20, 30, 40, ...) fehlt.

**Beispiel:** Das Ergebnis der Aufsummierung für den folgenden Strichcode ist 91, weshalb die Prüfziffer 9 ist.



Mathematisch lässt sich dieser *Algorithmus* zur Berechnung vereinfachen und in einem Diagramm veranschaulichen:



So kann man zur Berechnung der Prüfziffer zunächst alle Ziffern, die an ungerader (also an 1., 3., 5. usw.) Stelle von links im Code vorkommen, in einer Zeile aufschreiben, ihre Summe berechnen und die letzte Ziffer dieser Summe mit 3 multiplizieren (**erster Schritt**).

In die untere Zeile schreibt man alle Ziffern, die an gerader (also an 2., 4., 6. usw.) Stelle im Code vorkommen. Die letzte Ziffer ihrer Summe multipliziert man dann mit 1 (**zweiter Schritt**).

Als nächstes addiert man dann die letzten Ziffern der beiden Ergebnisse. Die Prüfziffer ist dann die Zahl, die zur nächsten Zehnerzahl fehlt (**dritter Schritt**).

## Aufgabe 2

Berechnet die Prüfziffern für folgende Codes und schreibt den Rechenweg mit auf.

a)   
 1 1 1 3 3 4 2

b)   
 8 7 6 2 3 4 4

# Code-Typen

Es gibt viele verschiedene Typen von Strichcodes. Die **GTIN-8** (Global Trade Item Number) oder – wie bis 2009 genannt – Europäische Artikelnummer (EAN) besteht aus lediglich acht Ziffern und wird somit vor allem auf Verpackungen gedruckt, die zu klein für längere Codes sind.

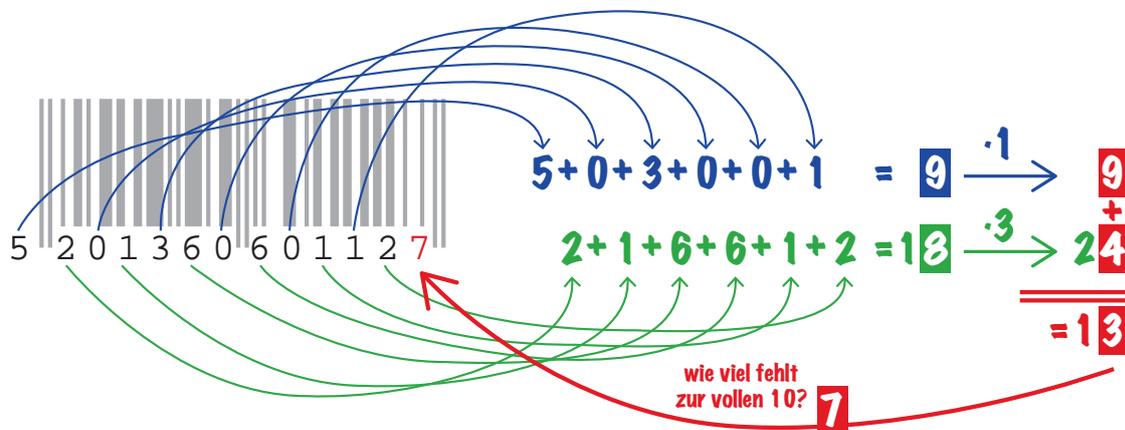


Doch auf einer Vielzahl von Artikeln und Verpackungen im Einzelhandel finden sich auch längere Codes. Am häufigsten kommt die 13-stellige **GTIN-13** vor.



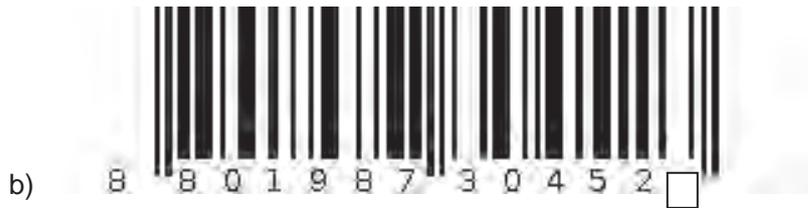
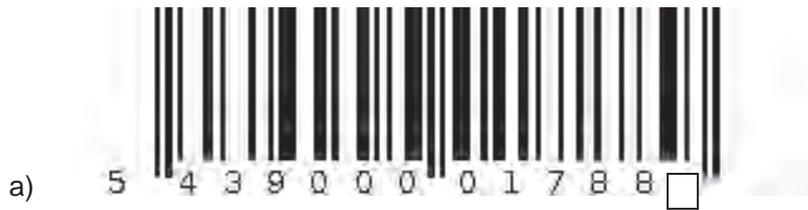
Auch bei den GTIN-13-Codes kann man die Prüfziffer berechnen. Das funktioniert im Grunde genauso, wie oben für die GTIN-8-Codes beschrieben. Allerdings beginnt man diesmal die Multiplikation mit 1 und wechselt dann zur 3 und so weiter. Im Anschluss wird wieder aufsummiert und die Zahl, die zur vollen Zehnerzahl fehlt, ist die Prüfziffer.

Auch dieser *Algorithmus* lässt sich (wie bei den GTIN-8 Codes) vereinfachen und in einem Diagramm darstellen:



## Aufgabe 3

Berechnet die Prüfziffern für folgende Codes, schreibt den Rechenweg mit auf.



# Das Geheimnis der Streifen

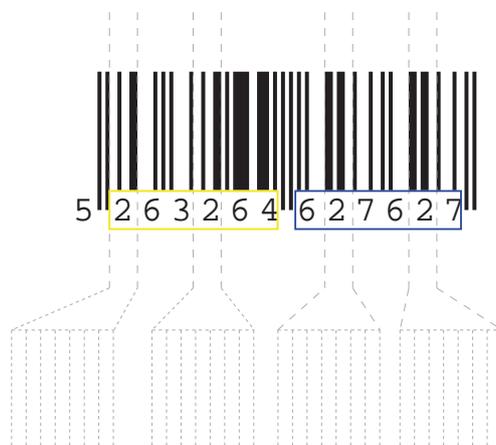
Ein Barcode besteht immer aus einer Darstellung als Strichcode und als Ziffernfolge. Die Bedeutung der Ziffernfolge habt ihr schon kennengelernt, aber wie kommen die Streifen zustande?

## Aufgabe 4

Untersucht in Partnerarbeit die Striche auf den Codes genauer.

- a) Übertragt die dargestellten Codierungen auf die unten vergrößerten Bereiche.

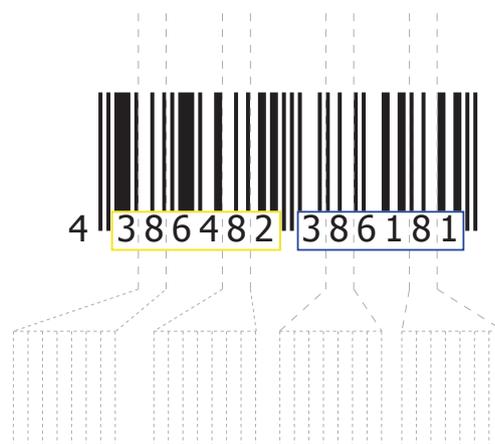
**Code für die Ziffer 2:**



**Tip:** Die 2 wird links anders kodiert als rechts, das ist kein Fehler. Die Erklärung folgt später. Übrigens sind nicht nur die schwarzen Striche von Bedeutung, sondern auch die weißen Abstände.

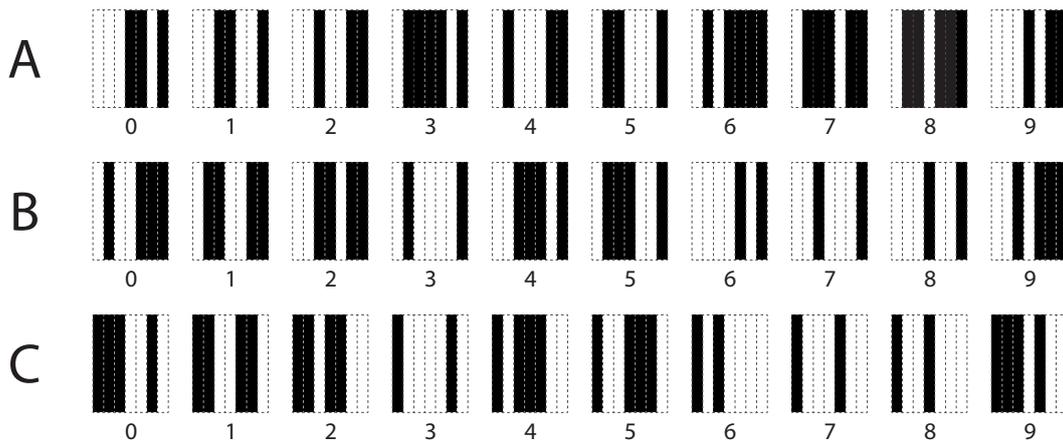
**Hinweis:** Vor und nach den beiden mittleren Trennbalken steht jeweils immer ein weißer Abstand. Diese beiden Stellen werden also nicht mitgelesen.

**Code für die Ziffer 8:**



- b) Was fällt euch bezüglich der Darstellung der Zahlen auf?
- c) Wie unterscheiden sich die Darstellungen der beiden Ziffern „2“ (oberes Beispiel) bzw. „8“ (unteres Beispiel) auf der linken (gelb) und der rechten Seite (blau)?

Wie ihr bereits herausgefunden habt, gibt es pro Ziffer drei verschiedene Arten der Darstellung (A, B und C):



Guckt man sich diese „Codereihen“ an, so fällt auf, dass die Darstellungen der Ziffern in den Codereihen A und C „vertauscht“ sind: Jede weiße Leerstelle in A wird in C zu einem schwarzen Balken und umgekehrt.

Zwischen den Darstellungen in den Codereihen B und C gibt es auch einen Zusammenhang: Sie sind nämlich „gespiegelt“.

Interessant ist auch, dass die Ziffern der Codereihen A und B immer mit (mindestens) einer weißen Leerstelle beginnen und mit (mindestens) einem schwarzen Balken enden.

## Aufgabe 5

Überprüft, ob folgende Aussage wahr oder falsch ist:

„Die Codierungen nach der Codereihe A sind immer aus einer ungeraden Anzahl an schwarzen Balken aufgebaut und die Codierungen nach Darstellung B und C sind immer aus einer geraden Anzahl an schwarzen Balken aufgebaut.“

## Aufgabe 6

- a) Untersucht nun die folgenden zwei Barcodes aus Aufgabe 4a und kreuzt an: Nach welcher Codereihe sind die Ziffern „2“ und „8“ links und rechts jeweils codiert?

A  
 B  
 C

A  
 B  
 C

A  
 B  
 C

A  
 B  
 C

A  
 B  
 C

A  
 B  
 C

A  
 B  
 C

Was fällt euch auf?

- b) Stellt eine begründete Vermutung darüber auf, warum die Ziffern auf der linken und rechten Seite des Barcodes unterschiedlich codiert werden.

Welche Codereihe verwendet wird, hängt von der ersten Ziffer der Länderkennung ab.

**GTIN-13:** Im linken Teil (Ziffer 2 – 7) wird die Codereihe A oder B verwendet. Im rechten Teil des Strichcodes (Ziffer 8 – 13) wird nur die Codereihe C verwendet.

GTIN-13		
Erste Ziffer	Links (Ziffer 2 – 7)	Rechts (8-13)
0	AAAAAA	CCCCCC
1	AABABB	CCCCCC
2	AABBAB	CCCCCC
3	AABBBA	CCCCCC
4	ABAABB	CCCCCC
5	ABBAAB	CCCCCC
6	ABBBAA	CCCCCC
7	ABABAB	CCCCCC
8	ABABBA	CCCCCC
9	ABBABA	CCCCCC

**GTIN-8:** Links von den Trennstrichen wird stets die Codereihe A verwendet, rechts immer Codereihe C.

GTIN-8	
Links (Ziffer 1 – 4)	Rechts (Ziffer 5 – 8)
AAAA	CCCC

Doch woher weiß der Barcode-Scanner, nach welchen Codereihen der Strichcode codiert wurde? Schließlich wird die erste Ziffer nicht als Strichcode dargestellt. Diesem letzten Geheimnis der Barcodes gehen wir nun gemeinsam auf die Spur.

Bei GTIN-8 Barcodes ist die verwendete Codereihe klar, denn es gibt nur eine.

Bei GTIN-13 Barcodes hingegen ist die erste Ziffer in der verwendeten Codereihe versteckt! Dadurch, dass jede Codierung nach A aus einer ungeraden Anzahl an schwarzen Balken und jede Codierung der Codereihen nach B und C aus einer geraden Anzahl an schwarzen Balken besteht, lässt sich mit einem Blick in die obere GTIN13-Tabelle feststellen, wie die erste Ziffer lauten muss.

Betrachten wir das folgende Beispiel doch einmal aus der Perspektive des Barcode-Scanners, der keine Zahlen, sondern nur schwarze Balken und weiße Leerstellen lesen kann.



Die erste codierte Ziffer besteht aus insgesamt fünf schwarzen Balken und zwei weißen Leerstellen. Da die Anzahl der schwarzen Balken **ungerade** ist, lautet die Regel, dass diese Ziffer nach **Codereihe A** codiert werden muss.

Schauen wir nun in der Codereihe A nach, so sehen wir, dass die Kombination aus einer weißen Leerstelle, drei schwarzen Balken, einer weißen Leerstelle und einem schwarzen Balken der Ziffer „3“ entspricht.

Die zweite Ziffer wird durch insgesamt drei schwarze Balken dargestellt; da drei auch eine **ungerade** Zahl ist, wurde auch diese Ziffer nach **Codereihe A** codiert. Die dritte Ziffer hingegen enthält genau zwei schwarzen Balken. Weil zwei **gerade** ist, wurde diese dritte Stelle also nach **Codereihe B** codiert.

Fährt man nach diesem Schema fort und erhält, dass dieser Barcode nach AABBBB CCCCCC codiert wurde, so lässt sich anhand der GTIN-13-Tabelle von vorhin feststellen, dass die erste Ziffer eine 3 sein muss.

Mit Hilfe der ersten Ziffer lässt sich nun auch die Prüfziffer wie in Aufgabe 2 berechnen. So kann das Barcode-Lesegerät überprüfen, ob die eingelesene Prüfziffer mit der berechneten Prüfziffer übereinstimmt. In diesem Fall kann das Lesegerät davon ausgehen, den Barcode korrekt eingelesen zu haben.

## Aufgabe 7

Suche jeweils ein Produkt mit einem GTIN-8 und einem GTIN-13 Code. Übertrage die Barcodes dieser Produkte in die unteren Abbildungen und achte darauf, dass jede Ziffer strikt durch genau die für sie vorgesehenen 7 Bits codiert wird. Nutze die GTIN-8- bzw. GTIN-13-Tabelle und die Darstellung der Codereihen als Hilfe. Codiere die Ziffernfolgen nach dem gelernten Muster, **ohne** die Ziffern darunter zu schreiben.

Wenn deine Sitznachbarin bzw. dein Sitznachbar und du fertig seid, dann tauscht eure Codierungen aus. Entziffert jeweils eure Barcodes. Sind die Prüfziffern richtig?

## App-Liste

App	Kurzbeschreibung
<p><b>Is It Vegan?</b></p> 	<ul style="list-style-type: none"> <li>∞ Checkt alle Inhaltsstoffe und gibt an, ob etwas vegetarisch oder vegan ist (englisch)</li> <li>∞ Ampelfarben</li> <li>➤ <a href="https://play.google.com/store/apps/details?id=net.isitvegan.androidfree&amp;hl=de">https://play.google.com/store/apps/details?id=net.isitvegan.androidfree&amp;hl=de</a></li> </ul>
<p><b>Codecheck: Barcode und QR-Scanner</b></p> 	<ul style="list-style-type: none"> <li>∞ Liest den Barcode und QR-Code von über 22 Millionen Artikeln</li> <li>∞ Nährwerte und Inhaltsstoffe in Ampelfarben</li> <li>∞ Allergie-Warnungen</li> <li>➤ <a href="https://play.google.com/store/apps/details?id=ch.ethz.im.codecheck&amp;hl=de">https://play.google.com/store/apps/details?id=ch.ethz.im.codecheck&amp;hl=de</a></li> <li>➤ <a href="https://itunes.apple.com/de/app/codecheck-lebensmittel-kosmetik/id359351047?mt=8">https://itunes.apple.com/de/app/codecheck-lebensmittel-kosmetik/id359351047?mt=8</a></li> </ul>
<p><b>Open Food Facts</b></p> 	<ul style="list-style-type: none"> <li>∞ Mitmachprojekt (englisch)</li> <li>∞ Zeigt gängige Allergene an (auch „Spuren von ...“)</li> <li>➤ <a href="https://play.google.com/store/apps/details?id=org.openfoodfacts.scanner&amp;hl=de">https://play.google.com/store/apps/details?id=org.openfoodfacts.scanner&amp;hl=de</a></li> <li>➤ <a href="https://itunes.apple.com/en/app/open-food-facts/id588797948">https://itunes.apple.com/en/app/open-food-facts/id588797948</a></li> <li>➤ <a href="http://world.openfoodfacts.org/">http://world.openfoodfacts.org/</a></li> </ul>
<p><b>Barcoo: Barcode&amp;QR-Scanner</b></p> 	<ul style="list-style-type: none"> <li>∞ Lebensmittel-Ampel (Farbkennzeichnung für Nährwerte und Vergleich)</li> <li>∞ Testberichte von Stiftung Warentest</li> <li>∞ Nachhaltigkeitsinformationen zu den Herstellern</li> <li>∞ Informationen zu kritischen Inhaltsstoffen, Bewertungen und Empfehlungen anderer Nutzer</li> <li>➤ <a href="https://play.google.com/store/apps/details?id=de.barcoo.android&amp;hl=de">https://play.google.com/store/apps/details?id=de.barcoo.android&amp;hl=de</a></li> <li>➤ <a href="https://itunes.apple.com/de/app/barcoo-barcode-scanner-qr/id339525465?mt=8">https://itunes.apple.com/de/app/barcoo-barcode-scanner-qr/id339525465?mt=8</a></li> </ul>

# AppGuide

## Aufgaben

1. Installiert in euren Gruppen eine der vorgestellten Apps auf eurem Smartphone und untersucht mit deren Hilfe die Nahrungsmittel auf Inhaltsstoffe, Herstellungsprozesse und eure spezielle Fragestellung (Allergene, vegane Ernährung, Tierversuche etc.). Notiert die Namen der Produkte und die Ergebnisse, die der Scanner liefert.

Produkte	Inhaltsstoffe	Notizen

2. Recherchiert im Internet zu euren Produkten und überprüft, ob sich eure Ergebnisse mit denen der Apps decken oder ob es Unterschiede gibt. Fasst eure Resultate zu einem Kurzvortrag zusammen, den ihr vor euren Mitschülerinnen und Mitschülern haltet. Nutzt zur Veranschaulichung ein Poster oder eine PowerPoint-Präsentation.
3. Schreibt anhand eurer Beobachtungen und Erfahrungen mit der App eine Bewertung, die ihr dann im World Wide Web veröffentlichen könnt. Auf der nächsten Seite sind einige Kriterien aufgelistet, an denen ihr euch orientieren könnt.

# Bewertungskriterien für Apps

## Bedienung

- ∞ In welchen Sprachen ist die App erhältlich?
- ∞ Ist die Bedienung selbsterklärend?
- ∞ Ist das Hauptmenü leicht verständlich?
- ∞ Kann man sich ohne größeren Aufwand wieder abmelden?
- ∞ Gibt es eine Einführung oder Hilfen für die Bedienung?
- ∞ Gibt es Werbeeinblendungen oder Pop-up-Fenster, die sich öffnen?

## Funktionen

- ∞ Was kann man mit der App alles tun?
- ∞ Braucht man Apps zur Ergänzung? Sind Käufe zur Ergänzung klar gekennzeichnet?
- ∞ Muss man für Zusatzfunktionen zahlen?
- ∞ Läuft die App absturzfrei/fehlerfrei?

## Datenschutz

- ∞ Greift die App nur auf die nötigsten Daten zu? Wird erklärt, warum diese Daten benötigt werden?
- ∞ Werden die Daten an Dritte weitergegeben?
- ∞ Gibt es eine Altersbeschränkung/Altersempfehlung?
- ∞ Werden Verbindungen zu sozialen Netzwerken wie Facebook oder Google+ eindeutig angeben und können diese abgeschaltet werden?

## Sonstiges

- ∞ Macht die Nutzung der App Spaß?
- ∞ Ist die App im App-Store leicht zu finden?
- ∞ Ist sie kostenlos/kostengünstig?
- ∞ Sind Titel, Logo und Beschreibung ansprechend?
- ∞ Hält die App, was sie verspricht?
- ∞ Gibt es Besonderheiten in der App?
- ∞ Würden wir die App insgesamt empfehlen?

# QR-Codes



1. Informatik



2. Wissen ist Macht



3. Wissensfabrik

# Recherche zu QR-Codes

## Aufgabe im QR-Code

Informiert euch anhand der gegebenen Materialien oder im Internet zu QR-Codes. Nutzt dabei angebrachte Quellen und belegt eure Antworten mit diesen. Bearbeitet die Aufgaben auf diesem Arbeitsblatt in Tandemarbeit, wir vergleichen die Ergebnisse danach im Klassenverband. Viel Spaß dabei!



## Wichtig

- ∞ Recherchiert wofür das QR in QR-Code steht und was es bedeutet.
- ∞ Wie und wo entstand der erste QR-Code?
- ∞ Beschreibt in eigenen Worten wie ein QR-Code aufgebaut ist.
- ∞ Wie viele Fehlerkorrekturlevel gibt es und wie heißen diese?
- ∞ Was kann man alles in einen QR-Code umwandeln und wie funktioniert das?
- ∞ Nennt verschiedene Anwendungsbereiche für QR-Codes und findet heraus, welche Datentypen in einen QR-Code umgewandelt werden können.
- ∞ Arbeitet heraus, welche Gefahren sich hinter QR-Codes verbergen können, insbesondere was die Nutzung mit dem Handy betrifft.

## Nützliche Internetseiten

- ∞ Hartz, Wilko (2013): **Basiswissen QR-Code**. Online: <http://qrcode.wilkohartz.de>
- ∞ IT Wissen – **Das große Onlinelexikon für Informationstechnologie**: Online: <http://www.itwissen.info/definition/lexikon/quick-response-QR-QR-Code.html>

# Recherche zu QR-Codes

## Aufgabe

Scannt den QR-Code. In ihm steckt die Aufgabe!



## Wichtig

- ∞ Recherchiert wofür das QR in QR-Code steht und was es bedeutet.
- ∞ Wie und wo entstand der erste QR-Code?
- ∞ Beschreibt in eigenen Worten wie ein QR-Code aufgebaut ist.
- ∞ Wie viele Fehlerkorrekturlevel gibt es und wie heißen diese?
- ∞ Was kann man alles in einen QR-Code umwandeln und wie funktioniert das?
- ∞ Nennt verschiedene Anwendungsbereiche für QR-Codes und findet heraus, welche Datentypen in einen QR-Code umgewandelt werden können.
- ∞ Arbeitet heraus, welche Gefahren sich hinter QR-Codes verbergen können, insbesondere was die Nutzung mit dem Handy betrifft.

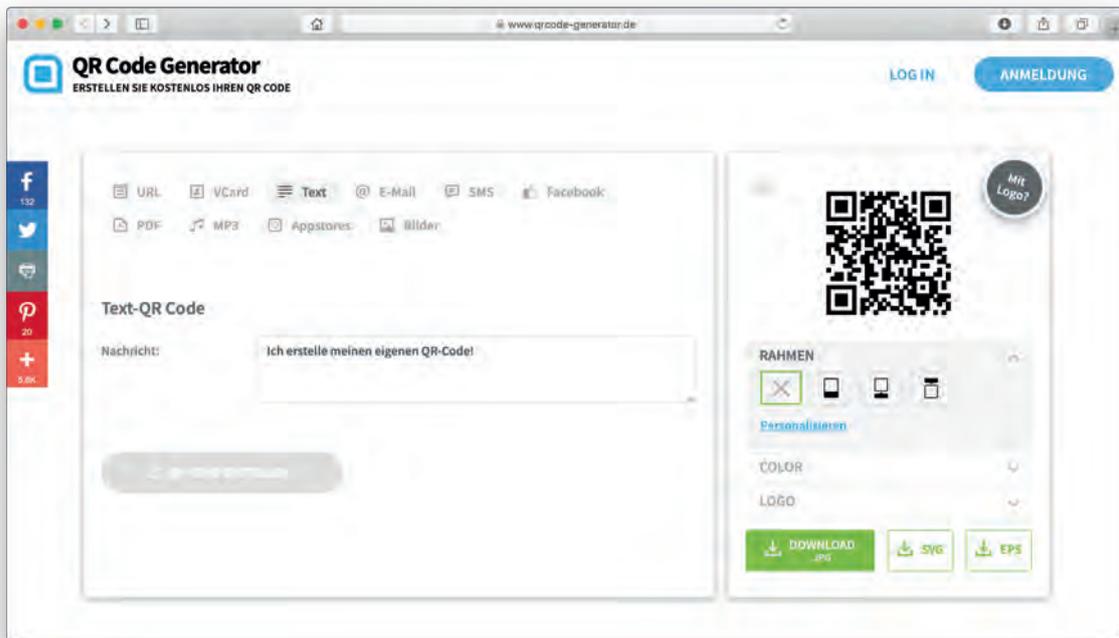
## Nützliche Internetseiten

- ∞ Hartz, Wilko (2013): **Basiswissen QR-Code**. Online: <http://qrcode.wilkohartz.de>
- ∞ IT Wissen – **Das große Onlinelexikon für Informationstechnologie**: Online: <http://www.itwissen.info/definition/lexikon/quick-response-QR-QR-Code.html>

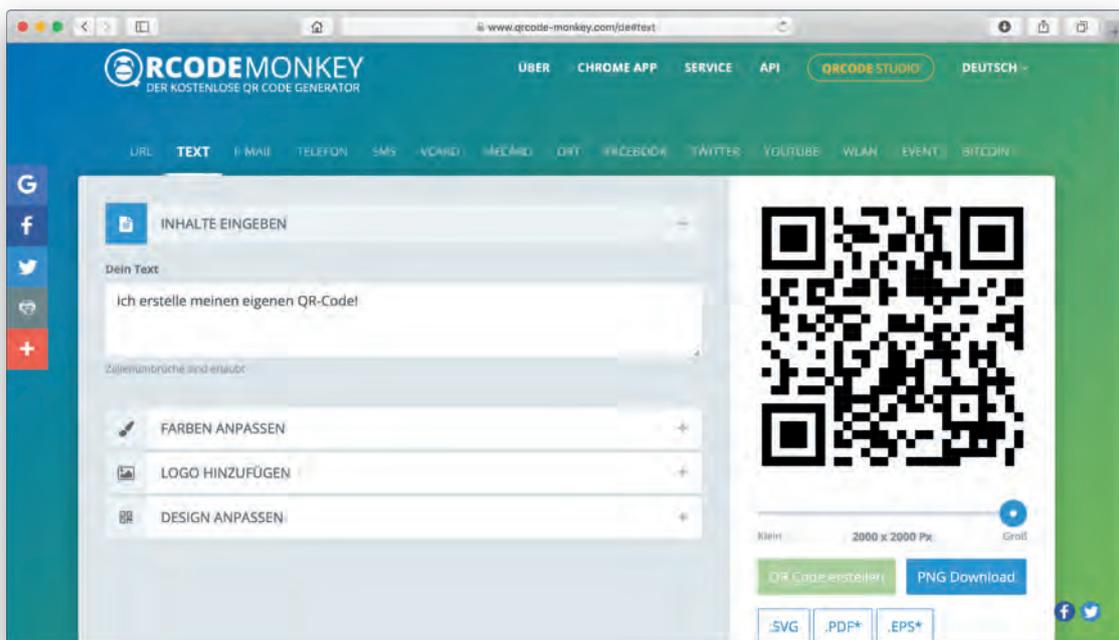
# QR-Codes selbermachen

Das Erstellen eines eigenen QR-Codes ist super einfach und lässt sich mit verschiedenen Online-Generatoren machen. Die folgenden zwei Webseiten können dafür genutzt werden. Sie bieten die Möglichkeit verschiedene Typen (URL, Text, Email, vCard etc.) von QR-Codes zu erzeugen:

<http://www.qrcode-generator.de>



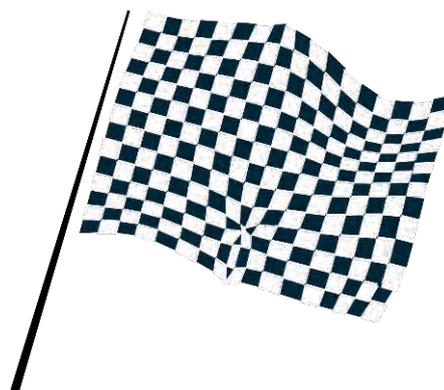
<http://www.qrcode-monkey.de>



# QR-Rallye

## Auf die Plätze ... fertig ... los!

Ihr gestaltet eure eigene Rallye durch die Schule zu einem Thema, das ihr euch selbst aussuchen könnt. Voraussetzung: Es sollte mit einem Schulfach zu tun haben oder mit der Geschichte eurer Schule. Die Fragen stellt ihr in Form von ausgedruckten QR-Codes bereit. Sie führen die Teilnehmer der Rallye zu verschiedenen Orten in eurer Schule und auf eurem Schulgelände.



## Erkundungstour

Wenn ihr euch auf ein Thema geeinigt habt und eure Lehrkraft einverstanden ist, könnt ihr direkt loslegen. Am besten schnappt ihr euch Stift und Papier und macht erstmal eine Erkundungstour auf dem Schulgelände. Wo gibt es interessante Dinge zu sehen, die mit eurem Thema zu tun haben, und an welchen Stellen lassen sich gut QR-Codes anbringen? Sammelt eure Ideen und die geeignetsten Orte. Vielleicht bekommt ihr einen Lageplan von der Schule, um die Orte einzuzeichnen.

## Rätsel, Fragen, Codes generieren

Denkt euch nun Fragen oder Rätsel aus, die die Rallye-Teilnehmer lösen müssen und die Hinweise darauf enthalten, wo sich die nächste Station und der nächste QR-Code befinden. Wandelt eure Rätsel oder Fragen mit Hilfe eines QR-Code-Generators (z.B. <http://goqr.me/de/>) um. Überlegt euch: Wie lassen sich die Codes an den von euch ausgewählten Orten so befestigen, dass sie sich auch leicht wieder entfernen lassen? Wie könnt ihr sie gut vor Regen schützen? Druckt die Codes aus und bringt sie in eine geeignete Form.

## Letzte Vorbereitungen

Macht einen Probendurchlauf und verteilt dabei eure QR-Codes an den richtigen Stellen. Vergewissert euch noch einmal, dass die Codes ordentlich befestigt sind und auch die Reihenfolge stimmt. Fertigt dabei ein Protokoll an, damit ihr nicht vergesst, wo die einzelnen Stationen liegen.

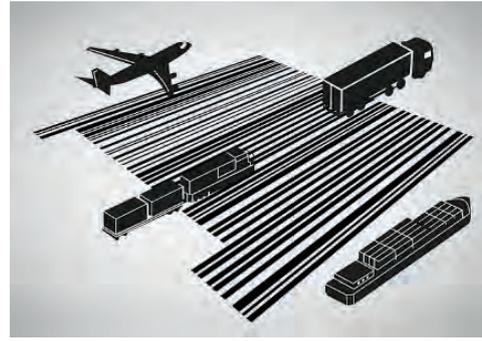
## Zum Schluss

Tauscht eure Rallye mit der einer anderen Gruppe. Ihr führt jetzt die Rallye der anderen aus und sie eure. Macht euch dabei Notizen, damit ihr die Rallye später in der Klasse vorstellen und bewerten könnt. Denkt dabei an die Kriterien, die wir zu Anfang festgelegt hatten.

**Viel Spaß!**

# Codes im Supermarkt und im Unternehmen

Sowohl Barcodes als auch QR-Codes begegnen uns überall im Alltag. Insbesondere im Supermarkt ist dieses charakteristische Kennzeichen auf nahezu jedem Produkt zu finden. Mussten früher Kassiererinnen und Kassierer die Preise manuell in die Kasse eintippen, können heute die meisten Artikel in Sekundenschnelle mit dem Scanner erfasst werden. Auch die Lagerung und die Logistik wurden mithilfe von Barcodes wesentlich vereinfacht. Die Einführung dieses Codes hat vor 40 Jahren die Handelsbranche revolutioniert.



Recherchiert zu unterschiedlichen Themenschwerpunkten und präsentiert eure Ergebnisse in der Klasse.

## Aufgabe

1. Bildet Gruppen mit jeweils 3 bis 4 Schülerinnen und Schülern.
2. Wählt gemeinsam eines der folgenden Themen aus:
  - IT in der Logistik – Wie kleine Striche die Welt verändern
  - Piep! – Wie der Barcode den Supermarkt revolutionierte
  - Informationen im Labyrinth – Die Bedeutung von QR-Codes
3. Überlegt euch Fragen zum Thema, beispielsweise: „Wie funktionieren Codes?“, „Wie sind sie aufgebaut?“ Listet alle Fragen auf.
4. Legt gemeinsam einen Arbeitsplan fest: Wer macht was, bis wann sollen einzelne Teile fertig sein und so weiter. Im Anschluss beantwortet ihr eure Fragen. Recherchiert dazu im Internet.
5. Überlegt gemeinsam, wie ihr eure Ergebnisse am besten präsentieren könnt (PowerPoint-Präsentation, Wandplakat, Produktion eines Erklärvideos, Rollenspiel, ...) und bereitet auf diese Weise euren Vortrag vor.
6. Präsentiert eure Ergebnisse im Plenum.

# Betriebserkundungen planen und durchführen

Für eine Exkursion mit Schülerinnen und Schülern (SuS) zu einem Unternehmen oder einem Supermarkt müssen im Vorfeld Voraussetzungen geklärt und Abläufe geplant werden. Dies gilt sowohl für Lehrerinnen und Lehrer (L) als auch für Unternehmensvertreterinnen und -vertreter (UV). Die folgende Tabelle soll einen Überblick und Hilfestellung bei der Planung geben. Die schwarzen Pfeile signalisieren, dass L und UV sich an dieser Stelle konkret abstimmen sollten:

Lehrerinnen und Lehrer		Unternehmensvertreterinnen und -vertreter
Kontaktaufnahme und Terminabsprache mit dem Betrieb, in dem die Exkursion stattfinden soll, Themenschwerpunkt (Bar- oder QR-Codes) festlegen, Zeiten und Ablauf mit UV besprechen.	↔	Terminabsprache mit Schule, zeitlichen Ablauf festlegen.
Genehmigung durch Schulleitung		Unternehmensvorstellung/Präsentation vorbereiten oder noch einmal durchgehen – schwierige Wörter oder Fachbegriffe sollten erklärt werden.
Planung der Anreise, Reisekosten ermitteln.		Betriebsrundgang zum Thema „Codes“ planen: Wo werden Barcodes oder andere Codes im Unternehmen genutzt, welche Bedeutung haben Codes im Unternehmen, welche Probleme gibt es mit den Codes, wie wird die Codierung in Zukunft aussehen, ...
SuS informieren, Elternbriefe schreiben, Einverständnis einholen.		Abteilungen darüber informieren, dass eine Schülergruppe kommt.
Versicherungsfragen klären.		
Kann der Betriebsleiter praktische Erlebnisse ermöglichen? Kann die Klasse aktiv mit einbezogen werden?	↔	Planen Sie aktive Tätigkeiten der SuS mit ein z. B.: Selbst Codes erstellen und scannen, Waren scannen, Waren aus Kaufmannsladen scannen, ein Wandplakat erstellen, Rallye durchführen etc.
Thematische und methodische Vorbereitung mit den SuS – Einführung in Bar- und QR-Codes.		Bei Interesse einen Feedback-Fragebogen erstellen, um Rückmeldung durch L zu erhalten.

## Exkursion durchführen

Das folgende Schaubild gibt Ihnen einen Überblick über den zeitlichen Ablauf der Exkursion. Diese kann je nach Betriebsgröße angepasst werden.



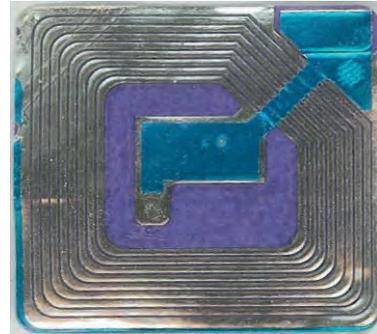
## Informationen für Unternehmensvertreterinnen und -vertreter

Zeit	Sozialform/ Lehrerimpuls	Inhalt/Unterrichtsgeschehen
<b>15 Min.</b>	Vortrag	Begrüßung der Schulklasse, Vorstellung des Unternehmens, ggf. mit Präsentation. Gehen Sie im Vorfeld Ihre Unternehmenspräsentation durch und achten Sie auf schwierige Fremdwörter oder Fachbegriffe, diese sollten Sie während der Präsentation den SuS erklären.
<b>20-50 Min.</b>	Vortrag	SuS erhalten eine Führung durch das Unternehmen, dabei wird insbesondere auf die Bedeutung von Codes (Barcodes oder QR-Codes) eingegangen: Wo kommen sie im Unternehmen überall zum Einsatz, welche Bedeutung hatte die Einführung von Codes, welche Probleme gibt es, wie könnte die Zukunft aussehen, ...
<b>20-40 Min.</b>	Gruppenarbeit	SuS bekommen die Gelegenheit, selbst aktiv zu werden und einen Scanner auszuprobieren. Sollten Sie dafür keine Möglichkeit in Ihrem Betrieb haben, tauschen Sie sich im Vorfeld mit L über mögliche Aufgaben aus.
<b>20 Min.</b>	Plenum/ Abschlussphase	SuS haben die Möglichkeit, offene Fragen zu klären und die Erkundung Revue passieren zu lassen. Feedback wird gegeben, sowohl von UV an SuS als auch von SuS an UV.

# Die Zukunft der Codes

Barcodes und QR-Codes haben die Wirtschaft, vor allem die Logistikbranche und den Einzelhandel, revolutioniert. Die Entwicklung neuer Techniken ist aber noch nicht abgeschlossen, Informatikerinnen und Informatiker, Ingenieurinnen und Ingenieure arbeiten an immer neuen Verfahren, um Arbeitsabläufe oder Produktionsprozesse effizienter zu machen und deren Fehleranfälligkeit zu minimieren.

Eine neue Möglichkeit stellen die sogenannten RFID-Chips (Radio Frequency Identification, deutsch: **R**adio**f**requenz-**I**dentifikation) dar. Mithilfe dieser Technologie können Daten vom Chip berührungslos und ohne Sichtkontakt ausgelesen werden. Während man beim Barcode noch ein Lesegerät benötigt, das Produkte einzeln und nur in geringer Entfernung auslesen kann, sendet der RFID-Chip Daten mit Hilfe von Funkwellen.



Diese Chips werden schon in einigen Bereichen eingesetzt. So finden sich RFID-Chips in den Wegfahrsperren mancher Autos. In den Autoschlüssel ist ein Transponder integriert, der vom Auto erkannt werden muss. Seit 2005 haben wir in Deutschland Reisepässe mit RFID-Technik. Auf dem Chip sind personenbezogene Daten wie Name, Adresse und Foto gespeichert, die auch aus einiger Entfernung ausgelesen werden können, beispielsweise an Flughäfen. Im brasilianischen Bundesstaat Bahia tragen sogar die Schülerinnen und Schüler Schuluniformen mit eingewähten RFID-Chips. Schulschwänzer haben so keine Chance – sobald ein Schüler oder eine Schülerin nicht pünktlich in der Schule erscheint, werden Lehrerinnen und Lehrer, aber auch die Eltern direkt benachrichtigt.

## Aufgabe

Ihr werdet beauftragt, einen neuen Chip zu entwickeln. Teilt euch dazu in Gruppen auf mit jeweils 4 bis 6 Schülerinnen und Schülern und sucht euch einen der folgende Schwerpunkte (a, b oder c) aus:



Welche Eigenschaften müsste ein Chip haben beziehungsweise welche Informationen müssten auf ihm gespeichert sein, um ...

- Betriebsabläufe beispielsweise in Logistikunternehmen zu verbessern?
- den Einkauf im Einzelhandel/Supermarkt zu vereinfachen und zu verbessern?
- unsere Welt zu Hause als Verbraucher zu ändern? (Z.B. im Bereich Sicherheit oder Bedienung von Geräten etc.)



Überlegt Euch Personas, um einen konkreten Nutzer mit seinen Bedürfnissen und Problemen zu ermitteln. Personas sind prototypische Nutzer oder Unternehmen. Welche Wünsche könnte die Person oder das Unternehmen bezogen auf euer Produkt haben? Wo liegen Schwierigkeiten, vielleicht in der Handhabung? Lasst eure Erfahrungen und Erkenntnisse aus der Betriebsbesichtigung einfließen und recherchiert darüber hinaus im Internet. Welche Probleme gibt es beispielsweise mit Barcodes? Warum braucht man eventuell neue Lösungen und so weiter.

### Synthese

Nachdem ihr Vermutungen über die Person oder das Unternehmen angestellt habt und auch Beobachtungen aus eurem Alltag habt einfließen lassen, entwickelt ihr daraus konkrete Szenarien. Beschreibt diese in kleinen Geschichten und malt sie auf. Kommen bestimmte Probleme häufiger vor? Zeichnet sich ein Muster ab?

### Ideen sammeln

Nun werden kreative Ideen und Lösungen gesucht. Schreibt und malt alles auf, was euch dazu einfällt – etwa um Betriebsabläufe zu verbessern oder spezielle Anwendungen zu Hause zu revolutionieren. Auch verrückte Ideen sind ausdrücklich erlaubt. Danach werdet ihr mit Klebepunkten über eure Ideen abstimmen. Die Idee, die die meisten Klebepunkte erhält, wird weiter verfolgt.

### Prototyp erstellen

Bastelt einen ersten Prototypen, ihr könnt dafür alle verfügbaren Materialien verwenden: Papier, Schere, Stifte, Knete, Luftballons, ... Der Prototyp dient dazu, die Lösungsidee in der Praxis zu überprüfen. Ihr könnt eure Idee auch als Wandplakat oder Präsentation visualisieren.

### Testen

Präsentiert euren Prototypen einem anderen Team. Wird die Lösung angenommen? Gibt es Feedback? Nun zeigt sich, ob eure Idee vielleicht weiterentwickelt werden kann oder aber verworfen werden muss.

# Musterlösungen

## B3.1

### Aufgabe 2

a)



b)



### Aufgabe 3

a)



b)



c)

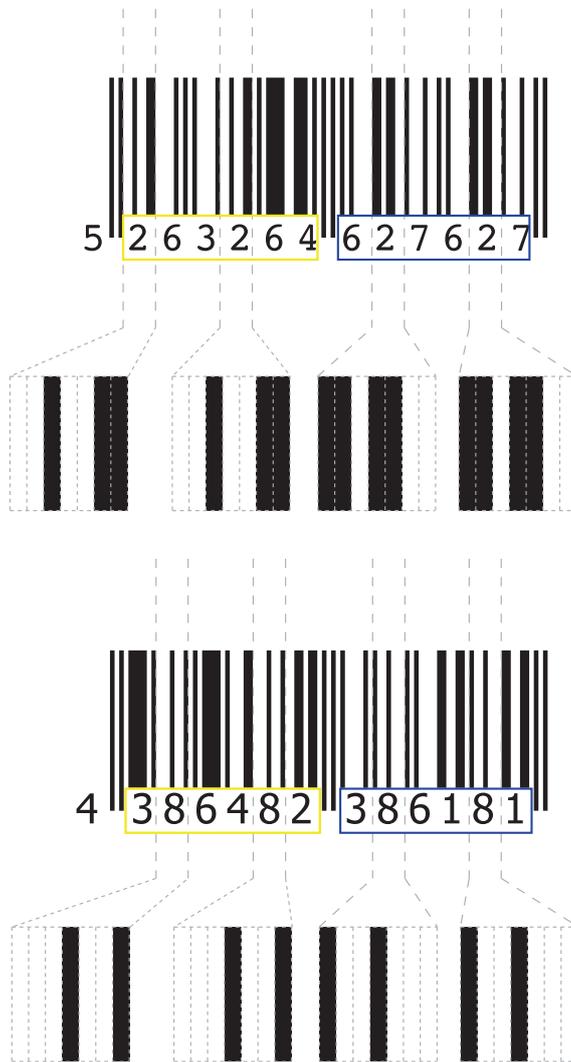


d)



## Aufgabe 4

a)



- b) Jede Ziffer wird mit 7 Strichen kodiert. Die Striche können dabei entweder schwarze Balken oder weiße Leerstellen sein.

Die kleinste Informationseinheit in der Informatik ist ein Bit. Ein Bit kann zwei Zustände darstellen: nämlich 0 (weiß bzw. Leerstelle im Barcode) oder 1 (schwarzer Balken im Barcode). Jede Ziffer des Barcodes wird also mit 7 Bit codiert / dargestellt.

Die Codierungen der linken Ziffern beginnen immer mit (mindestens) einer weißen Leerstelle, während die Ziffern auf der rechten Seite immer mit einer schwarzen Stelle anfangen. Gleichzeitig enden die Ziffern auf der linken Seite immer mit einem schwarzen Balken, die auf der rechten immer mit einer weißen Leerstelle.

- c) Im oberen Beispiel werden die Stellen der „2“, die links weiß codiert werden, rechts schwarz codiert und umgekehrt. Man spricht hierbei von einer bitweisen Invertierung.

Im unteren Beispiel hingegen sind die Balkendarstellungen der Ziffer „8“ auf der linken und rechten Seite des Barcodes „gespiegelt“. Die Spiegelachse ist dabei das mittlere Trennzeichen.

## Aufgabe 5

Die Aussage ist wahr.

## Aufgabe 6

a)

5	2	6	3	2	6	4	6	2	7	6	2	7	4	3	8	6	4	8	2	3	8	6	1	8	1
<input checked="" type="radio"/> A	<input checked="" type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	<input type="radio"/> A	
<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input checked="" type="radio"/> B	<input checked="" type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	<input type="radio"/> B	
<input type="radio"/> C	<input type="radio"/> C	<input checked="" type="radio"/> C	<input checked="" type="radio"/> C	<input type="radio"/> C	<input type="radio"/> C	<input checked="" type="radio"/> C	<input checked="" type="radio"/> C	<input checked="" type="radio"/> C																	

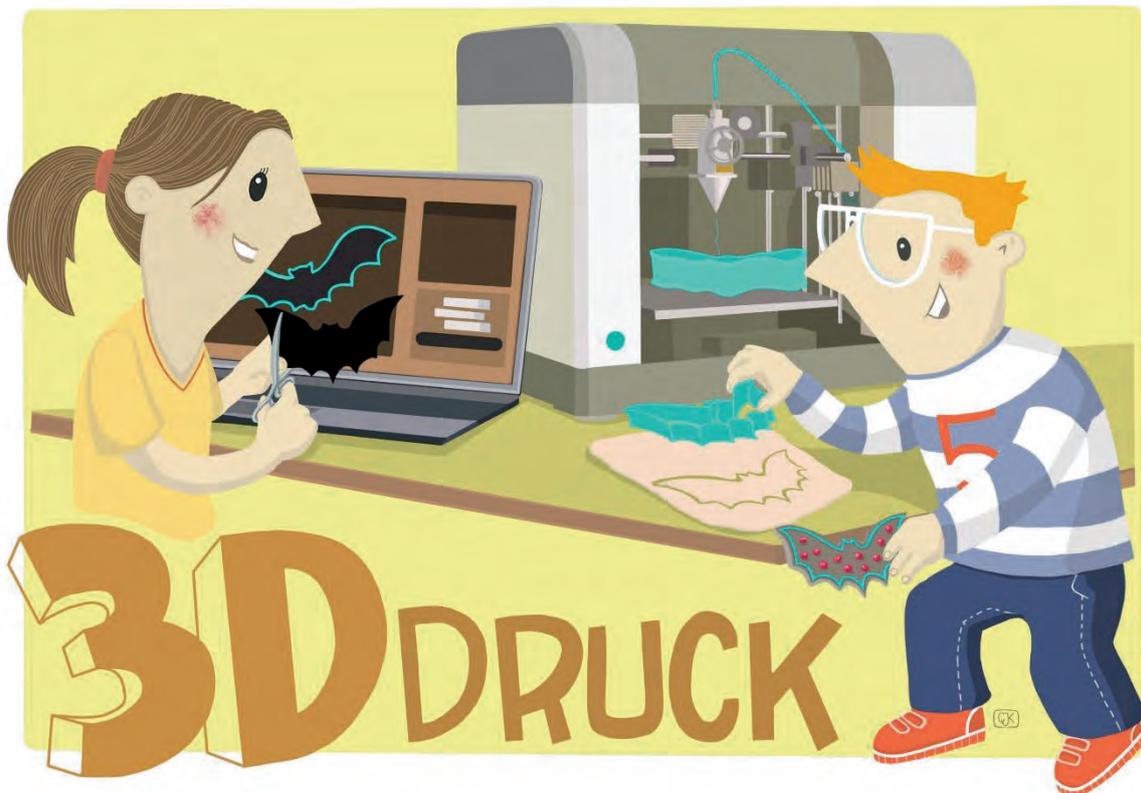
Auffällig ist, dass die rechts stehenden Ziffern immer nach Codereihe C codiert werden.

Bei den Ziffern auf der linken Seite ist kein Muster erkennbar.

- b) Durch eine Verwendung unterschiedlicher Codereihen links und rechts des Trennstriches lässt sich gewährleisten, dass der Barcode-Scanner den Code auch „falschherum“ erkennt. Oder achten KassiererInnen beim Scannen darauf, den Artikel immer richtig herum zu halten?

# IT2School

Gemeinsam IT entdecken



Eine Entwicklung von



In Kooperation mit



Im Auftrag der



## Inhalt

1	3D-Druck, Modellierung und Augmented/Virtual Reality.....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Rolle der Unternehmensvertreter*innen.....	4
5	Inhalte des Moduls.....	5
5.1	3D-Modellierung .....	5
5.2	3D-Druck.....	8
5.3	Virtual/Augmented Reality .....	11
6	Unterrichtliche Umsetzung.....	12
6.1	Grober Unterrichtsplan .....	14
6.2	Stundenverlaufsskizzen.....	16
6.2.1	Verlauf für die Grundschule.....	16
6.2.2	Verlauf für die Sek I und Sek II.....	18
7	Einbettung in verschiedene Fächer und Themen .....	22
8	Anschlussthemen.....	23
9	Literatur und Links .....	23
10	Arbeitsmaterialien .....	25
11	Glossar.....	25
12	FAQs und Feedback.....	26

# 1 3D-Druck, Modellierung und Augmented/Virtual Reality

Seit in den letzten Jahren 3D-Drucker immer günstiger geworden sind, sind sie auch für Heimanwender\*innen und Bastler\*innen sowie für den Bildungsbereich erschwinglich. Einige Schulen haben sich schon 3D-Drucker angeschafft, um im Mathematikunterricht oder im Fach Kunst vielfältige Projekte umzusetzen.



In diesem Modul möchten wir einen kleinen Einblick in die 3D-Modellierung und den 3D-Druck geben und Anregungen für die Praxis unterbreiten. Die Schülerinnen und Schüler können in Projekten ihre eigenen kreativen Ideen umsetzen und sich die eigenen Modelle in einer virtuellen Realität ansehen. Über Parametrisches Design ist auch ein kreativer Einstieg in blockbasierte Programmierung möglich.

<b>Lernfeld/Cluster:</b>	IT selbst machen	
<b>Zielgruppe/Klassenstufe:</b>	X	4. bis 5. Klasse
	X	6. bis 7. Klasse
	X	8. bis 10. Klasse
	X	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	3 – 6 Stunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Funktionsweise des 3D-Druckers verstehen</li> <li>• Sich mit den Möglichkeiten des 3D-Drucks auseinandersetzen</li> <li>• Einsatzgebiete und moderne Produktionsweisen kennenlernen</li> <li>• Eigene Formen und Figuren modellieren und drucken</li> <li>• Einen Einstieg in blockbasierte Programmierung finden</li> <li>• Grundlegende Kenntnisse über Schleifen, Variablen, Verzweigungen und Methoden kennenlernen und bei der Gestaltung von parametrischen Designs anwenden können</li> </ul>	
<b>Vorkenntnisse der Schüler*innen:</b>	Keine	
<b>Vorkenntnisse der Lehrkraft:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Kenntnisse im 3D-Druck (falls er selbst bedient wird)</li> </ul>	
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Kenntnisse im 3D-Druck (falls er selbst bedient wird)</li> </ul>	
<b>Sonstige Voraussetzungen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Internet-Zugang</li> <li>• Zugang zu einem 3D-Drucker (dies kann auch über Anbieter im Internet oder ein Partnerunternehmen erfolgen)</li> </ul>	



## 2 Warum gibt es das Modul?

In vielen Bereichen der Industrie und der Wissenschaft ist die Erstellung von Prototypen und Produkten durch 3D-Drucker schon fest etabliert. Durch immer niedrigere Kosten sowie durch die Maker-Bewegung<sup>1</sup> wurde die Bekanntheit und auch die Zugänglichkeit von 3D-Druckern vorangetrieben. Viele Maker sind in sogenannten Makerspaces oder FabLabs (fabrication laboratories) organisiert und bilden lokale Anziehungspunkte in vielen Städten rund um den Globus. Damit ist die industrielle Produktion für jedermann möglich.

Aufgrund vielfältiger Einsatzbereiche, insbesondere in Wirtschaft und Wissenschaft, wird es in der Zukunft immer wichtiger, diese neuartigen Technologien zu verstehen und anzuwenden. Auch für die Schule eröffnet das Modellieren und 3D-Drucken vielfältige Potentiale. Zum einen beinhaltet die Thematik die Bereiche Mathematik, Informatik, Technik, Kunst und Kreativität, wodurch sich vielfältige Einsatzszenarien in den entsprechenden Fächern ergeben, aber auch Projektwochen, AGs und der Einsatz in Schüler\*innenfirmen sind denkbar.

Zum anderen üben moderne Technologien wie 3D-Drucker eine hohe Faszination auf Schülerinnen und Schüler aus, wodurch die Motivation beim Lernen erfahrungsgemäß erhöht ist. Schülerinnen und Schülern erschließen sich einen Lernraum, in dem sie die Möglichkeit haben, die digitale Welt zu erkunden und mitzugestalten. Dies ist auch aus lerntheoretischer Sicht von Vorteil. Vertreterinnen und Vertreter des Konstruktivismus und insbesondere dessen Weiterentwicklung durch Seymour Papert (Konstruktivismus) sehen Lernen als konstruktiven Prozess. Lernen wird durch das Selbermachen ermöglicht und schult dabei Problemlösekompetenzen, kommunikative Fähigkeiten und Teamarbeit.

## 3 Ziele des Moduls

- Funktionsweise des 3D-Druckers verstehen
- Sich mit den Möglichkeiten des 3D-Drucks auseinandersetzen
- Einsatzgebiete und moderne Produktionsweisen kennenlernen
- Eigene Formen und Figuren modellieren und drucken
- Grundlegende Programmierkenntnisse über Schleifen, Variablen, Verzweigungen und Methoden kennenlernen und bei der Gestaltung von parametrischen Designs anwenden können

## 4 Rolle der Unternehmensvertreter\*innen

Im *Modul B4 – 3D-Druck* hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten aktiv mitzuwirken. Hier einige Anregungen:

- Unterstützung der Lehrkraft - Co-Teacher: Der\*Die Unternehmensvertreter\*in kann eine Einführung in die 3D-Modellierung geben und die Lehrkraft direkt im Unterricht unterstützen.

---

<sup>1</sup> Bewegung des Selbermachens (Do-It-Yourself – DIY), auch mit digitalen Mitteln

- Druckmöglichkeit bieten: Bisher haben nur wenige Schulen einen eigenen 3D-Drucker, daher könnte man 3D-Modelle der Schülerinnen und Schüler im eigenen Unternehmen drucken.
- Gastgeber\*in: Die Schulklasse könnte eingeladen werden, um sich die 3D-Drucker anzusehen und kleine Projekte, wie beispielsweise mit dem Cookie Caster oder den Einstieg in BlocksCAD selbst umzusetzen.
- Bericht aus dem Unternehmen – Special-Guest: Man könnte über die Bedeutung von 3D-Druckern in der Industrie berichten, insbesondere auch über die Einsatzgebiete im eigenen Unternehmen.

## 5 Inhalte des Moduls

### 5.1 3D-Modellierung

Mit 3D-Druckern kann man mittlerweile verschiedenste Materialien in Form bringen, ob Lebensmittel wie Pasta oder Schokolade, Metalle oder Kunststoffe. Es werden Ersatzteile und Prototypen gedruckt, Prothesen aus Titan, Bauteile für Flugzeuge und vieles mehr. Gedruckt wird mit Hilfe eines additiven Verfahrens, d.h. dass das Druckmaterial in dünnen Schichten aufgetragen wird und Schicht für Schicht ein dreidimensionales Objekt entsteht. Je nach Material werden verschiedene Verfahren angewendet. Ein gängiges Verfahren ist das *Fused Deposition Modeling (FDM)*. Hierbei wird ein Kunststoff-Filament durch eine beheizte Düse geleitet und geschmolzen. Die einzelnen Schichten härten sofort an der Luft aus.

Bevor man mit dem 3D-Drucker Objekte drucken kann, benötigt man eine geeignete Vorlage. Dafür gibt es mittlerweile 3D-Scanner, die Objekte wie beispielweise Tassen oder auch Menschen abscannen und dann daraus eine Druckvorlage erstellen.



Mit Hilfe von geeigneter Software können eigene Modelle erstellt werden. Für Kinder in der Grundschule eignet sich zum Einstieg besonders der *Cookie Caster*. Mit dessen Hilfe können 2D-Zeichnungen in 3D-Keksaustechförmchen verwandelt werden. Erläuterungen zu diesem Tool finden sich im Lehrkräfte-Material B4.1.

Zur Vertiefung eignet sich das Online-Tool *Tinkercad*. Die Software ist übersichtlich aufgebaut und bietet eine Reihe von Grundformen (Quader, Zylinder, Pyramide etc.) mit denen man schnell neue eigene Modelle konstruieren kann. Durch die vorgegebenen Grundformen eignet sich das Programm auch für die Grundschule. Das Werkstück wird am Ende in den Rapid-Prototyping-Standard STL exportiert, um es an einen 3D-Drucker zu senden.

**Hinweis:** Um Tinkercad nutzen zu können, ist das Anlegen eines Zugangs erforderlich. Die Anmeldung kann zum einen über einen bestehenden Google-, Apple-, Microsoft- oder Facebook-Account erfolgen, *oder* über die Angabe einer gültigen E-Mail-Adresse, des Geburtstages und eines Benutzernamens bei der Registrierung vollzogen werden. Die Lehrkraft kann jedoch zur Steigerung der Datensicherheit und zur Moderation der Schülerinnen und Schüler auch sogenannte „**Klassen**“ einrichten. Dieses Vorgehen ist bei der unterrichtlichen Verwendung von Tinkercad ausdrücklich zu empfehlen. Das Einrichten eines Lehrkräfte-Zugangs und das Anlegen von Klassen in Tinkercad werden im Folgenden dargestellt:

1. Öffnen Sie die Seite <https://www.tinkercad.com/join> und klicken Sie auf „Lehrkräfte beginnen hier“.
2. Registrieren Sie sich im Folgenden mit einer von Ihnen erreichbaren E-Mail-Adresse.

Ggf. müssen Sie Ihren Status als Lehrkraft bestätigen. Klicken Sie dazu auf „Start now“ unter „Are you a teacher?“ in der E-Mail, die Sie nach der Registrierung (Schritt 2) erhalten haben und melden Sie sich mit Ihren soeben hinterlegten Nutzerdaten an.

3. Klicken Sie in der oberen Statuszeile auf „Lehren“.



4. Auf der sich dann öffnenden Seite (<https://www.tinkercad.com/teach>) finden Sie unter „Erste Schritte“ einen Abschnitt mit dem Titel „Manage your classroom in Tinkercad“. Klicken Sie hierin auf „Add your students through Tinkercad Classrooms“.



5. Auf der sich im Folgenden öffnenden Seite (<https://www.tinkercad.com/classrooms>) werden die von Ihnen angelegten Klassen angezeigt. Über die Schaltfläche „Neue Klasse erstellen“ können Sie eine neue Klasse anlegen. Wählen Sie dabei bestenfalls aussagekräftige Attribute.

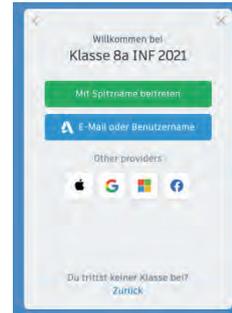
Neue Klasse	
Klassenzimmer-Name	Klasse 8a INF 2021
Klassenstufe	B-9 Jahre / 3. Klasse
Thema	Informatik
<input type="button" value="Abbrechen"/> <input type="button" value="Klasse erstellen"/>	

6. Wählen Sie nun die von Ihnen angelegte Klasse aus der Liste aus. In der sich daraufhin öffnenden Ansicht können Sie Schülerinnen und Schüler hinzufügen. Hier genügt eine Angabe von Vornamen. Das Tool legt automatisch Spitznamen an, die üblicherweise den gewählten Vornamen um einen numerischen Code ergänzen.

Klasse 8a INF 2021						
Schüler Entwürfe Aktivität Co-Lehrer Sicherer Modus						
Schüler hinzufügen		Klassencode	Aktion auswählen -		Class code: NIUG-B7NM-3KQW	Nach Name suchen
	Schüler	Anmeldedaten	Typ	Aktivität	Sicher	Menu
<input type="checkbox"/>	Axel	axel5506	Seat		<input type="checkbox"/>	...
<input type="checkbox"/>	Markus	markus1915	Seat		<input type="checkbox"/>	...
<input type="checkbox"/>	Katharina	katharina2614	Seat		<input type="checkbox"/>	...
<input type="checkbox"/>	Ira	ira9432	Seat		<input type="checkbox"/>	...
<input type="checkbox"/>	Tolij	tolij0669	Seat		<input type="checkbox"/>	...
<input type="checkbox"/>	Nils	nil5465	Seat		<input type="checkbox"/>	...



7. Über einen Klick auf „Klassencode“ können Sie einen Link zu Ihrem Tinkercad-Klassenraum erzeugen, den Sie dann mit Ihrer Klasse teilen können. Teilen Sie Ihren Schülerinnen und Schüler ihre individuellen Spitznamen mit.
8. Wenn Ihre Schülerinnen und Schüler diesen Link aufrufen, erscheint ihnen eine Ansicht im Stil des rechts abgebildeten Screenshots. Dort müssen Ihre Schülerinnen und Schüler die Schaltfläche „Mit Spitzname beitreten“ wählen und ihre individuellen Spitznamen eingeben.
9. Nun verfügen Ihre Schülerinnen und Schüler über einen eingeschränkten Zugang zu Tinkercad. Sie können eigene Entwürfe anlegen, diese aber aus datenschutzrechtlichen Gründen nicht online teilen. Ein Export der Modelle (bspw. im STL-Format) ist Ihren Schülerinnen und Schülern aber natürlich dennoch möglich.

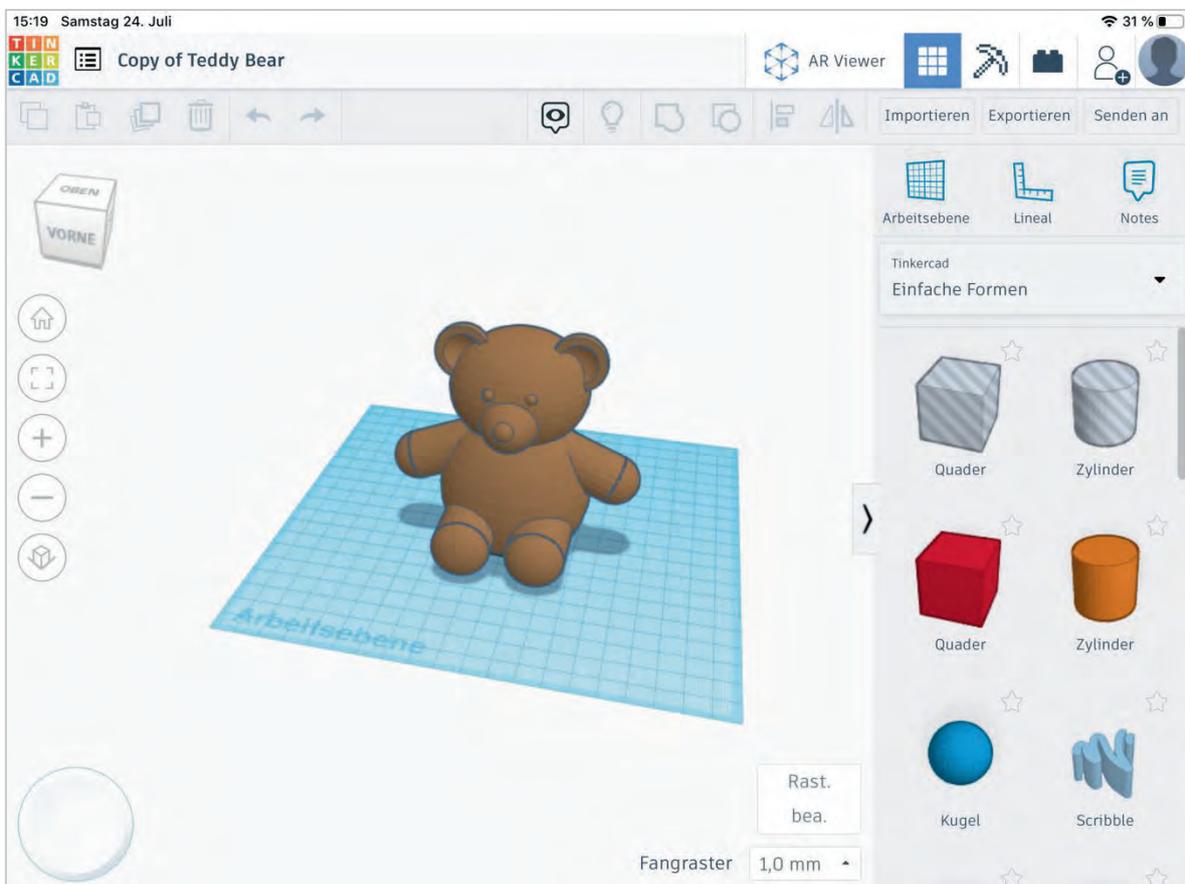


Außerdem ist von Tinkercad eine App für das **iPad** verfügbar (<https://apps.apple.com/us/app/tinkercad/id1469440830>).



Ihre Schülerinnen und Schüler können sich auch in der iPad-Version entweder über eigenständige Zugangsdaten oder über die zuvor beschriebene „Klassen“-Funktion (*empfohlen*) anmelden.

Eigene Projekte können in der Browser-Version gestartet und auf dem iPad fortgesetzt werden (oder umgekehrt). Die Bedienung der iPad-App ist sehr nah an der Nutzung der browser-basierten Version.



*SketchUp Make* bzw. *SketchUp Free* ist ein weiteres kostenloses Programm, um eigene Modelle zu konstruieren. Das Programm wurde ursprünglich von Google entwickelt und bietet auch für Anfänger/innen einen guten Einstieg in die 3D-Modellierung, dabei enthält es aber wesentlich mehr Gestaltungsmöglichkeiten als Tinkercad.

Die erstellten Modelle in SketchUp können im Anschluss als Augmented Reality oder Virtual Reality betrachtet werden. Hierfür wird eine 3D-Brille aus Pappe für das Smartphone gebastelt.

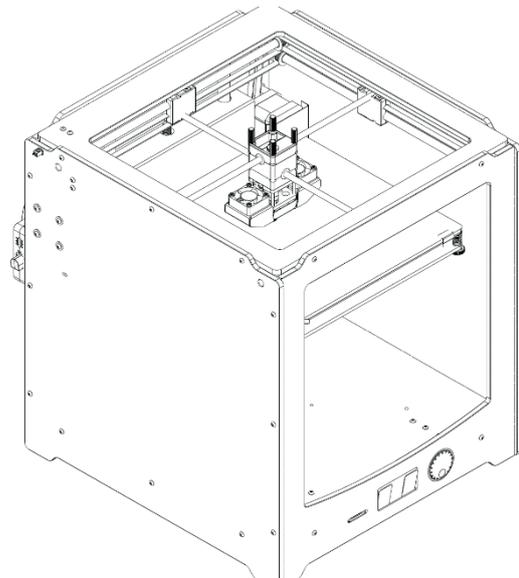
## 5.2 3D-Druck

Mit dem Begriff „3D-Drucken“ bezeichnet man Fertigungsverfahren von dreidimensionalen Werkstücken, die durch das schichtweise Auftragen von Material entstehen.

Mittlerweile gibt es eine Vielzahl von Verfahren, die zur additiven Fertigung genutzt werden können, wobei nur wenige für den Einsatz in der Schule geeignet sind. Beim sogenannten DLP-Verfahren wird eine harzartige Flüssigkeit, das sogenannte Resin, gezielt mit UV-Licht ausgehärtet. Analog dazu funktioniert das SLA-Verfahren bei dem anstelle einer UV-Lichtquelle ein gezielter Laser zum Einsatz kommt. Obwohl sich mit den Verfahren feine und präzise Modelle mit einer Schichthöhe von 0,01mm fertigen lassen und Einsteigergeräte bereits für unter 200€ verfügbar sind, wird pauschal der Einsatz in Schulen aus gesundheitlichen Gründen nicht empfohlen. Das Resin kann zu Reizungen der (Schleim-) und Augen führen, ist giftig für die Umwelt und muss dementsprechend fachgerecht entsorgt werden. Meistens ist der Einsatz von Lösungsmitteln zur Reinigung erforderlich. Falls dennoch mit diesem Verfahren gearbeitet werden soll, ist unbedingt eine Schutzausrüstung bestehend aus Laborkittel, Schutzbrille und Nitril-Handschuhen empfohlen. Zudem darf nur in gut ventilierten Räumen gedruckt werden.

Geeigneter für den Einsatz in der Schule ist daher das FFF- (oder FDM-)Verfahren. Dabei wird Kunststoff geschmolzen und durch diese feine Düse auf einem Druckbett aufgetragen. Einsteigermodelle beginnen als Bausatz bereits ab 130€, erfordern aber entsprechendes Know-How und Erfahrung für eine zuverlässige Fertigung. FFF-Drucker unterscheiden sich nicht nur in ihrer Größe und ihrem Bauraum, sondern auch in der Funktionsweise der Bewegungs-Steuerung. Manche Drucker haben ein festes Druckbett und ein Druckkopf, welcher sich in X, Y, Z Richtung bewegt (z.B. Voron 2), andere Drucker haben ein Druckbett, welches sich in Y-Richtung bewegt und einen Druckkopf mit Bewegung in X und Z Richtung (z.B. Prusa i3) und wiederum andere Drucker verfügen über ein Druckkopf mit Bewegung in XY-Richtung und einem Druckbett in Z-Richtung (z.B. Drucker aus der Ultimaker-Reihe). Besonders interessant anzusehen sind sogenannte Delta-Drucker. Jeder Stil hat seine eigenen Vor- und Nachteile, die jedoch für den schulischen Einsatz vernachlässigbar sind.

In der rechten Abbildung ist ein Drucker aus der Ultimaker Reihe abgebildet. Ein solcher Drucker besteht aus vier Schrittmotoren (zwei für die XY Bewegung des Druckkopfes, einer für die Z-Bewegung der Druckplatte, einer für die Förderung des Materials). Der Druckkopf setzt sich aus mehreren Lüftern, einer Halterung und einem Hotend zusammen. Die Lüfter an den Seiten sind Bauteillüfter. Diese kühlen das frisch aufgetragene Material ab, damit dieses aushärtet und so formstabil bleibt. Ein weiterer Lüfter befindet sich am Hotend und verhindert, dass das Material zu weit oberhalb der Düse (Nozzle) weich wird und eine Verstopfung verursacht. Das Hotend verfügt am unteren Ende über einen Heizblock. Dort befindet sich ein Heizelement, ein Temperaturfühler und eine Düse, aus welcher das Druck-Material extrudiert wird. Der gesamte Druckkopf wird entlang zweier Präzisionswellen bewegt, welche wiederum durch Schlitten, die mit Riemen und Rollen an den Schrittmotoren befestigt sind, bewegt werden. Auf der Rückseite des Druckers ist der Extruder zu erkennen. Dieser greift das Filament und bewegt es präzise entlang eines Schlauches (Bowdentube genannt) zum Druckkopf. Der Schlauch ist zur Restriktion des Filaments auf dem Weg zum Hotend zwingend notwendig und sollte an beiden Enden (Extruder und Hotend) fest fixiert sein.



Das Filament wird auf dem Druckbett aufgetragen. Beliebte Druckflächen sind Glas oder Federstahlblech mit PEI Beschichtung. Nach jeder aufgetragenen Schicht bewegt der Z-Motor das Druckbett mit Hilfe einer Gewindestange um einen festen Wert weiter nach unten. Typische Schichthöhen betragen bei einer 0,4mm Düse 0,15 bis 0,3mm. Das Auftragen der Schichten und Absenken des Druckbettes wird so lange wiederholt, bis der Druck abgeschlossen ist. Je nach 3D-Modell, Drucker und Druckeinstellungen (Vorheizen der Düse, Druckbett und ggf. Druckraum, Schichthöhe, Wandstärke, Füllung, Stützstrukturen, Druckgeschwindigkeit, ...) kann ein Druck zwischen wenige Minuten und mehreren Tagen dauern. Der Dinosaurier aus dem Arbeitsmaterial B4.9 wurde beispielsweise in zwei Stunden gedruckt.

Beliebte Materialien sind PLA, PETG, ABS und Nylon. Da ABS und Nylon vergleichsweise hohe Temperaturen (250°C Nozzle, >80°C Druckbett) und idealerweise eine geschlossene Einhausung mit einer Lufttemperatur über 40°C benötigen, können diese Kunststoffe nicht mit jedem Drucker zuverlässig gefertigt werden. Zusätzlich strömt ABS beim Drucken Dämpfe aus, die Benzol und Styrol enthalten. Für den schulischen Kontext sollte daher auf diese Materialien verzichtet werden. PETG und PLA lassen sich mit niedrigeren Temperaturen (<230°C) drucken, benötigen keine extra Einhausung und sondern nach aktuellem Kenntnisstand beim Drucken keine giftigen Dämpfe ab. PETG hat bessere mechanische Eigenschaften als PLA, neigt aber dazu ungewollte Fäden im Druck zu ziehen. Um die Fadenbildung zu reduzieren, sollte das Material immer trocken gelagert und ab und zu getrocknet werden. Direkte Sonneneinstrahlung über längere Zeit ist zu vermeiden. Zusätzlich sollte vor dem Kauf darauf geachtet werden, dass das Filament den korrekten Durchmesser hat. Je nach Drucker ist ein Durchmesser von 2,85mm oder 1,75mm erforderlich.

Um einen Drucker in Betrieb zu nehmen, wird neben dem Filament auch ein Slicing Programm benötigt. Das Programm erzeugt aus dem 3D-Modell, welches üblicherweise im STL oder 3MF Format abgespeichert wird, eine G-code Datei. Als G-code bezeichnet man Steuerbefehle für

CNC-Maschinen. Mittlerweile gibt es mehrere gute, kostenfreie Slicing Programme (Cura, PrusaSlicer, SuperSlicer, ...) mit vorgefertigten Profilen für alle gängigen 3D-Drucker.

Im Slicing Programm können Druckparameter, wie Schichthöhe, Füllung, Druckgeschwindigkeit oder Stützstrukturen eingestellt werden. Üblich ist eine Schichthöhe von 0,2mm bei einer 0,4mm Düse. Für nicht funktionale Teile reicht eine Wandstärke aus 2 Schichten und einer Füllung von 10% aus. Funktionale Teile sollten mindestens eine 4-schichtige Wand und eine Füllung über 40% besitzen. Die erzeugte G-Code Datei kann dann exportiert werden. Je nach Drucker ist eine Speicherkarte oder USB-Stick erforderlich. Manche Drucker können die Datei aber auch via WiFi direkt aus dem Slicer empfangen. Mit dem Bedienfeld auf dem Drucker oder einer Weboberfläche kann der Druck gestartet werden.

Tipps im Umgang mit 3D-Druckern in der Schule:

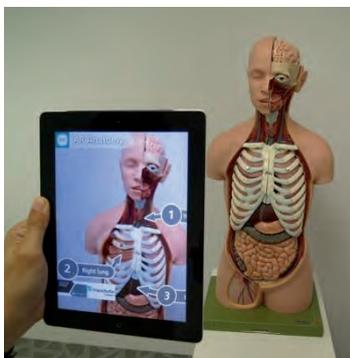
Fehler	Problem	Lösung
Ungeeigneten G-code erzeugt	Jeder Drucker hat spezifische Hardware, Größen und kann mit verschiedenen Materialien geladen werden. Daher muss G-code <u>speziell</u> für die entsprechende Maschine mit dem entsprechenden Material erzeugt werden. Wenn zum Beispiel ABS mit PLA Temperaturen gedruckt wird, dann funktioniert der Druck im besten Fall nicht. Im schlimmsten Fall verstopft die Düse, der Extruder frisst sich ins Filament oder es entsteht ein „Blob of Death“.	Einen extra PC mit voreingestellten Druck- und Drucker Profilen anbieten  Nur eine Filamentsorte pro Drucker anbieten. Kein mitgebrachtes Filament akzeptieren.
3D-Drucker unbeaufsichtigt gelassen	Vor allem die aller erste Schicht ist kritisch für einen erfolgreichen Druck. Zeichnet sich bereits in der ersten Schicht beispielsweise eine schlechte Haftung ab, dann ist es ratsam den Druck sofort abbrechen. Der Druck kann sich sonst vom Druckbett lösen. Im besten Fall hat man ein Druckbett voller Kunststoff <b>Spaghetti</b> ; im schlimmsten Fall den „Blob of Death“. In beiden Fällen ist der Druck aber misslungen.	Immer die erste Schicht beobachten und bei Problemen abbrechen. Danach regelmäßigen Abständen den Zustand des Druckers prüfen.  Keinen Druck unter Zeitdruck starten.
3D-Druck haftet nicht	Siehe oben	Druckbett von fettigen Fingerabdrücken befreien.  Druckbett entstauben.  Druckbett neu ausrichten.

		Klebestift können helfen, aber dieser sollte nur sparsam verwendet werden
3D-Druck haftet zu gut	Wenn der Druck zu gut haftet, neigen SuS dazu den Druck mit Gewalt zu entfernen. Als mögliche Folge muss das Druckbett oder die Achsen neu kalibriert werden.	Abstand der Düse zum Druckbett anpassen  Entfernbarer Magnetdruckplatte verwenden. Nach dem Druck lässt man die Platte etwas abkühlen. Danach kann diese vom Druckbett entfernt werden und leicht gebogen werden. So löst sich der Druck.
Unqualifizierte Reparatur	3D-Drucker sind Maschinen, die Wartung benötigen. Diese Wartungen, wie z.B. das Tauschen der Nozzle, Freimachen eines verstopften Hotends, Einfetten der Linearführungen oder ähnliches sollte nur von qualifizierten Personen durchgeführt werden, da womöglich beim Versuch einer Reparatur Teile des Druckers zu Schaden kommen könnten. Stichwort: <a href="#">Verschlimmbessern</a>	Bei Problemen immer selbst kümmern

### 5.3 Virtual/Augmented Reality

Die eigenen Modelle lassen sich nicht nur 3D-drucken und dann haptisch erfahren, sondern mit Hilfe aktueller Smartphones/Tablets und ggf. weiteren Tools auch virtuell in die eigene Realität einfügen.

Dabei wird grundsätzlich zwischen zwei verschiedenen Technologien unterschieden, die oft miteinander gleichgesetzt werden:

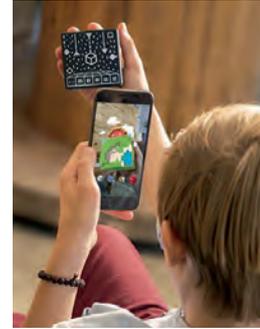


**Augmented Reality** (kurz AR, deutsch „erweiterte Realität“) bezeichnet die computergestützte Erweiterung der Realität. Zumeist erfolgt diese auf visueller Ebene. Bei AR wird auf Bild- oder Videodaten zurückgegriffen und diese werden mit computergenerierten Zusatzinformationen ergänzt.

Zunächst insbesondere für den militärischen Einsatz konzipiert (bspw. Erweiterung der Ansicht von Kampjet-Pilotinnen und -Piloten durch in den Helm integrierte Systeme), kennen wir AR mittlerweile in verschiedensten Anwendungssituationen: Beim Schauen von Fußballspielen, wenn in Spielanalysen Informationen zu Sprintgeschwindigkeiten ergänzt oder virtuelle Abseitslinien gezogen werden beispielsweise. Beim Online-Shopping nach Möbeln kommen entsprechende Technologien bei der Möglichkeit, diese als Modell direkt im heimischen

Wohnzimmer zu platzieren, ebenfalls zum Einsatz. Auch in medizinischen und in Bildungszusammenhängen (siehe Abb. oben links) kommen vermehrt AR-Technologien zum Einsatz

Eine Ergänzung der Umwelt um eigenständig gestaltete 3D-Modelle fällt ebenfalls unter AR. Im Zusatzmaterial B4.6 wird dargestellt, wie sich hierzu der *MergeCube* nutzen lässt (siehe rechte Abb.).



Bildquelle: Merge Labs, Inc., 2019



In einer virtuellen Realität (**Virtual Reality**, VR) wird hingegen gänzlich in eine computergenerierte Welt „eingetaucht“. Gängiges Mittel dazu sind sogenannte VR-Brillen (siehe linke Abb.).

Eine verhältnismäßige kostengünstige Alternative zu entsprechenden Head-Mounted-Displays (HMD, also auf dem Kopf getragenen Bildschirmen) greift auf ein Informatiksystem zurück, das die meisten von uns aus ihrer Hosentasche kennen: Das eigene Smartphone. Passend geformte Halterungen (bspw. aus Karton) mit zwei Sammellinsen ermöglichen so in darauf ausgelegten Apps

eine stereoskopische Ansicht von Inhalten, die einen räumlichen Eindruck vermitteln.



Bildquelle: [https://commons.wikimedia.org/wiki/File:Assembled\\_Google\\_Cardboard\\_VR\\_mount.jpg](https://commons.wikimedia.org/wiki/File:Assembled_Google_Cardboard_VR_mount.jpg)

Im Zusatzmaterial B4.6 wird die Verwendung des *Google Cardboards* vorgeschlagen und dargestellt.

## 6 Unterrichtliche Umsetzung

Für den Einsatz des 3D-Druckers in der Schule gibt es zahlreiche Ideen. Viele Einsatzszenarien eignen sich auch für fächerübergreifendes Arbeiten. Im Folgenden werden einige Möglichkeiten für den Einsatz in Schule und Unterricht beispielhaft genannt:

### Mathematik

Im Mathematikunterricht können geometrische Formen modelliert und ausgedruckt werden. Auch können Wahrzeichen oder Häuser der Stadt, beispielsweise die eigene Schule vermessen und maßstabsgetreu modelliert und ausgedruckt werden.

### Kunst

Im Rahmen des Arbeitsbereichs Plastik/ Objekt oder auch Design und Architektur können sich die Schülerinnen und Schüler mit dreidimensionalen Gestaltungen auseinandersetzen. Kunstvolle Objekte können designt oder auch Häuser samt Inneneinrichtung oder ganze Städte entworfen und gedruckt werden.

### Chemie/Biologie

In Biologie oder auch Chemie können Anschauungsobjekte modelliert werden, beispielsweise Moleküle, Zellen oder Organe.

## **Informatik/Technik**

Im Bereich Informatik, IT und Technik können die Schülerinnen und Schüler ihre Kenntnisse und Fertigkeiten im Gebrauch des Computers als Werkzeug mit vielfältigsten Einsatzmöglichkeiten entdecken. Die Schülerinnen und Schüler können etwas über die Funktionsweise sowie die Komponenten eines 3D-Druckers erfahren, auch ist denkbar einen eigenen 3D-Drucker zusammen zu bauen. Mit dem in diesem Modul vorgestellten Parametric Design Tool *BlocksCAD* lässt sich darüber hinaus ein kreativer ProgrammierEinstieg finden, in dem grundlegende Programmierkonzepte wie Schleifen, Verzweigungen, Variablen und Prozeduren behandelt werden können.

## **Wirtschaft**

Die Schülerinnen und Schüler können alle Schritte einer Produktion, von der Idee über die erste Zeichnung eines Prototyps, über die digitale Modellierung bis zum fertigen Produkt durchlaufen.

## **Politik/Sozialkunde**

Die Schülerinnen und Schüler können sich im Sozialkundeunterricht mit den sozialen Auswirkungen moderner Produktionstechniken für jedermann auseinandersetzen. Dabei kann die Marktbewegung genauso thematisiert werden wie das Upcycling, um ein Verständnis für die gesellschaftliche Dimension der Digitalisierung insbesondere des 3D-Drucks zu bekommen.

## **Schüler\*innenfirma**

Im Rahmen von Schüler\*innenfirmen haben Schülerinnen und Schüler die Möglichkeit, Produkte zu erstellen und zu verkaufen, beispielsweise Schlüsselanhänger etc. Auch Auftragsarbeiten können angenommen und verwirklicht werden.

## **Kreative Projektideen/Projektwoche**

Im Rahmen von Projektwochen, AGs oder Workshops können beispielsweise Brettspiele mit Figuren aus dem 3D-Drucker entwickelt werden, Darsteller für Trickfilme produziert oder Schmuckstücke designt werden.

## 6.1 Grober Unterrichtsplan

<b>Grundschule: Cookie Caster 2.0/ Cookie CAD und Tinkercad</b>	
<b>Unterrichtsszenario</b>	<b>Kurze Zusammenfassung</b>
Einstieg	Die Schülerinnen und Schüler designen mit schwarzem Papier/Tonkarton und Stift Formen für Keksausstecher.
Vertiefung	Die Form aus schwarzem Tonkarton wird ausgeschnitten und auf weißem Grund fotografiert (oder eingescannt). Das Foto wird auf den PC übertragen und bei CookieCAD hochgeladen Die gewünschte Form wird nun mit Hilfe des CookieCAD erstellt und anschließend gedruckt
Abschluss	Gemeinsames Plätzchen backen mit den eigenen Ausstechförmchen
ab 4. Klasse ggf. Vertiefung	Nach dem ersten Einstieg mit dem Cookie Caster, besteht die Möglichkeit ab der 4. Klasse mit dem Online-Tool Tinkercad eigene einfache 3D-Modelle zu konstruieren.
Abschluss	Basteln einer 3D-Pappbrille, Betrachten der erstellten 3D-Modelle in der Virtuellen Realität

<b>Sek I und Sek II: Cookie Caster 2.0/CookieCAD, Tinkercad und SketchUp</b>	
<b>Unterrichtsszenario</b>	<b>Kurze Zusammenfassung</b>
Einstieg	Einstieg mit der Online-Software Cookie Caster oder CookieCAD erste schnelle Ergebnisse können produziert werden
Einstieg	Einführung in Tinkercad oder SketchUp, erste Übungen mit der Software
Vertiefung	Umsetzung eines eigenen Projekts, die Schülerinnen und Schüler haben die Möglichkeit, eigene kreative Ideen umzusetzen, ggf. mit Bezug zum entsprechenden Fach <ol style="list-style-type: none"> <li>1. Ideenphase/ Brainstorming oder Methode Design Thinking, erste Modellierung mit Knete oder Bausteinen/ Lego</li> <li>2. Vorstellung in der Gruppe, Feedback geben, ggf. Modell überarbeiten</li> <li>3. Modellierung mit Hilfe einer CAD-Software, z.B. SketchUp Make</li> <li>4. Funktionsweise eines 3D-Druckers kennenlernen</li> <li>5. Prototypen ausdrucken</li> <li>6. Präsentieren</li> </ol>
Abschluss	Basteln einer 3D-Pappbrille, Betrachten der erstellten 3D-Modelle in der Virtuellen Realität

<b>Sek I und Sek II: BlocksCAD</b>	
<b>Unterrichtsszenario</b>	<b>Kurze Zusammenfassung</b>
Einstieg	Einstieg in die Oberfläche von BlocksCAD durch die*den Lehrende*n
Erarbeitung	Erste Übungen mit BlocksCAD mithilfe des Arbeitsblattes 4.7 und der Hilfekarten 4.8. Erstellen eines dreidimensionalen Blumen-Modells
Vertiefung	Umsetzung eines eigenen Projekts, die Schülerinnen und Schüler haben die Möglichkeit, eigene kreative Ideen umzusetzen, ggf. mit Bezug zum entsprechenden Fach. Die einzelnen Phasen können sich dabei an den im vorangegangenen Szenario vorgestellten Abschnitten orientieren. Beide Szenarien unterscheiden sich lediglich darin, dass hier am Ende eine parametrisch gestaltete Form gestaltet werden sollte.
Abschluss	Basteln einer 3D-Pappbrille, Betrachten der erstellten 3D-Modelle in der Virtuellen Realität

## 6.2 Stundenverlaufsskizzen

### Abkürzungen/Legende

AB = Arbeitsblatt; L = Lehrkraft; MuM = Mitschüler\*innen; SuS = Schüler\*innen; UV = Unternehmensvertreter\*in

#### 6.2.1 Verlauf für die Grundschule

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
5 -10 min	Einstieg	Plenum	Einstiegsfrage: Wer hat schon mal von 3D-Druckern gehört? Was heißt denn 3D? Film-Tipp: Sendung mit der Maus – 3D-Druck: <a href="https://www.wdrmaus.de/filme/sachgeschichten/3d_druck.php5">https://www.wdrmaus.de/filme/sachgeschichten/3d_druck.php5</a> Erklärung des Vorhabens: Keksausstecher selbst designen und drucken mit einem 3D-Drucker. Zur Veranschaulichung können Keksausstecher gezeigt werden.	Laptop, Beamer oder 3D-Drucker, Keksausstecher
30 min	Vertiefung	Einzelarbeit oder Tandemarbeit	Die Schülerinnen und Schüler designen ihre Ausstecher: <b>Schritt 1:</b> Mit Zettel und Stift werden Formen gezeichnet. <b>Schritt 2:</b> Die Formen werden auf schwarzen Tonkarton übertragen und aus-geschnitten. <b>Schritt 3:</b> Die Formen aus schwarzem Tonpapier werden auf weißem Grund abfotografiert oder eingescannt.	B4.1 Lehrkraft B4.2 Arbeitsanweisung für SuS, Papier, Stifte, schwarzer Tonkarton, Scheren, Fotokamera
10 min	Vertiefung	Einzel- oder Tandemarbeit	Das Foto wird auf den PC übertragen und bei CookieCAD hochgeladen Die gewünschte Form wird nun mit Hilfe von CookieCAD erstellt und anschließend gedruckt <a href="https://cookiecad.com/">https://cookiecad.com/</a> Alternativ können die Formen auch direkt im Cookie Caster designt werden.	PC, 3D-Drucker, Internet
120 min	Abschluss	Plenum	Gemeinschaftliches Plätzchen backen mit den selbst designten Keksausstechern; während des Backens kann das additive Verfahren des 3D-Druckers mit Hilfe von Zuckerguss nachempfunden werden.	Rezept für Plätzchen, Backzutaten, Keksausstecher

### Vertiefung ab der 4. Klasse

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
30 min	Einstieg, Ideenphase	Tandemarbeit	Nach dem ersten Einstieg mit dem <i>Cookie Caster 2.0</i> oder <i>CookieCAD</i> , besteht die Möglichkeit ab der 4. Klasse mit dem Online-Tool <i>Tinkercad</i> eigene einfache 3D-Modelle zu konstruieren. In der ersten Ideenphase können Modelle mit Knete oder Bausteinen konstruiert werden. Die Kinder überlegen sich, was sie gerne modellieren möchten. Es besteht auch die Möglichkeit ein Thema vorzugeben, z.B. „Mein Traumhaus“, „Ein moderner Stuhl“, o.ä.	Knete oder Bausteine, Lego
30 min	Einführung	Plenum	Einführung in die Software <i>Tinkercad</i> durch die Lehrkraft oder den* die Unternehmensvertreter*in	B4.3, PC, Internet, Tinkercad
80 min	Vertiefung	Tandemarbeit	Umsetzung des eigenen Projekts	PC, Internet, Tinkercad
-	Druckphase		Sollte die Schule keinen eigenen Drucker zu Verfügung haben können die fertigen Druckvorlagen ggf. im Partnerunternehmen gedruckt werden. Des Weiteren besteht die Möglichkeit die Konstruktionen bei einem 3D-Druck-Service drucken zu lassen. Es gibt viele Online-Anbieter in einigen Städten aber auch schon Copy-Shops mit 3D-Druckern. Darüber hinaus könnte man auch beim nächsten Makerspace oder FabLab nachfragen, ob die Möglichkeit besteht, vor Ort zu drucken.	3D-Drucker, Filament
15 min	Präsentation	Plenum	Präsentation der Ergebnisse	

## Virtual Reality

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
10 min	Einstieg	Plenum	Klärung der Frage: Was ist Virtual Reality und Augmented Reality?	
50 min	Einstieg	Tandemarbeit	3D-Datenbrille für das Smartphone nach Anleitung basteln und ggf. anmalen oder verzieren	AB 4.6
15 min	Vertiefung	Tandemarbeit	Modelle in Tinkercad in das Format obj (Wavefront Object Format) exportieren und speichern und im Browser in Holobuilder.com öffnen	PC, Internet AB 4.6
15 min	Abschluss	Tandemarbeit	Die Modelle können mit Hilfe der 3D-Brille angesehen werden, ggf. können im Anschluss daran auch noch andere Videos mit der Brille angesehen werden (z.B. Google Expedition oder ZDF 360°-App) oder eigene 360°-Filme produziert werden.	AB 4.6

### 6.2.2 Verlauf für die Sek I und Sek II

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
30 min	Einstieg	Tandemarbeit	Mit dem CookieCaster 2.0 oder CookieCAD können direkt kleine Keksausstecher designt und produziert werden. Die Keksausstecher können am PC mit Hilfe des CookieCasters gezeichnet und ausgedruckt werden <a href="https://cookie-cad.com/">https://cookie-cad.com/</a> <a href="https://www.cs.technik.fhnw.ch/cookiecaster/">https://www.cs.technik.fhnw.ch/cookiecaster/</a>	3D-Drucker
30 min	Einstieg	Plenum	Einführung in SketchUp durch die Lehrkraft (ggf. auch eine andere Modellierungssoftware), erste Übungen mit der Software	AB 4.4, Beamer, Laptop, Internet, Video B4.5
15 min	Ideenphase	Tandemarbeit	Ideen für die eigene 3D-Modellierung entwickeln	

				Ggf. kann auch ein Thema vorgegeben werden (z.B. Modell der Schule, des Klassenraums, etc.	
15 min	Präsentation	Plenum		Vorstellung der Ideen, ggf. Feedbackrunde	
120-240 min	Vertiefung	Tandemarbeit		Modellierung eines eigenen Objekts mit der ausgewählten Software	
-	Vorbereitung und Druckphase	-		Falls ein 3D-Drucker in der Schule vorhanden ist, kann zuvor die Funktionsweise in einer Doppelstunde mit dem Arbeitsmaterial B4.9 erschlossen werden. Im Anschluss können die Modelle ausgedruckt werden. Falls kein 3D-Drucker zur Verfügung steht, besteht die Möglichkeit im Partnerunternehmen die Objekte drucken zu lassen. In einigen Städten gibt es auch schon CopyShops zum Ausdrucken. Im Internet gibt es mittlerweile zahlreiche Online-Shops, die sich auf das 3D-Drucken spezialisiert haben.	ggf. Material B4.9
20 min	Präsentation	Plenum		Präsentation der Ergebnisse, Reflexion und Auswertung	

### Virtual Reality

Zeit	Phase	Sozialform/Impuls	Inhalt/Unterrichtsgeschehen	Material
10 min	Einstieg	Plenum	Klärung der Frage: Was ist Virtual Reality und Augmented Reality	
50 min	Einstieg	Tandemarbeit	3D-Datenbrille für das Smartphone nach Anleitung basteln und ggf. anmalen oder verzieren	AB 4.6
15 min	Vertiefung	Tandemarbeit	Modelle mit Hilfe der App ViewER/VR auf dem Smartphone öffnen	AB 4.6, PC, Internet, Smartphone
15 min	Abschluss	Tandemarbeit	Die Modelle können mit Hilfe der 3D-Brille angesehen werden, ggf. können im Anschluss daran auch noch andere Videos mit der Brille angesehen werden (z.B. Google Expedition oder ZDF 360°-App)	

## Parametrisches Design mit BlocksCAD

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
15 min	Einstieg	Plenum	<p>Klärung der Frage, was Parametrisches Design ist und wo dort Parameter auftreten. Dazu eignet sich bspw. ein abschnittsweises Durchgehen der ersten Seite des AB 4.7</p> <p>Optionale Ergänzung: Die SuS recherchieren online weitere Parametrische Designs</p>	S.1 von AB 4.7
15 min	Einstieg	Lehrkraftpräsentation	<p>Die Lehrkraft präsentiert den SuS die BlocksCAD (<a href="http://www.blocks cad3d.com">www.blocks cad3d.com</a>) Oberfläche.</p> <p>Dabei sollte auf folgende Punkte eingegangen werden:</p> <ul style="list-style-type: none"> <li>• Optional: Über das Anlegen eines Nutzerkontos lassen sich die Zwischenstände von den SuS online speichern. Dazu müssen die SuS einen freien Benutzernamen wählen und Geburtsmonat und -jahr, Geschlecht sowie Land angeben.</li> <li>• Über einen Klick auf „Create Now“ gelangt man zur eigentlichen Programmierumgebung.</li> <li>• Über einen Klick auf den Globus (oben links neben dem BlocksCAD-Logo) lässt sich die Sprache anpassen</li> <li>• Links sind verschiedene Kategorien gelistet, auf die im Arbeitsblatt bzw. auf den Hilfefkarten verwiesen wird. Zum Einstieg eignet es sich vielleicht, aus den 3D-Formen einen Würfel in die Arbeitsfläche (weitere Bereich in der Mitte des Fensters) zu ziehen. Die Parameter lassen sich nun auf gewünschte Werte einstellen.</li> </ul>	Lehrer*innen-Computer, Beamer und Projektionsfläche

			<ul style="list-style-type: none"> <li>• Aus dem Programmcode lässt sich durch einen Klick auf „Rendern“ ein dreidimensionales Modell erstellen, dass sich dann anschließend aus verschiedenen Perspektiven betrachten lässt.</li> <li>• Später werden die SuS deutlich aufwändigere Modelle gestalten, bei denen der Renderprozess länger dauern kann. Hierzu empfiehlt es sich, zum schnellen Testen die Glätte des Modells auf „niedrige“ zu stellen. Hinweis: Bevor Modelle heruntergeladen werden, um sie bspw. zu drucken, sollte jedoch eine „hohe“ Qualität gewählt werden</li> <li>• Um den BlocksCAD Code lokal zu speichern (wenn also keine BlocksCAD-Accounts angelegt wurden), muss „Projekte – Lade Blöcke auf deinen PC herunter“ ausgewählt werden. Um diese Projekte später wieder zu importieren, wird analog mit „Projekte – Lade Blöcke von deinem PC hoch“ verfahren.</li> </ul>	
60-90 min	Vertiefung	Tandemarbeit	<p>Die SuS erarbeiten das AB 4.7 in Tandemarbeit. Je nach Lernstand der Gruppe bietet es sich an, dies abschnittsweise (bspw. nach Aufgabe 1, 2 etc.) im Plenum zu sichern.</p> <p>An den entsprechenden Stellen sollten die Begriffe Parameter, Schleife, Variable und Verzweigung (letztere optional) erklärt werden.</p>	AB 4.7 für jede/n S, Hilfefkarten 4.8 einzeln ausdrucken und vorne (optimalerweise) laminiert auslegen
20 min	Abschluss	Plenum	Die SuS präsentieren sich gegenseitig ihre Modelle.	
–	Projektarbeit (optionale Ergänzung)		Optional können die SuS nun eigene dreidimensionale, parametrisch gestaltete Formen (die nicht zwingend Blumen-förmig sein müssen) entwerfen, die im Anschluss daran bei Verfügbarkeit eines 3D-Druckers auch gedruckt werden können.	



## 7 Einbettung in verschiedene Fächer und Themen

Die folgenden Kompetenzen finden sich entweder in den Bildungsstandards der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

### Informatik

Die Schülerinnen und Schüler ...

- bewerten die Bedeutung eines Informatiksystems für das Individuum und die Gesellschaft
- beschreiben Modelle als vereinfachtes Abbild der realen Welt
- reflektieren und beurteilen die eigene Modellierung
- wählen zur Lösung eines Problems geeignete Standardsoftware (Textverarbeitung, Tabellenkalkulation, Erfassen und Verwaltung von Daten, Bildbearbeitung) aus,
- dokumentieren und präsentieren ihre Arbeitsergebnisse

Bei Behandlung des Parametrischen Designs außerdem:

- strukturieren Handlungsabläufe in Teileinheiten
- benennen typische Bestandteile von algorithmischen Abläufen wie Schleifen, Verzweigungen, Variablen
- entwickeln einen Algorithmus auf experimentelle Weise

### Mathematik

Die Schülerinnen und Schüler ...

- können zwei- und dreidimensionale Darstellungen von Bauwerken zueinander in Beziehung setzen und nach Vorlage bauen, zu Bauten Baupläne erstellen, Kantenmodelle und Netze untersuchen. (GS)
- erkennen und beschreiben geometrische Strukturen in der Umwelt
- stellen Körper (z.B. Modell) dar und erkennen Körper aus ihren entsprechenden Darstellungen
- zeichnen und konstruieren geometrische Figuren unter Verwendung angemessener Hilfsmittel wie Lineal, Zirkel oder Geometriesoftware.

### Kunst

Die Schülerinnen und Schüler ...

- können Architekturmodelle mit adäquaten Werkzeugen und Materialien erstellen
- verfügen über Erfahrungen im Umgang mit technischen Medien und Verfahren der Bildbearbeitung und können sie zur Lösung von gestalterischen Aufgaben einsetzen
- können raumhafte Konstruktionen erfinden und bauen
- können proportional, stofflich und plastisch Dinge darstellen und produzieren
- entwickeln und skizzieren zielgruppenbezogen und auf Basis von Designkriterien Ideen zu einem Produkt

## Sozialkunde

Die Schülerinnen und Schüler...

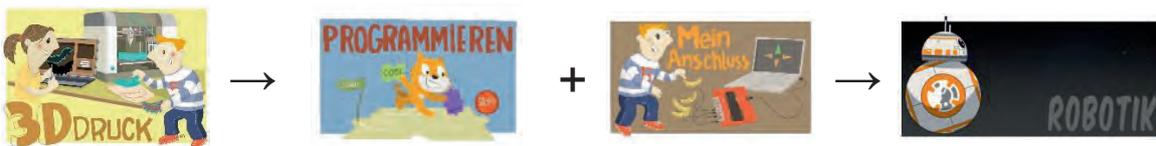
- erkennen, wie technisch-industrieller Fortschritt die Berufs- und Lebenswelt des Einzelnen und die Gesellschaft verändert
- Dimensionen und Ausmaß der derzeitigen Veränderungen erfassen und individuelle und politische Bewältigungsmöglichkeiten erörtern
- den Betrieb als Stätte der Produktion und Kooperation kennen lernen und Problemfelder, insbesondere Umweltfragen, erörtern

## 8 Anschlusssthemen

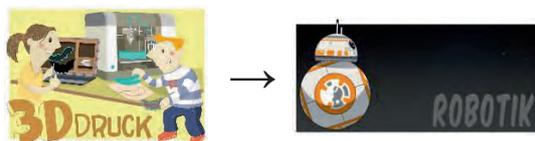
Als Anschlusssthemen im Zusammenhang mit IT2School bieten sich folgende Module an:

### IT spielend entdecken

Wenn in ihrem Unterricht weiterhin das spielerische Entdecken und das kreative Gestalten im Mittelpunkt stehen soll, dann können Sie folgende Module auswählen:



oder



## 9 Literatur und Links

- **3D-Drucken – Unsere Erlebnisse mit dem 3D-Drucker:** <https://3drucken.ch/> - Ideen für die Schule, Tipps für Hard- und Software
- Horsch, Florian (2013): **3D-Druck für alle – Der Do-It-Yourself-Guide.** Carl Hanser Verlag München
- **Thingiverse.com:** Eine Plattform, um 3D-Modelle, Designs und Ideen zu teilen.
- **Software Sketchup Make:** <https://www.sketchup.com/de/download/all>
- **Minecraft Print:** <http://www.printcraft.org/> Mit Hilfe von printcraft können Objekte in dem Spiel Minecraft erstellt und anschließen als Druckdatei heruntergeladen werden.
- **BeetleBlocks** ist eine grafische Entwicklungsumgebung (ähnlich zu Scratch) um 3D-Modelle zu erstellen. <http://beetleblocks.com/>

- **BlocksCAD** ist auch eine grafische Entwicklungsumgebung <https://www.blockscad3d.com/>
- **Makerbot Education (2015):** Makerbot in the Classroom. Eine Einführung in 3D-Druck und Design. Online: <https://www.makerbot.com/stories/education/makerbot-in-the-classroom-a-resource-for-educators/>
- **3Doodler Start:** 3D-Stift zum Malen für Kinder: <http://3doodlerstart.com/>
- **Google Expeditions:** App für virtuelle Schulausflüge. Ziel von *Expeditions* ist es, Orte oder historische Ereignisse mit Hilfe einer VR-Brille und dem Smartphone erlebbar zu machen. [https://edu.google.com/products/vr-ar/expeditions/?modal\\_active=none#about](https://edu.google.com/products/vr-ar/expeditions/?modal_active=none#about) (im Google Play Store)
- **ZDF 360°-App:** ZDF-Produktionen in 360° für das Smartphone und VR-Brille <http://vr.zdf.de/> (im Google Play Store)
- **Arte 360 App:** Arte-Produktionen in 360° für das Smartphone und VR-Brille <http://sites.arte.tv/360/de> (im Google Play Store)
- **Holobuilder:** Online-Tool um Virtual Reality-Rundgänge o.ä. zu erstellen [www.holobuilder.com](http://www.holobuilder.com)

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B4.1	Keksausstecher	Einführung in die Modellierung eigener Keksausstecher (für die Lehrkraft)
 B4.2	Keksausstecher mit CookieCAD	Arbeitsauftrag für Schülerinnen und Schüler
 B4.3	Einführung in Tinkercad	Einführung für die Lehrkraft
 B4.4	Einführung in Sketch Up	Tutorial für Schülerinnen und Schüler
 B4.5	Video-Tutorial	Tutorial für Lehrkraft, Schülerinnen und Schüler
 B4.6	Virtual Reality	Anregung zur Einbettung von Inhalten zu Augmented und Virtual Reality im Unterricht
 B4.7	Parametrisches Design mit BlocksCAD	Einführung in Parametrisches Design und Arbeitsaufträge zum Kennenlernen von BlocksCAD
 B4.8	Hilfekarten	Hilfekarten zu Material B4.7
 B4.9	Aufbau und Funktionsweise von 3D-Druckern	Arbeitsmaterial zum Aufbau und zur Funktionsweise von 3D-Druckern
 B4 Muster	Musterlösung	Musterlösung für das Arbeitsmaterial B4.9

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
FDM – Fused Deposition Modeling (ehemals patentierter Begriff)	Fertigungsverfahren im Bereich des 3D-Druckens, eine alternative Bezeichnung dieses Verfahrens lautet Fused Filament Fabrication (FFF), das gedruckte Objekt wird Schicht für Schicht ausgedruckt
STL - Standard Transformation Language	Datei-Format für 3D-Modelle, wie sie in CAD-Programmen erstellt werden
Upcycling	Abfallprodukte oder nicht mehr verwendete Produkte werden in „neue“ nützliche Produkte umgewandelt, 3D-Druck kann hier weiterhelfen. Ersatzteile oder zusätzliche Bauteile können passgenau modelliert und ausgedruckt werden.

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Eigene Keksausstecher gestalten

## Cookie Caster

Mit Hilfe des Cookie Casters lassen sich auf einfachstem Weg individuelle Keksausstecher selbst produzieren und drucken. Daher ist der Cookie Caster als Einstieg für Kinder in den 3D-Druck bestens geeignet.

Um einen eigenen Keksausstecher zu erstellen, benötigt man nur einen Computer mit Internetzugang.

Aufgerufen werden kann der CookieCaster der Fachhochschule Nordwestschweiz FHNW unter <https://www.cs.technik.fhnw.ch/cookiecaster/> und nach Klicken auf den *Los geht's!* Button kann das Gestalten des Ausstechers starten.

Willkommen beim CookieCaster 2.0

Möchtest Du Deine eigene persönliche Keksförmung gestalten?

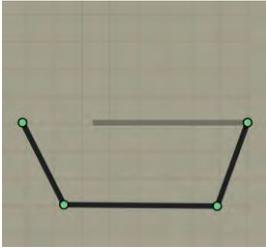
Los geht's!

Bist Du neu hier? Klicke auf das Fragezeichensymbol oben rechts, um die wichtigsten Schritte zu lernen.

## Erste Formen zeichnen

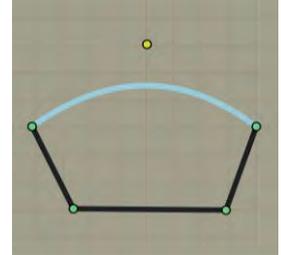
Die Benutzeroberfläche ist schlicht aufgebaut und dadurch intuitiv zu bedienen:





Auf der Arbeitsfläche lassen sich einfach Formen zeichnen. Dazu klicken Sie mit der Maus in das Feld, um den Startpunkt festzulegen. Mit jedem weiteren Mausklick werden Knotenpunkte gesetzt, die sich automatisch miteinander verbinden. Die zweidimensionalen Modelle von Keksausstechern bestehen aus geschlossenen Pfaden.

Nachdem die Knotenpunkte gesetzt wurden und der Pfad geschlossen wurde, erscheint nach der Auswahl einer Kante zwischen zwei Knotenpunkten ein weiterer Punkt. Dieser Punkt kann mit gedrückter Maustaste in jede gewünschte Richtung gezogen werden. Dadurch können Rundungen oder individuelle Formen entstehen.



Um die Figur mit einem 3D-Drucker auszudrucken, muss das Modell über einen Klick auf „Export 3D“ als STL-Datei heruntergeladen und mit der Slicing-Software des verfügbaren Druckers weiterverarbeitet werden.

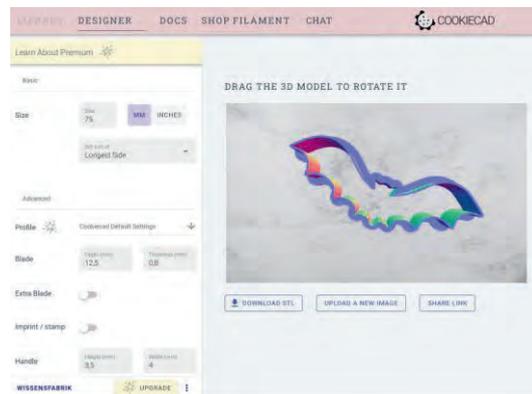
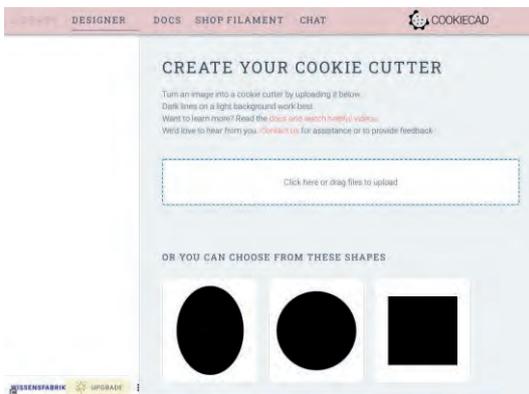
## CookieCAD

Der im Vorherigen vorgestellte Cookie Caster ermöglicht das eigenständige Zeichnen von Formen, die zu einem dreidimensionalen Modell umgewandelt werden und sich dann über einen 3D-Drucker zu einem Keksausstecher verwandeln lassen.



Das Tool CookieCAD (<https://app.cookiecad.com>) ermöglicht es, alternativ (möglichst kontraststarke) Bilder hochzuladen, deren Konturen dann automatisiert zu einem 3D-Modell eines Keksausstechers verarbeitet werden. Dazu wird ein kostenloses Benutzerkonto benötigt<sup>1</sup>.

Klicken Sie auf *Click here or drag files to upload*, um ein Bild hochzuladen. Danach können Sie u. a. Einstellungen zur Größe des Modells vornehmen. Die Umrisse werden dann automatisch nachgezeichnet. In der Vorschau ist das 3D-Modell zu sehen. Abschließend kann das Modell kostenfrei über einen Klick auf *Download STL* heruntergeladen werden.

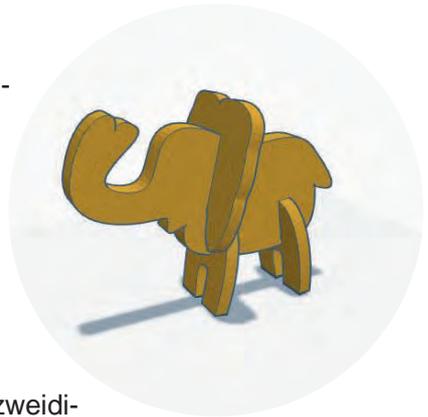


<sup>1</sup> Es empfiehlt sich, zum Zweck der Anlegung eines Profils auf eine sogenannte Wegwerf-E-Mail zurückzugreifen. Der Service des Vereins Internet Ulm hat sich in den letzten Jahren als zuverlässiger Dienst herausgestellt: <https://ulm-dsl.de/> Es bietet sich insbesondere für den Einsatz in unteren Klassenstufen an, Kontozugänge für die Schülerinnen und Schüler vorzubereiten. Auch hierfür kann der genannte Wegwerf-E-Mail-Dienst genutzt werden.

## Keksausstecher mit Inkscape und Tinkercad

Tools wie der Cookie Caster oder CookieCAD sind bestens geeignet, um schnell eigene Keksausstecher-Modelle zu gestalten.

Für detailliertere Modellierungen – wie sie beispielsweise für Keksausstecher, die sich später für 3D-Kekse (siehe rechts abgebildetes Modell) nutzen lassen, benötigt werden – sind diese Tools jedoch nur eingeschränkt nutzbar. Zur Gestaltung von entsprechend komplexeren Keksausstechern stehen aber andere Tools zur Verfügung:



**Inkscape** ist eine freie Software zur Bearbeitung und Erstellung zweidimensionaler Vektorgrafiken. In Kombination mit **Tinkercad** lässt sich Inkscape nutzen, um Modelle von Keksausstechern zu gestalten – auch, um damit später bspw. 3D-Kekse auszustechen.



**INKSCAPE**  
Draw Freely.

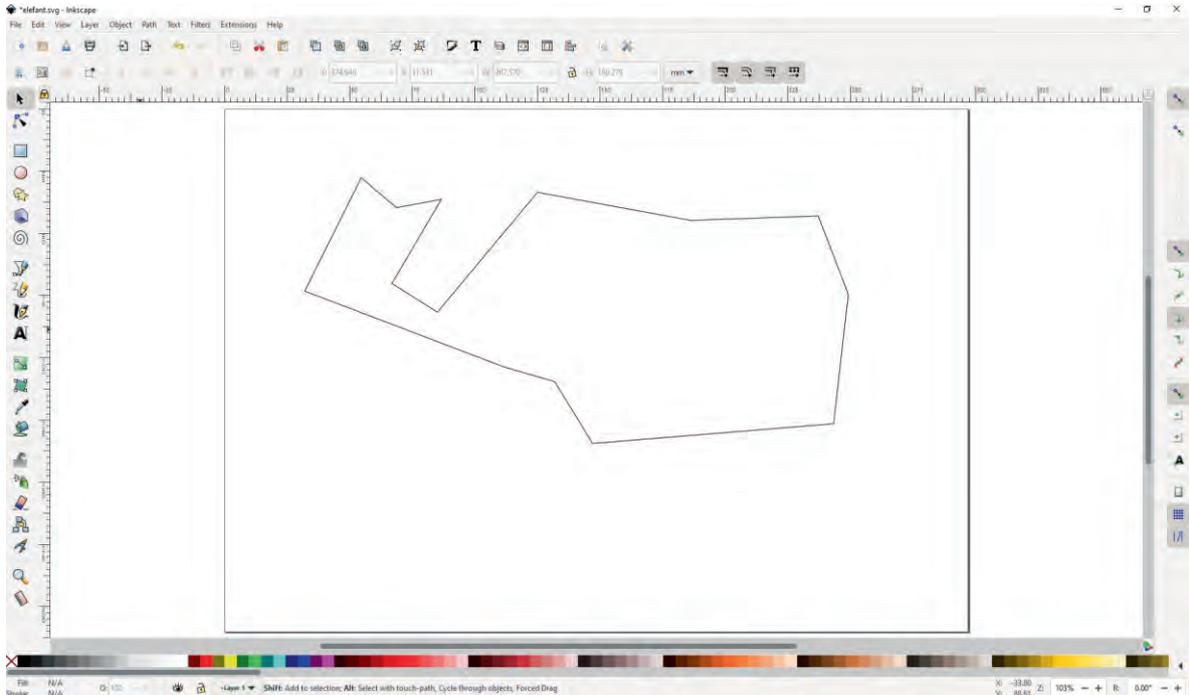
Inkscape ist unter <https://inkscape.org/de/> **kostenlos** herunterladbar. Es gibt Versionen für Windows, macOS und Linux. Das Programm muss zunächst auf dem Computer installiert werden.

Häufig gewählte Motive für 3D-Kekse sind Tiere (siehe Abb. oben). Im (projektartigen) Unterricht bietet es sich an, die Schülerinnen und Schüler zunächst umsetzbare Formen diskutieren zu lassen und anschließend Prototypen (bspw. aus Pappe/Karton) zu entwickeln. Im Anschluss daran kann die Umsetzung der Formen in Inkscape und Tinkercad passieren.

Zunächst wird das Programm geöffnet und ein neues Dokument angelegt. Es bietet sich an, analog zum Vorgehen bei CookieCAD auch hier eine Bilddatei von Roh-Entwürfen zu importieren (*File* → *Import...*).

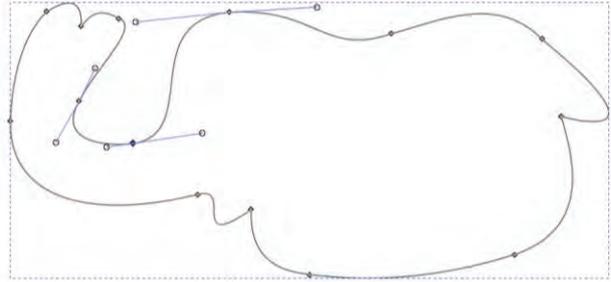


Danach wird über das Zeichnen-Tool (Icon siehe linke Abb.) ein **geschlossener Pfad** in einer groben Form der späteren Keksausstecher erstellt.

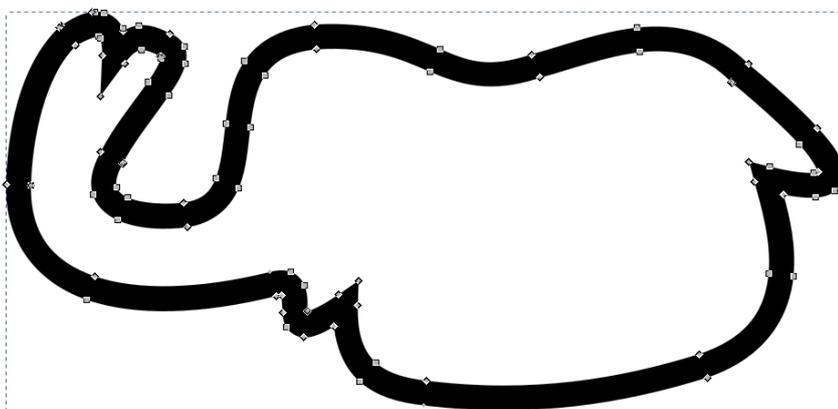
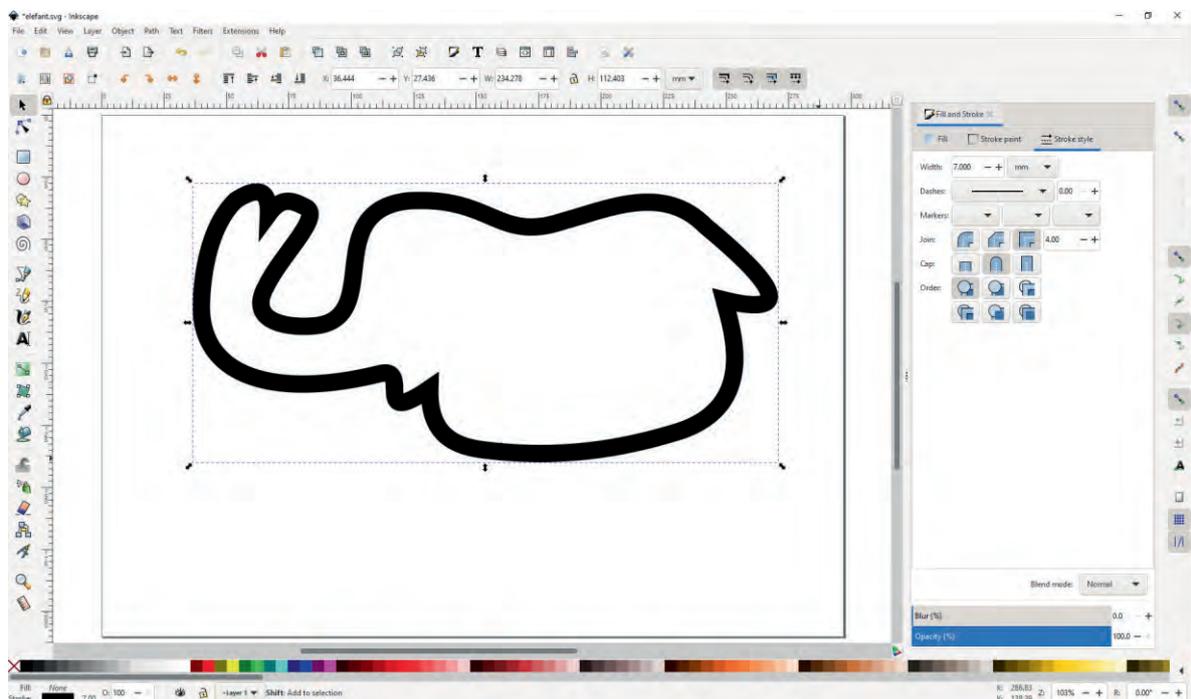




Krümmungen zwischen zwei Pfadpunkten lassen sich nun über sogenannte **Bézierkurven** umsetzen. Dazu muss zunächst das Pfad-Editieren-Werkzeug ausgewählt werden (Icon siehe linke Abb.). Über eine Anpassung der dann angezeigten Kontrollpunkte lassen sich Krümmungen erzeugen.



Ist ein zufriedenstellendes Ergebnis erreicht, so muss im nächsten Schritt die Konturdicke angepasst werden. Sie stellt die Stärke des späteren Ausstechers dar. Zur Anpassung der Konturlinie muss der Pfad zunächst ausgewählt werden. Das Menü zur Einstellung öffnet sich nach einem Aufruf von *Object* → *Fill and Stroke...*). Im Anschluss können die für die jeweilige Keks-ausstecher-Form benötigten Anpassungen vorgenommen werden. Wichtig ist, dass das Objekt unter *Fill* keine Füllfarbe zugewiesen bekommen hat (siehe rechts abgebildetes Icon). **X**



Anschließend muss der **Pfad in ein Objekt umgewandelt** werden, damit Tinkercad ihn später beim Import erkennt und das Innere frei lässt.

Hierzu muss das Objekt ausgewählt und dann über *Object* → *Stroke to path* umgewandelt werden. Änderungen der Konturstärke sind nun nicht mehr ohne Weiteres möglich.

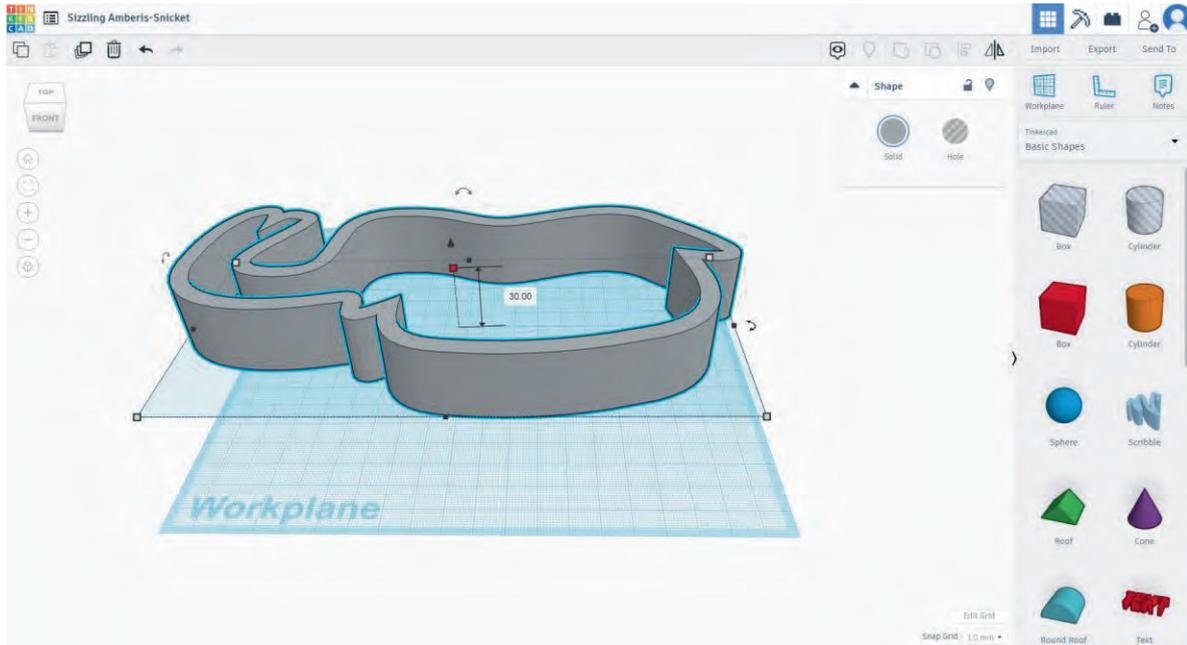
Das Keksausstecher-Modell kann nun als **SVG-Datei** gespeichert werden (*File* → *Save as...*).

Im Anschluss lässt es sich in **Tinkercad**<sup>2</sup> importieren. Dazu muss nach der Anmeldung in Tinkercad (<https://www.tinkercad.com>) auf *Import* → *Chose a file* geklickt und im Anschluss die zuvor gespeicherte SVG-Datei ausgewählt werden.

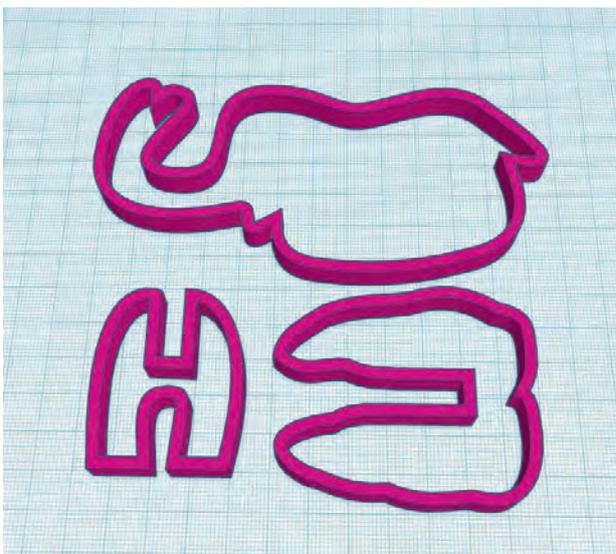


Im sich daraufhin öffnenden Fenster sollten „Center on: Artboard“ und „Scale (%): 100“ ausgewählt sein.

Die **Höhe des Keksausstechers** lässt sich nun in Tinkercad anpassen, indem der entsprechende Kontrollpunkt ausgewählt (siehe folgende Abbildung) und verschoben wird. Dabei ist auch die Angabe der Höhe (in mm) möglich, indem der gewünschte Wert in das dann angezeigte Feld eingegeben wird.



Tinkercad erweitert also die zweidimensionale Datei (SVG) um die dritte Dimension. Das so erzeugte Modell kann aus Tinkercad (bspw. als STL-Datei) exportiert und dann für den 3D-Druck weiterverwendet werden.



Mit etwas Übung können so Keksausstecher produziert werden, die beispielsweise die zu Beginn motivierten dreidimensionalen Kekse ermöglichen.

Bei der Zubereitung des Keksteigs sollte besonders darauf geachtet werden, dem Teig genügend Zeit zum Auskühlen zu lassen, bevor die Formen ausgestochen werden. Auch nach dem Backen und vor dem Zusammenstecken einzelner Keks-Teile sollten die Backprodukte zunächst vollständig ausgekühlt sein.

<sup>2</sup> Weitere Informationen zu Tinkercad finden sich in der Modulbeschreibung und dem entsprechenden Arbeitsmaterial.

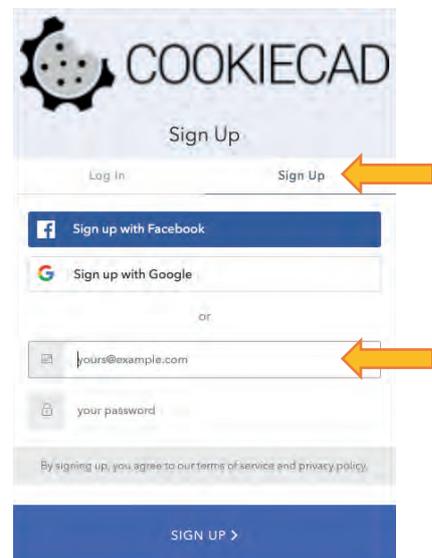
# Keksausstecher mit CookieCAD

Mit Hilfe von CookieCAD können ganz einfach Keksausstecher selbst erstellt werden.

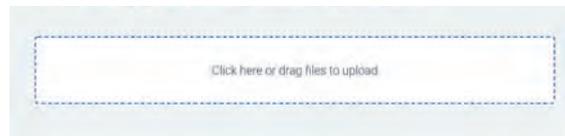
Gehe dabei wie folgt vor:

1. Nimm dir einen Zettel und einen Stift und designe deine eigene Form z. B. ein Tier, ein Haus, ein Wahrzeichen deiner Stadt, ...
2. Wenn du mit deiner Form zufrieden bist, schneide sie aus und übertrage sie auf schwarzen Pappkarton.
3. Schneide den schwarzen Pappkarton aus und fotografiere ihn auf weißem Untergrund ab. Speichere das Foto auf dem Computer.
4. Nun öffnest du den Internetbrowser am Computer und gibst folgende Adresse ein: <https://app.cookiecad.com>

5. Wenn du CookieCAD zum ersten Mal benutzt, musst du ein Benutzerkonto erstellen. Dazu verwendest du am besten zuerst eine Wegwerf-E-Mail an. Das geht beispielsweise über den Dienst des Vereins Internet Ulm: <https://ulm-dsl.de/>. Wenn du schon ein Benutzerkonto hast oder deine Lehrerin bzw. dein Lehrer dir Zugangsdaten gegeben hat, kannst du mit Punkt 6 weitermachen, nachdem du dich mit deinen Daten angemeldet hast.

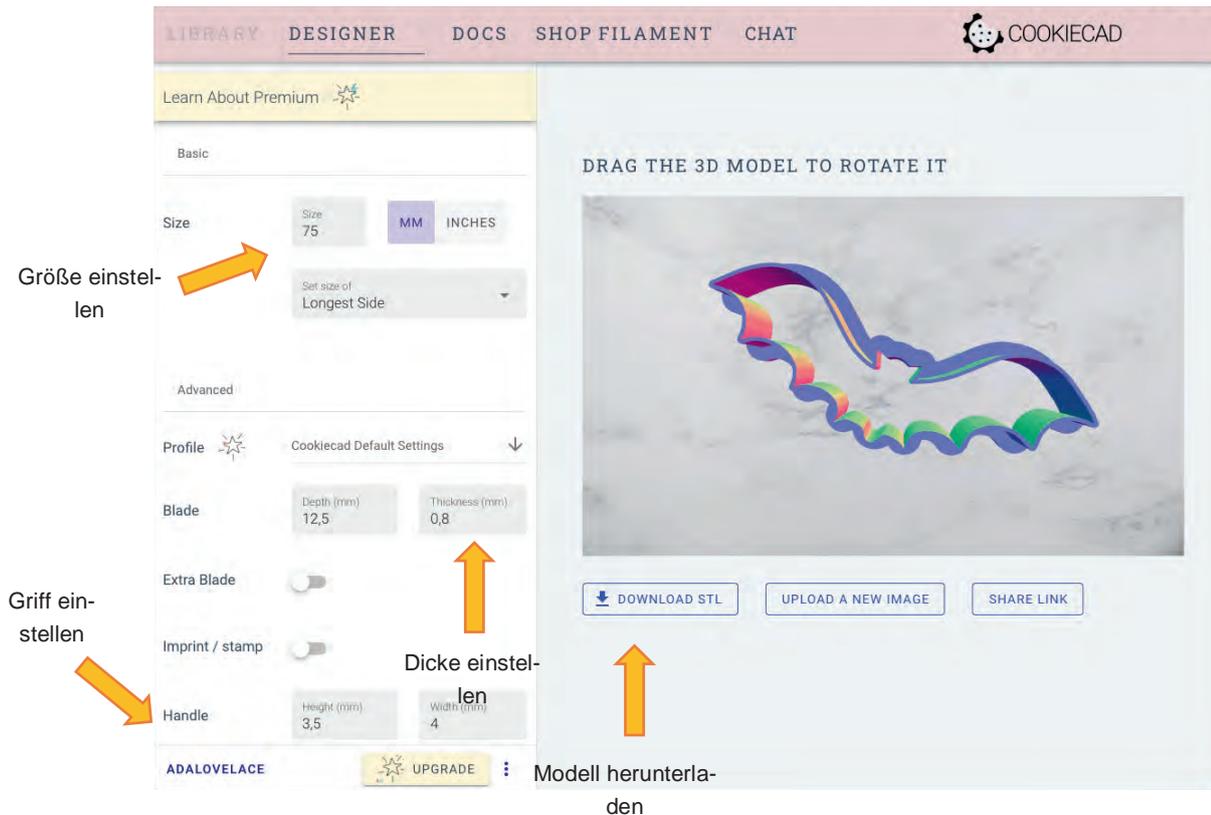


- a. Unten links findest du den Button *Log In* (das heißt „anmelden“). Nachdem du darauf geklickt hast, öffnet sich ein Fenster, das so aussieht wie das rechts abgebildete.
  - b. In dem Fenster klickst du zunächst auf Sign Up (das lässt sich mit „registrieren“ übersetzen). Unten gibst du dann deine Wegwerf-E-Mail-Adresse ein (bspw. `adalovelace@ulm-dsl.de`) und wählst ein Passwort.
6. Klicke auf *Click here or drag files to upload* (das heißt: *Zeichne Deinen eigenen Keksausstecher*). Danach öffnet sich ein Fenster, indem du dein Bild auswählen kannst.



7. Nachdem du dein Bild hochgeladen hast, benötigt das Programm ein paar Sekunden, um ein Keksausstecher-Modell daraus zu entwickeln.

- Anschließend öffnet sich eine Ansicht, in der du das Keksausstecher-Modell betrachten kannst. Ziehe dazu bei gedrückter Maustaste an dem Modell. Mit dem Mausrad kannst du rein- und rauszoomen.



- In der Ansicht kannst du unter anderem die Größe des Keksausstechers einstellen. Hier einigt ihr euch am besten mit eurer Lehrerin bzw. eurem Lehrer auf eine Größe.

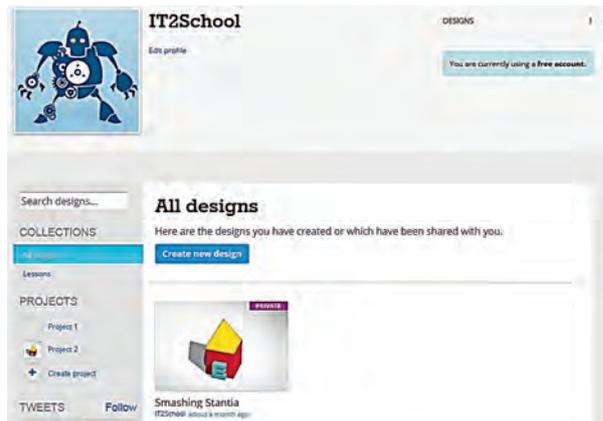
- Abschließend kann das Modell über den Button *Download STL* heruntergeladen werden. Dann kann es beispielsweise weiterverarbeitet werden, um es 3D zu drucken.

# Tutorial: 3D-Modellierung mit Tinkercad

Tinkercad (<https://www.tinkercad.com>) ist ein kostenloses, browserbasiertes CAD-Modellierungsprogramm. Das Programm ist mittlerweile größtenteils in deutscher Sprache nutzbar, sehr leicht und intuitiv zu bedienen und eignet sich daher auch für jüngere Schülerinnen und Schüler. Viele Bausteine (Quader, Kugel, Pyramide, etc.) werden zur Verfügung gestellt, wodurch einfachste Modelle wie im Baukastenprinzip erstellt werden können.

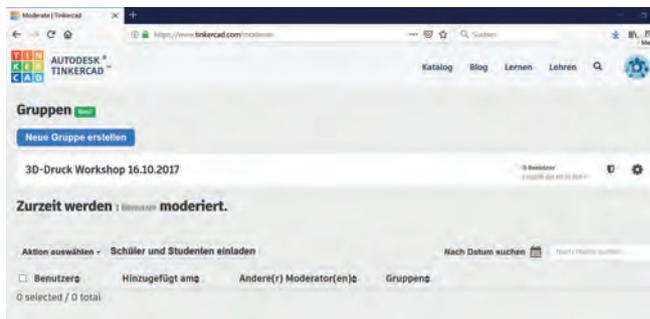
Alle, die Tinkercad nutzen wollen, müssen sich einen eigenen privaten Account einrichten. Lehrkräfte haben außerdem einen „Klassenraum“ zur Verfügung, um Schülergruppen zu moderieren. Dieser kann im eigenen Profil unter „Kinder moderieren“ erstellt werden. Dort können auch Schüler/innen eingeladen und zu einzelnen Gruppen hinzugefügt werden.

**Privataccount:** In der eigenen Übersicht finden sich die bisher erstellten Modelle und ein Button, um ein neues Projekt zu starten.



Dashbord eines privaten Accounts

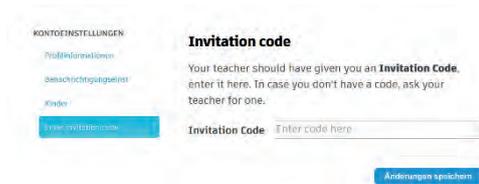
## Gruppe:



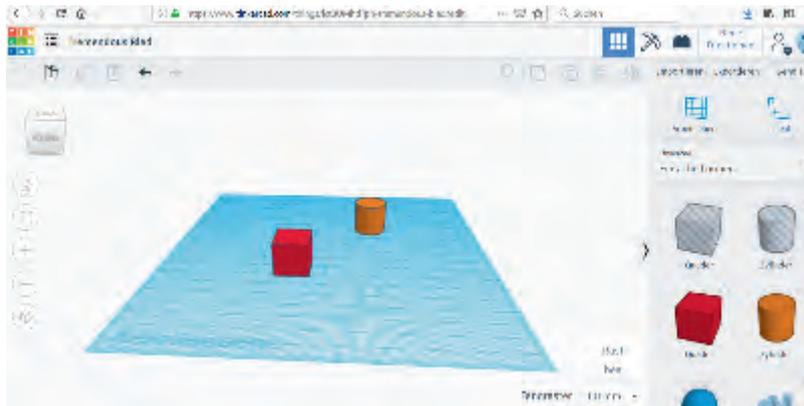
Übersicht für eine Gruppe

Um als Lehrkraft Schülerinnen und Schüler zum „Klassenraum“ hinzuzufügen, benötigen diese einen Code. Dieser Code wird bei Klick auf den Link „Lehren“ oben rechts in der Menüleiste erzeugt. Dieser Code kann entweder per Mail oder auch analog an die Schülerinnen und Schüler gegeben werden. Diese können in ihrem Account unter Profil auf den Link „Enter invitation Code“ den Code eingeben und sind somit mit dem Klassenraum der Lehrkraft verbunden.

Die Lehrkraft kann dann alle Designs und Projekte der/s Schülers/in sehen.

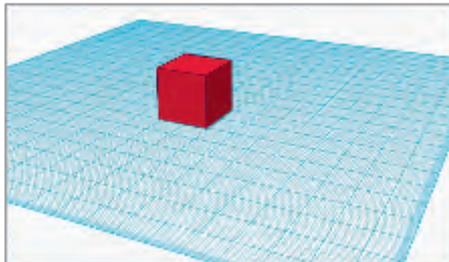


## Die Modellierungs-Oberfläche



Auf der blauen Ebene (Arbeitsebene) werden die gewünschten Objekte platziert. In der rechten Leiste finden sich zahlreiche Grundformen sowie Buchstaben, Zahlen und Symbole, die mit gedrückter Maustaste auf das blaue Feld gezogen werden.

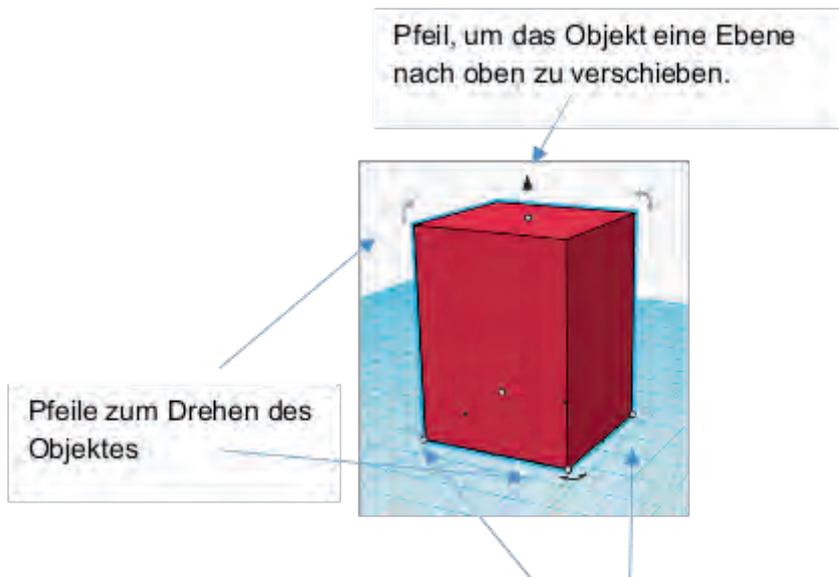
Links oben befindet sich ein Würfel mit Perspektivangaben, um das Objekt aus allen Blickwinkeln zu betrachten. Neben den Pfeilen kann die Oberfläche auch mit der rechten Maustaste gedreht werden. Bei Klick auf das Haus wird die Arbeitsebene mit dem Objekt aus der Standardsicht gezeigt (von vorne und schräg oben). Mit den Plus- und Minustasten kann die Ansicht vergrößert oder verkleinert werden.



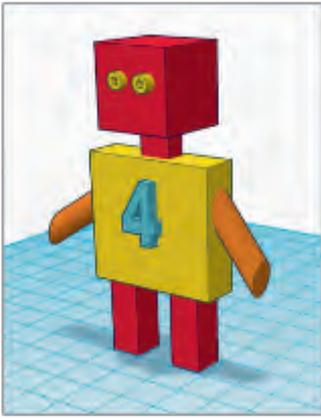
### Objekte platzieren

Mit gedrückter Maustaste können einzelne Objekte auf die Oberfläche gezogen werden.

### Objekte vergrößern und drehen



An den weißen Eckpunkten können die Objekte größer oder kleiner skaliert werden. Hierfür einen weißen Punkt anklicken und auf die gewünschte Größe ziehen.

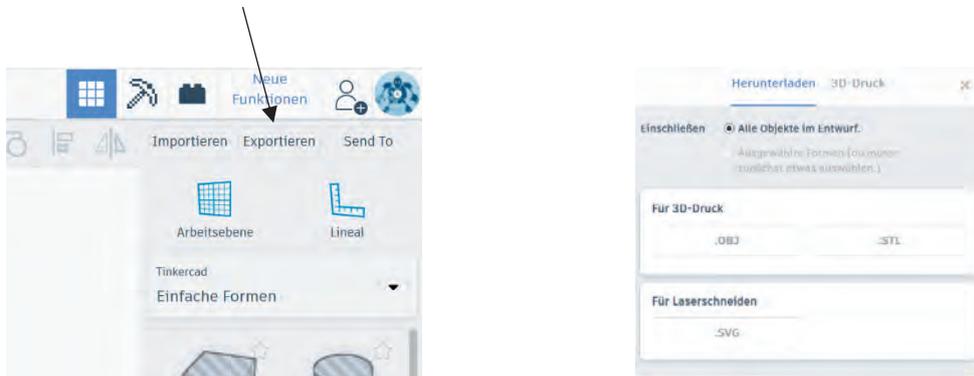


Die einzelnen vorgefertigten Bausteine ermöglichen es, einfache Strukturen zu modellieren.

Lassen Sie Ihre Schülerinnen und Schüler zu Beginn einfache Figuren, wie z.B. ein Haus modellieren. Sie können daran üben, wie beispielsweise eine Pyramide auf einen Quader gesetzt werden kann.

## Ausgabe

Rechts oben, unter dem Button „Exportieren“ kann das fertige Modell gespeichert und als Druckdatei heruntergeladen werden.



**.stl-Format:** Standard Transformation Language – 3D-Druckformat, am verbreitetsten

**.obj-Format:** Wavefront Objekt Format – Alternative zu STL-Dateien, wenn Informationen über Farben oder Materialien angegeben werden sollen

**.svg-Format:** Scalable Vector Graphics (SVG, englisch für skalierbare Vektorgrafik), Format für die Verarbeitung am Lasercutter.

## Link-Tipps

Einführung in TinkerCAD: <https://www.tinkercad.com/learn>

Erste Schritte mit Tinkercad. <https://thredom.de/3d-drucker-software/tinkercad-tutorial-deutsch>

Tinkercad mit Schülerinnen und Schülern unter 13 Jahren: <https://learningwithlucie.blogspot.de/2015/05/using-tinkercad-with-students-under-13.html>

# SketchUp – Tutorial für ein Haus

SketchUp ist eine Software zur Erstellung dreidimensionaler Objekte. Sie kann in reduzierter Form kostenlos genutzt werden. In diesem Tutorial wollen wir ein Haus mit SketchUp modellieren.

Die kostenfreie Version der Software SketchUp Make 2017 kann hier heruntergeladen und installiert werden: <https://www.sketchup.com/de/download/all>

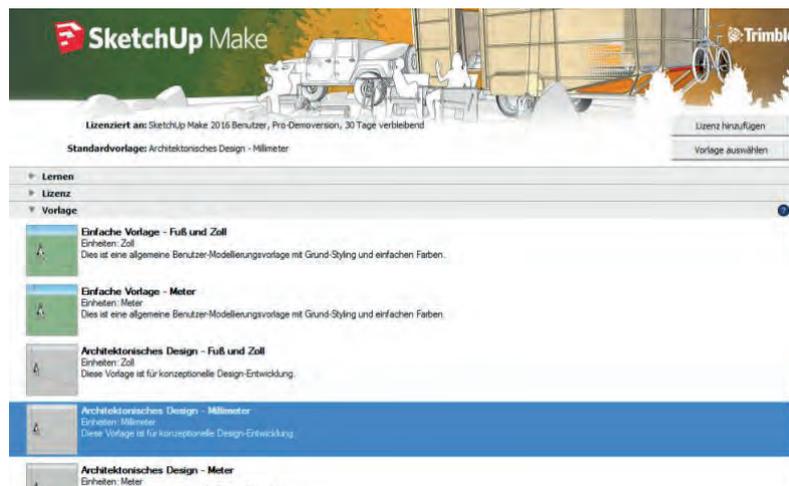
Darüber hinaus gibt es die Möglichkeit, SketchUp Free browserbasiert, das heißt ohne Installation, zu nutzen. Dazu muss man sich lediglich einen Account einrichten (Name, Emailadresse und Passwort): <https://www.sketchup.com/de/products/sketchup-free>

Klick auf „Mit dem Modellieren loslegen“.

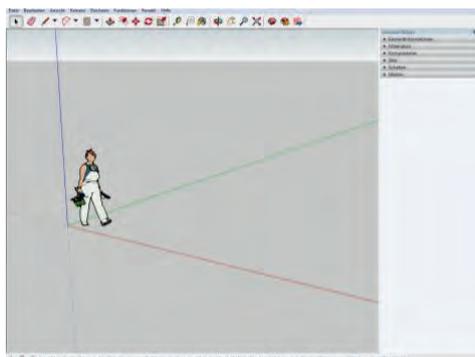
Dieses Tutorial arbeitet mit der Installationsversion SketchUp Make 2017.

## Start

Nach dem Start des Programms wählt man zu Beginn eine Vorlage aus, d.h. man entscheidet sich für eine Maßeinheit: Meter, Millimeter oder Zoll. Tipp: Wählt Millimeter aus.

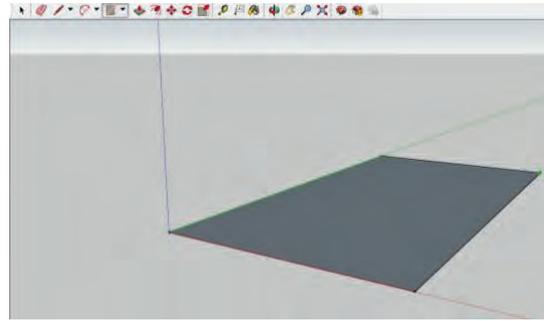


Nun wird euch das Konstruktionsfenster angezeigt. Oben, wie in anderen Programmen auch findet ihr die Symbolleiste und darunter die Zeichenfläche. In die Zeichenfläche sind drei Raumachsen zur Orientierung eingezeichnet. Außerdem seht ihr einen Menschen, der euch weitere Orientierung im Raum geben soll. Für unsere Modellierung benötigen wir den Menschen nicht. Klickt ihn an und löscht ihn mit der Entfernen-Taste.



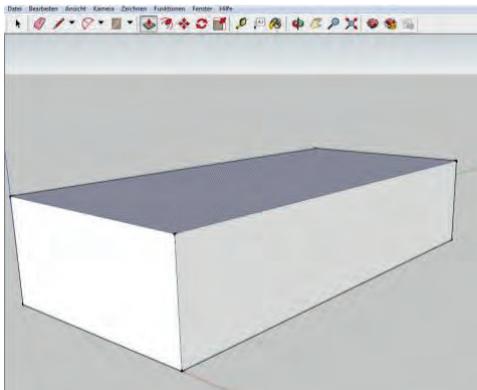
## Schritt 1: Rechteckfunktion

Um ein Haus zu modellieren, benötigen wir zuerst ein Rechteck. Wählt dazu die Rechteckfunktion in der oberen Werkzeugleiste aus. Zieht mit gedrückter Maustaste ein Rechteck in der gewünschten Größe auf. Startet am Besten auf dem Nullpunkt, an dem alle Linien aufeinander treffen.



## Schritt 2: Formfunktion

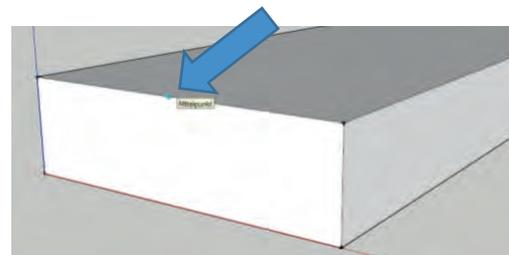
Um nun das Rechteck in eine dreidimensionale Form zu bringen, wählt das Werkzeug *Drücken und Ziehen* aus. Geht mit dem Cursor auf eurer Rechteck und klickt es an.

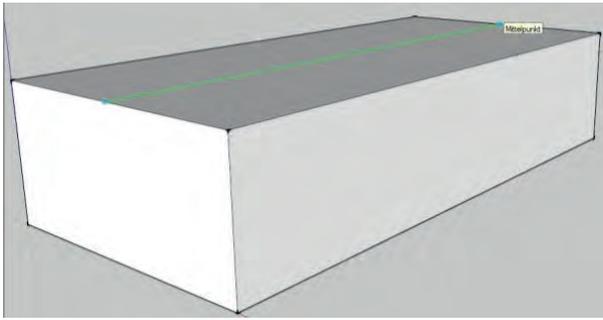


Bewegt den Cursor, um die Fläche in eine 3D-Form zu drücken oder zu ziehen. Wenn die gewünschte Form aufgezo-gen ist, klickt noch einmal, um den Vorgang zu beenden.

## Schritt 3: Linienfunktion

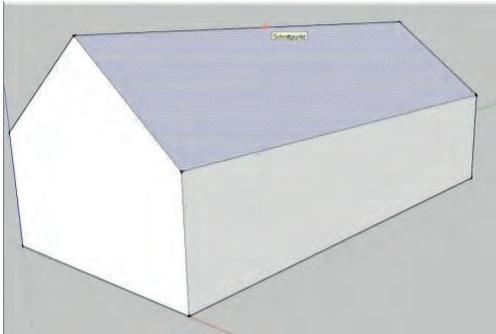
Im nächsten Schritt modellieren wir ein Dach für unser Haus. Dafür wählen wir den *Stift* in der Werkzeugleiste aus. Wenn ihr nun mit dem Stift entlang der Seitenlinie fährt, wird euch angezeigt, wenn ihr den Mittelpunkt erreicht habt – ein blauer Punkt wird angezeigt.





Nun klickt ihr den Punkt an und geht mit eurem Stift auf die gegenüberliegende Seite, an dem ihr auch auf den Mittelpunkt klickt. Es erscheint eine Line.

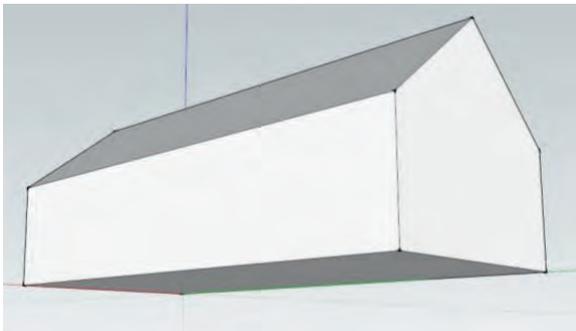
Um das Dach zu modellieren brauchen wir als nächstes das *Verschiebewerkzeug*



Nachdem ihr es ausgewählt habt, klickt auf die Mittellinie, die ihr euch gerade gezogen habt und zieht sie nach oben.

Nun sieht es schon nach einem Haus aus.

## Schritt 4: Rotier- und Verschiebefunktion

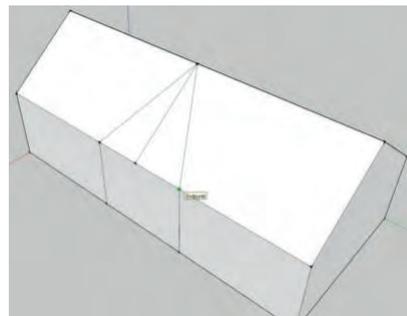


Mit Hilfe der *Rotierfunktion* könnt ihr euch euer Modell von allen Seiten ansehen. Die Kamera dreht sich um das Modell. Die *Handfunktion* bewegt die Kamera horizontal und vertikal. Probiert es einfach mal aus.

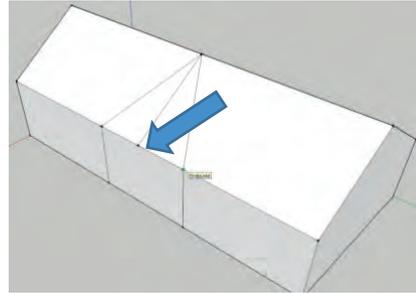


## Schritt 5: Anfügen eines Anbaus

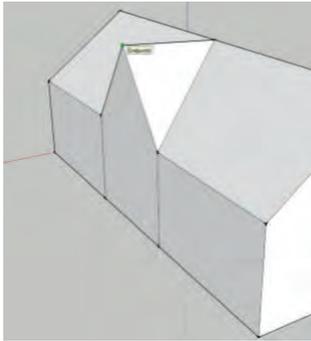
In diesem Schritt wollen wir unserem Haus noch einen Anbau anfügen. Wählt dafür wieder den *Stift* aus und zeichnet folgende Linien ein:



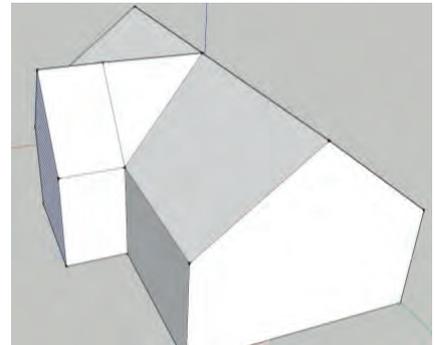
Wählt danach wieder das *Verschiebewerkzeug* aus und setzt an folgendem Punkt an:



Zieht den Punkt mit Hilfe des Werkzeugs nach oben. Dein Haus sollte nun folgendermaßen aussehen.

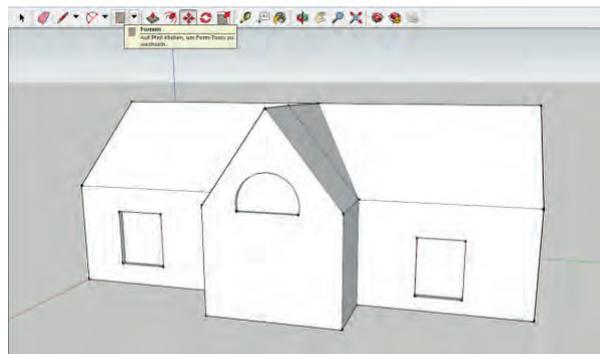


Mit Hilfe des Werkzeug *Drücken und Ziehen* könnt ihr nun den gesamten Teil des Hauses herausziehen. Zieht es in die gewünschte Größe.



## Schritt 6: Einfügen von Fenstern und Türen

Nun könnt ihr euer Haus mit Fenstern und auch einer Tür versehen. Die passenden Werkzeuge kennt ihr schon. Wählt das Werkzeug *Formen* aus und zieht ein Rechteck an der passenden Stelle auf. In dem unteren Beispiel wurden die Fenster mit Hilfe des Werkzeugs *Drücken und Ziehen* ein wenig nach innen versetzt. Probiert auch andere Formen aus, wie beispielsweise die Bögen, um runde Fenster zu gestalten.



## Tipp

Auf der rechten Seite findet ihr den Mentor. Er zeigt euch immer an, wie ein Werkzeug funktioniert. Wenn ihr also mal nicht wisst, wie ein Werkzeug funktioniert, dann klickt das entsprechende Werkzeug an und schaut euch dann die Beschreibung im Mentor an.



# Virtual und Augmented Reality

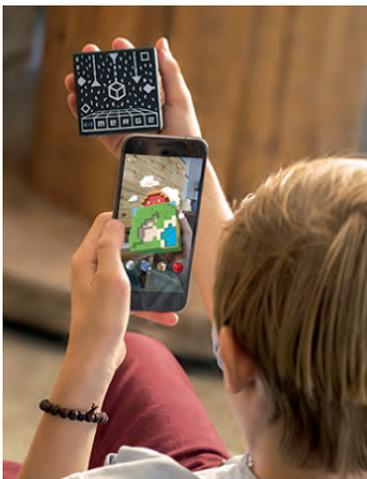
Virtual Reality (virtuelle Realität) und Augmented Reality (erweiterte Realität) sind derzeit die Trends unserer Zeit. Doch was genau verbirgt sich dahinter?

Ob mit Hilfe von 360-Grad-Videos oder programmierten Simulationen, mit Virtual Reality (VR) kann man in neue Welten eintauchen. Unter Verwendung von sogenannten VR-Brillen ist man mittendrin und kann durch Bewegen des Kopfes den Bildausschnitt bestimmen – genau wie in der Realität.



Augmented Reality ist die Erweiterung unserer realen Umgebung mit virtuellen Elementen. Das bekannteste Beispiel hierfür ist derzeit wohl Pokémon GO. In der realen Welt lassen sich virtuelle Fantasiewesen fangen.

Für das Betrachten der erstellten 3D-Modelle auf dem Smartphone gibt es viele Apps. Im Folgenden werden verschiedene Möglichkeiten vorgestellt, auch solche, mit deren Hilfe man die 3D-Modelle als Augmented und Virtual Reality ansehen kann.



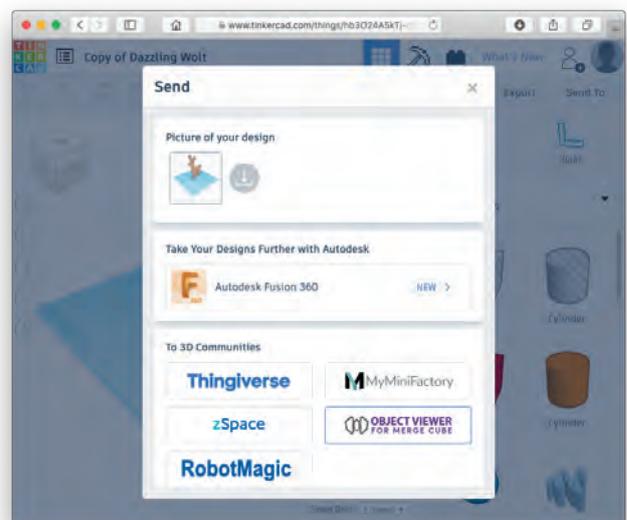
Bildquelle: Merge Labs, Inc., 2019

## Augmented Reality mit dem MERGE Cube

Mit dem MERGE Cube (<https://mergevr.com/cube/de>), einem holografischen Objekt zum Spielen und Lernen, lassen sich selbstgestaltete – bspw. mit TinkerCAD oder BlocksCAD – 3D-Modelle virtuell in die Realität holen.

Der MERGE Cube eignet sich so für eine realistische Vorschau der in diesem Modul selbst gestalteten (bspw. mit TinkerCAD oder BlocksCAD) 3D-Modelle. Im Sinne des *Rapid Prototyping* Gedankens lassen sich so verschiedene Designs ausprobieren, ohne die Modelle erst kost- und zeitspielig ausdrucken zu müssen.

Um dreidimensionale Modelle auf den MERGE Cube zu projizieren, empfiehlt es sich, sie über TinkerCAD an den *Object Viewer for Merge Cube* zu übertragen (siehe Abbildung). Über die *Send To* Schaltfläche lässt sich ein Code erzeugen, der anschließend in die kostenlose App *Object Viewer* für **Android** und **iOS** Geräte (Tablets und Smartphones) eingetragen wird.



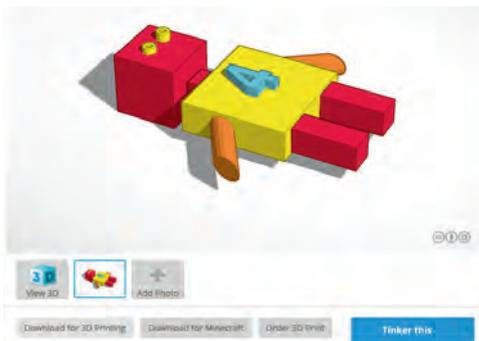
Ein *Virtual Reality* Modus für den MERGE Cube ist laut Hersteller aktuell in Vorbereitung.

## 3D-Rundgänge mit dem Holobuilder

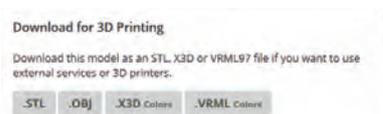
Der Holobuilder ist ein kostenloses Online-Tool mit dessen Hilfe 360°-Rundgänge und VR-Inhalte produziert werden können. Hierfür muss ein Account erstellt werden. Danach kann man beispielsweise auch SketchUp-Modelle oder TinkerCad-Modell hochladen und betrachten oder mit Hilfe von 360°-Aufnahmen einen Rundgang durch die Schule produzieren.

### SketchUp und TinkerCad für den Holobuilder

Möchte man ein SketchUp 3D-Modell hochladen, so muss dieses vorher in eine Collada-Datei exportiert werden. Hierzu geht man im Menü von Sketch Up auf Datei und anschließend auf Exportieren und wählt dort „3D-Modell“ aus. Nun kann man das Modell als Collada-Datei speichern.

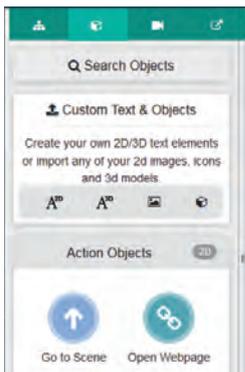


Auch TinkerCad-Modelle lassen mit Hilfe des Tools Holobuilder über eine VR-Brille betrachten. Dafür muss die Datei im OBJ-Format heruntergeladen werden. Klicken Sie dafür auf *Download for 3D Printing* und wählen Sie das Dateiformat *OBJ* aus.

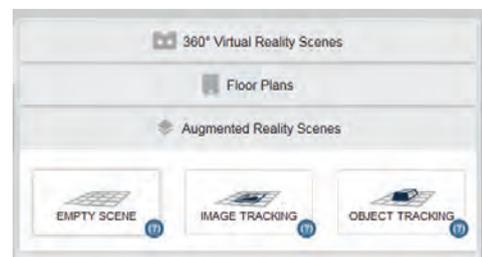


### Dateien im Holobuilder öffnen

Um Dateien zu öffnen bzw. vorhandene 3D-Modelle zu platzieren, muss zu Beginn *Augmented Reality Scene* und dann *Empty Scene* ausgewählt werden.



Im Anschluss daran wählt man rechts im Menü die Leiste *Custom Text & Objects* aus und klickt ganz rechts auf auf den kleinen Quader (*Add a custom 3D Model*)



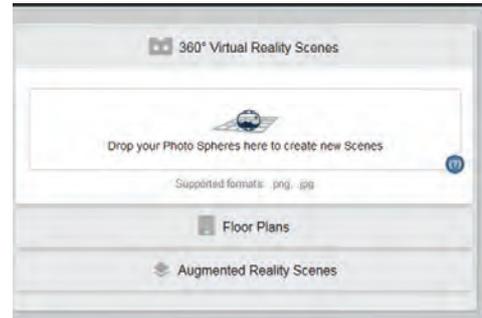
Nun öffnet man die gewünschte Datei und importiert sie. Das Tinkercad-Modell oder das SketchUp-Modell ist nun im Holobuilder. Mit Hilfe des Share-Buttons erhält man einen QR-Code, den man wieder mit dem Smartphone abschnappen kann. Nun kann man sich das Modell als Augmented oder Virtual Reality ansehen.



## Schulrundgang mit dem Holobuilder

Um einen Rundgang durch die Schule zu produzieren, benötigt man zu Beginn 360°-Bilder. Diese können mit Hilfe verschiedener Apps erstellt werden, z.B. mit der App 360 Panorama.

Auf der Plattform „Holobuilder“ können die 360°-Bilder hochgeladen werden. Klicken Sie dazu zu Beginn auf *360° Virtual Reality Scene* und wählen sie das entsprechende Bild auf ihrem Computer aus.



In der linken Menüleiste hat man nun die Möglichkeit noch weitere 360-Bilder hochzuladen, wodurch Rundwege möglich werden. Beispielsweise kann man Bilder



vom Klassenraum erstellen, vom Flur, der Aula, vom Schulhof etc. und die Bilder mit dem Holobuilder miteinander verbinden.

Ist man mit seinem Rundgang fertig, so kann auch er auch wieder mit Hilfe eines QR-Codes geteilt und auf dem Smartphone mit Hilfe einer Brille betrachtet werden.

### Links:

- ∞ Holobuilder: [www.holobuilder.com](http://www.holobuilder.com) (optimiert für Google Chrome)
- ∞ How to build a Holo with your smartphone: <https://www.youtube.com/watch?v=Amkanbnz5iw>
- ∞ 360 Panorama: <https://play.google.com/store/apps/details?id=com.vtcreator.roid360&hl=de>

## Google Cardboard

### 1.1 Funktionsweise und -umfang

Eine virtuelle Realität (**Virtual Reality**, VR) ist ein von einem Computer erzeugtes Modell einer Welt, in die sich bspw. mit VR-Brillen „eintauchen“ lässt.

Das **Google Cardboard** ist ein einfaches VR-Gerät, das du selbst zusammenbasteln kannst! Mithilfe deines Smartphones, ein bisschen Pappe und gegebenenfalls zwei Linsen lässt sich so schnell eine günstige Alternative zu professionellen VR-Brillen herstellen.

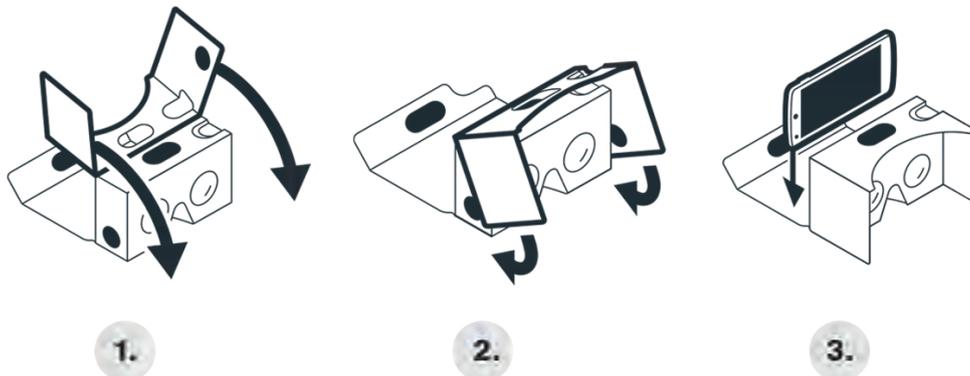


Bildquelle: [https://commons.wikimedia.org/wiki/File:Assembled\\_Google\\_Cardboard\\_VR\\_mount.jpg](https://commons.wikimedia.org/wiki/File:Assembled_Google_Cardboard_VR_mount.jpg)

Durch die mit beiden Augen getrennte Wahrnehmung eines geteilten Bildinhalts wird der Eindruck erweckt, man sei mitten im Geschehen. In einigen Apps kann durch den Lagesensor des Smartphones außerdem der Blickwinkel auf die Situation geändert werden.

Unter <https://arvr.google.com/cardboard/> findest du Bausätze und verschiedene Apps, mit denen du das Cardboard nutzen kannst.

Wenn ihr das IT2School-Material zu diesem Modul habt, dann könnt ihr die Pappbrillen nach dem folgenden Schema zusammenbauen.



Bildquelle: <https://vr.zdf.de>

Zur ersten Nutzung ist die VR-App des ZDF empfehlenswert, die ihr über <https://vr.zdf.de> kostenlos herunterladen könnt.

Auch auf YouTube gibt es zahlreiche Videos mit VR-Funktion. Sucht dazu einfach in der YouTube-App auf eurem Smartphone nach „360°“. Eindrucksvolle Bilder sind unter anderem auf dem Kanal von „National Geographic“ zu finden. Klickt nach der Wiedergabe des von euch ausgesuchten Videos auf das Cardboard-Symbol (siehe rechtes Icon).



### 1.2 Einrichtung

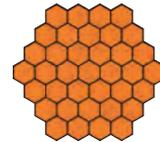
Einige Apps benötigen Informationen zu dem konkret verwendeten Cardboard. Wenn ihr das Cardboard aus dem haptischen Material zu dem IT2School-Modul B4 vorliegen habt, dann könnt ihr an der entsprechenden Stelle in den Apps einfach den links abgebildeten QR-Code scannen, um die Daten zu übermitteln.

# Parametrisches Design mit BlocksCAD

In der Natur findet man beispielsweise bei Eiskristallen, Blumen oder Bienenwaben **organische Formen** vor, die durch ihre symmetrischen Muster nicht nur hohe Stabilitäten aufweisen, sondern zugleich auch faszinierend aussehen.



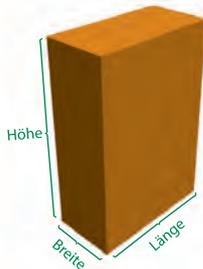
**Moderne Architektur** – wie sie z. B. beim Nationalen Schwimmzentrum in Peking (links) oder bei der begehbaren Achterbahn in Duisburg (unten) zu bestaunen ist – greift diese Muster aus der Natur auf ästhetische Art und Weise auf.



Um solche Formen effektiv *computergestützt designen* (engl. *Computer Aided Design* oder abgekürzt *CAD*) zu können, kann man auf sogenanntes **Parametrisches Design** zurückgreifen. *Parameter* beschreiben dabei variable Größen, die charakteristisch für bestimmte Objekte sind.

So wird bspw. ein quaderförmiges Objekt durch die **Parameter Länge, Breite und Höhe** und ein

würfel x 10 y 20 z 30



Zylinder durch seinen *Radius* und seine *Höhe* definiert. Durch Anpassung dieser Parameter lassen sich aus wenigen Grundkörpern (Kugeln, Würfel<sup>1</sup>, Zylinder, Ringe) alle erdenklichen Körper formen.



zylinder radius 10 höhe 5

Auch *Abstand* oder *Winkel* zwischen Objekten lassen sich mit Parametern beschreiben. Durch eine Anpassung von einzelnen Parametern kann man also geschickt grundlegende Eigenschaften (bspw. die Anzahl und Größe der Waben in der Außenhaut des Schwimmzentrums) verändern, ohne kleinschrittig alle einzelnen Teile (bspw. alle Waben einzeln) anpassen zu müssen. Parametrisches Design ermöglicht also beispielsweise auch eine einfache Umsetzung von organischen Formen beim Entwurf von Gebäuden.

Ein Tool, mit dem sich Parametrisches Design umsetzen lässt, heißt *BlocksCAD* ([www.blocks-cad3d.com](http://www.blocks-cad3d.com)). Mit den folgenden Aufgaben sollt ihr *BlocksCAD* am Beispiel der Modellierung einer Blume kennenlernen.

Für Aufgaben, die mit einem 😊 markiert sind liegen für euch

Hilfekarten bereit, auf die ihr einen Blick werfen dürft, nachdem ihr die Aufgabe selbst versucht habt zu lösen.



<sup>1</sup> In *BlocksCAD* können verschiedene Quader mit dem Programmierbaustein *Würfel* erzeugt werden.

## Aufgabe 1: Formen

- 😊 a) Verwendet den *Kugel*-Block und modelliert eine Kugel mit einem Radius von 18 Längeneinheiten.
- 😊 b) Modelliert nun zusätzlich eine *Kiste*, die eine Höhe von 40, eine Länge von 20 und eine Breite von 10 Längeneinheiten hat.
- ∞ Welchen Block benötigt ihr dafür?
  - ∞ Welche Seite gehört zu welcher Achse im Koordinatensystem?
  - ∞ Was passiert, wenn ihr den Wert „nicht zentriert“ auf „zentriert“ setzt und anschließend das Modell neu rendert?
- 😊 c) Entfernt nun die beiden bisher verwendeten Blöcke aus eurer Arbeitsfläche. Das funktioniert, indem ihr sie mit der rechten Maustaste anklickt und „Baustein löschen“ auswählt. Untersucht anschließend den *Zylinder*-Block: Beschreibt ...
- ∞ ... was passiert, wenn ihr für den Parameter *Radius 2* einen Wert wählt, der sich vom Parameter *Radius 1* unterscheidet?
  - ∞ ... welchen Einfluss hat das Schloss auf die Eingabe der Parameter für *Radius 1* und *Radius 2*?



## Aufgabe 2: Transformationen

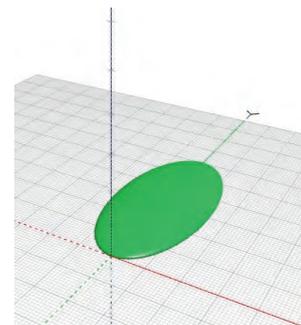
Startet mit einer leeren Arbeitsfläche und zieht einen *Kugel*-Block (Radius ca. 10 bis 20 Längeneinheiten) hinein.

- 😊 a) Streckt nun diese Kugel, sodass sie die Form eines Ostereis hat. Hinweis: Dazu muss der z-Parameter des *skalieren*-Befehls auf einen Wert gesetzt werden, der größer als 1 ist. (Achtet darauf, dass ihr bei *BlocksCAD* einen Punkt statt eines Kommas eingeben müsst, z.B. 6.3 statt 6,3)
- 😊 b) Beschreibt was passiert, wenn ihr für den z-Parameter einen Wert einsetzt, der kleiner als 1 ist (z.B. 0.2)?
- 😊 c) Verzerrt die Kugel nun zusätzlich auch noch entlang der y-Achse, sodass sie die Form eines flach liegenden Blumenblattes hat.



Eine grundlegende Eigenschaft vieler parametrischer Design ist, dass sie symmetrisch (um den Koordinatenursprung) angeordnet sind.

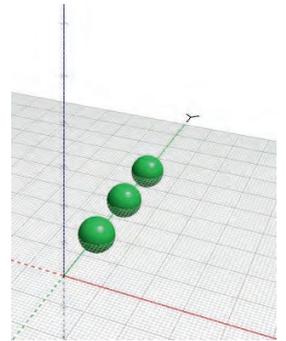
- 😊 d) Verwendet das Blatt, das ihr unter c) gestaltet habt.
- ∞ Welchen Block benötigt ihr, um es entlang der y-Achse zu verschieben?
  - ∞ Wie weit müsst ihr das Blatt entlang der y-Achse verschieben, damit es an den Koordinatenursprung aneckt (siehe Bild)?



## Aufgabe 3: Schleifen und Variablen

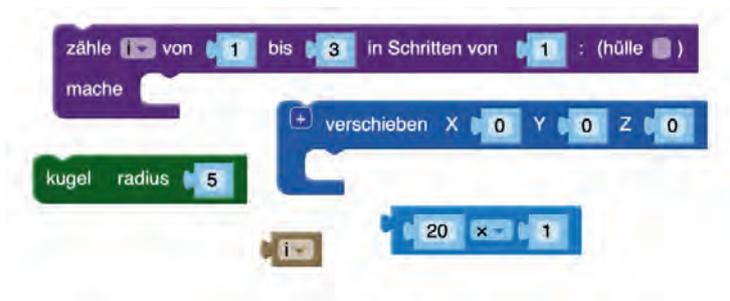
Deaktiviert nun für's Erste euer Batt aus Aufgabe 2d. Klickt dazu mit der rechten Maustaste auf die Blöcke und wählt den Punkt „Baustein deaktivieren“ aus.

- 😊 a) Ordnet drei Kugeln mit einem Radius von 5 Längeneinheiten so an, dass sie alle 20 Längeneinheiten voneinander entfernt entlang der y-Achse „aufgereiht“ sind (siehe rechte Abbildung).



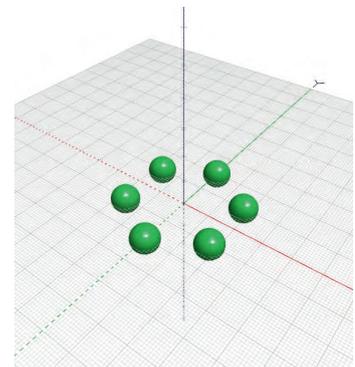
Statt sich wiederholende Anordnungen wie die rechts abgebildete über einzelne Blöcke umzusetzen, kann man in *BlocksCAD* auch sogenannte **Schleifen** benutzen. Mit einer Schleife lässt sich eine Anweisung (bspw. eine Kugel zu modellieren) beliebig oft wiederholen. Um die einzelnen Durchläufe einer Schleife voneinander zu unterscheiden greift man auf **Variablen** zurück.

- 😊 b) Setzt die folgenden Blöcke so zusammen, dass sie die Kugeln wie oben rechts abgebildet anordnet.



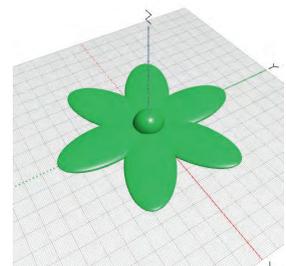
- c) Welche Parameter müsst ihr nun lediglich ändern, um ...
- ∞ ... die Anzahl der Kugeln zu erhöhen?
  - ∞ ... ihren Abstand voneinander anzupassen?
  - ∞ ... den Radius aller Kugeln zu ändern?

- 😊 d) Ändert euren Code und ergänzt ihn um einen *rotieren*-Block, sodass eine kreisförmige (etwa wie in der rechten Abbildung) erzeugt wird.



Hinweis: Diesmal unterscheiden sich die einzelnen Schleifen-Durchläufe voneinander. Die Variable *i* muss also nun in den *rotieren*-Block eingebaut werden!

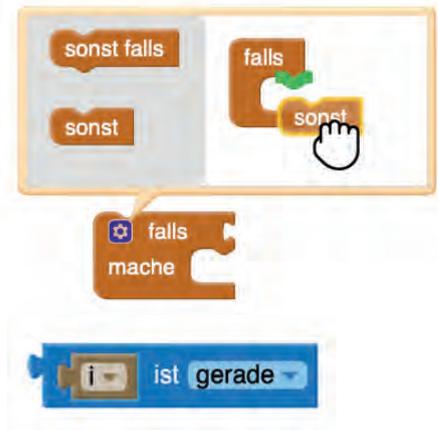
- ∞ Wie hängt der Winkel zwischen den einzelnen Kugeln von der Anzahl der Kugeln ab?
- 😊 e) Ersetzt nun den *Kugel*-Block aus diesem Code mit dem Blatt aus Aufgabe 2d, das ihr deaktiviert habt. Hinweis: Ein *verschieben*-Block wird dann überflüssig.
- ∞ Was passiert?
  - ∞ Ändert die Parameter so ab, dass sich eine Blume ergibt, die ungefähr die rechts abgebildete Form hat.
  - ∞ Ergänzt auch eine Knospe (*Kugel*-Block)!
- 😊 f) An welcher Stelle müsst ihr einen Parameter ändern, damit die Blätter sich aufrichten?



## Aufgabe 4 (optional!): Verzweigungen

Die einzelnen Schleifendurchläufe lassen sich auch in Abhängigkeit von der Variable  $i$  gestalten. Hierzu gibt es unter dem Abschnitt *Logik* einen *falls...-mache...-Block*, der sich durch einen Klick auf das Zahnrad auf um einen *sonst*-Abschnitt ergänzen lässt (siehe rechts).

So lässt sich z. B. über Hinzufügen eines *i-ist-gerade*-Blocks (siehe rechts) eine Blume gestalten, bei der jedes zweite Blatt bspw. schmaler ist, als die anderen.



- 😊 a) Passt euren Code unter Verwendung der oben angegebenen Bausteine so an, dass die Blätter an geraden Stellen eine andere Form haben als die an ungeraden (wie bspw. bei der Blume auf dem Bild).

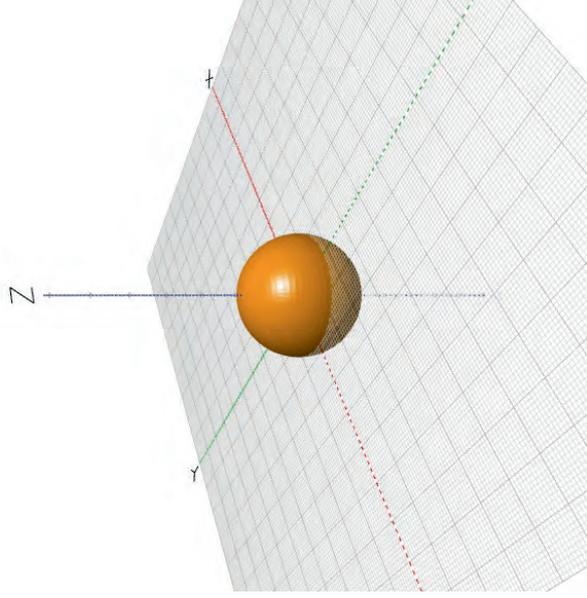


## Aufgabe 5: Mengenoperationen

- Die Knospe und die Blätter gehören zusammen und ergeben eine Blume. In *BlocksCAD* lassen sich diese Teile ebenfalls *vereinigen*. Dazu gibt es unter dem Menüpunkt „Mengenoperationen“ einen *Vereinigung*-Block. Verwendet ihn, um die Blätter mit der Knospe zu verbinden.
- Was passiert, wenn ihr statt des *Vereinigungs*-Blocks den *Differenz*-Block auswählt?
- Erkundet auf dieselbe Art und Weise auch den *Schnittmenge*- sowie den *Hülle*-Block.

# 1 a) Formen: Kugel

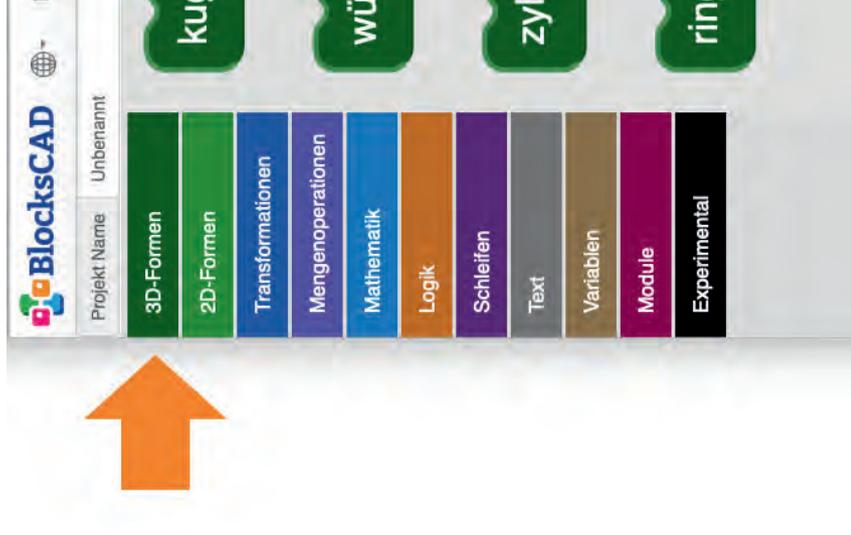
So soll es aussehen:



Folgenden Block benötigt ihr dafür:

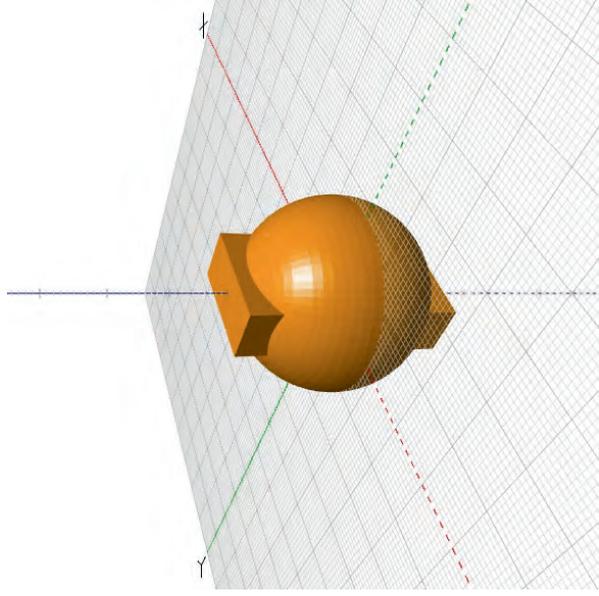


Hier findet ihr ihn:

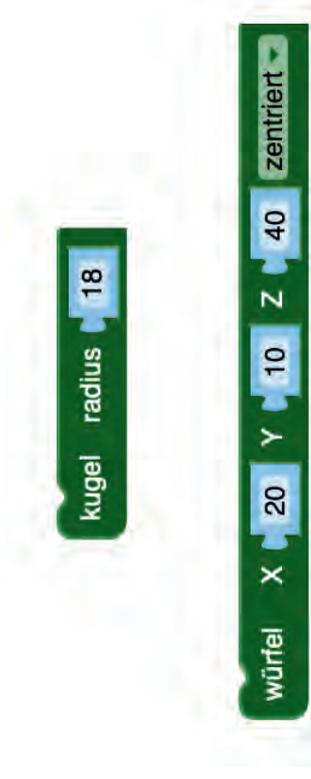


## 1 b) Formen: Kugel und Würfel

So soll es aussehen:



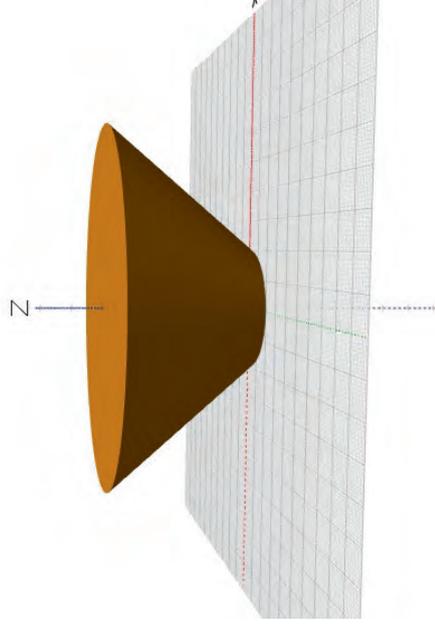
Folgende Blöcke benötigt ihr dafür:



- ∞ Wir Menschen bewegen uns im Raum normalerweise entlang der x- und y-Achse.
- ∞ Die z-Achse ist dann die, die nach oben bzw. unten zeigt.
- ∞ Die Höhe einer Kiste wird also üblicherweise über den z-Parameter definiert.
- ∞ Setzt man den Würfel auf „zentriert“, so wird sein Mittelpunkt (genau genommen sein geometrischer Schwerpunkt) in den Koordinatenursprung gelegt.

## 1 c) Formen: Zylinder

Das passiert:



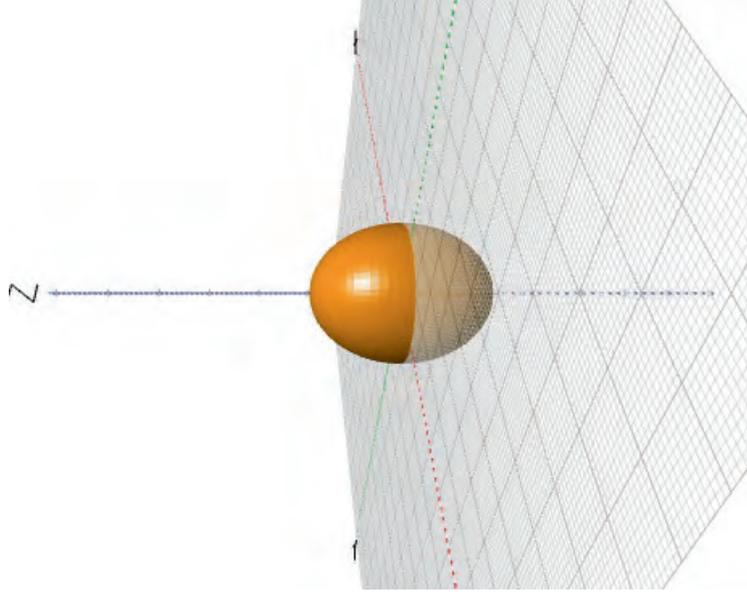
Erklärung:

- ∞ Wenn man für den Parameter für *Radius 1* einen Wert setzt (z. B. 20), der sich von *Radius 2* unterscheidet (z. B. 60), dann ändert sich der Radius des Zylinders mit zunehmender Höhe von Radius 1 auf Radius 2.
- ∞ Dazu muss das Schloss geöffnet sein
- ∞ Bei geschlossenem Schloss werden die beiden Radien aneinander angepasst

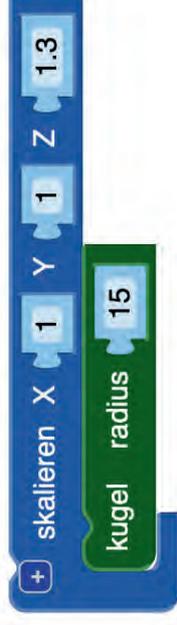


## 2 a) Transformationen: Skalieren

So kann es aussehen:



Folgende Blöcke benötigt ihr dafür:

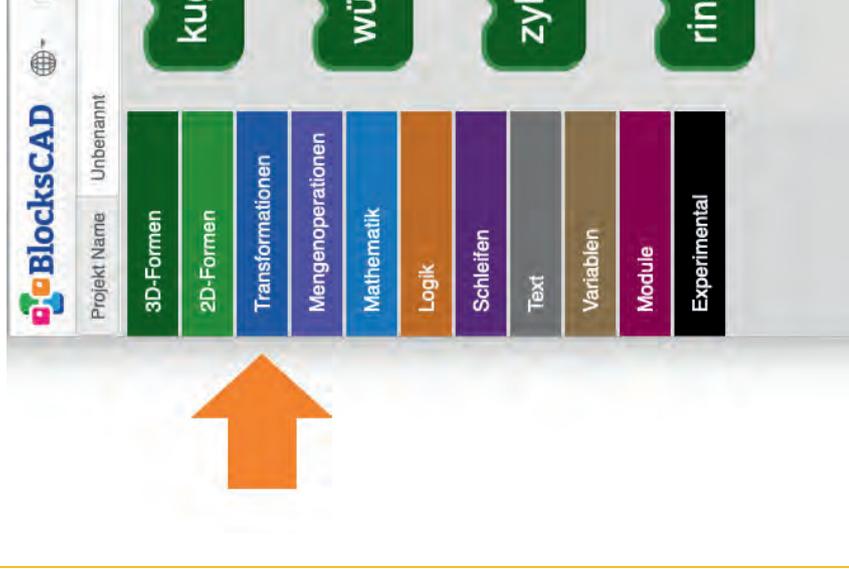


Durch die Verbindung des *skalieren*-Blocks mit dem *Kugel*-Block wird die Kugel skaliert.

Man sagt auch: „Die *skalieren*-Operation bzw. *skalieren*-Funktion wird auf die *Kugel* angewandt.“

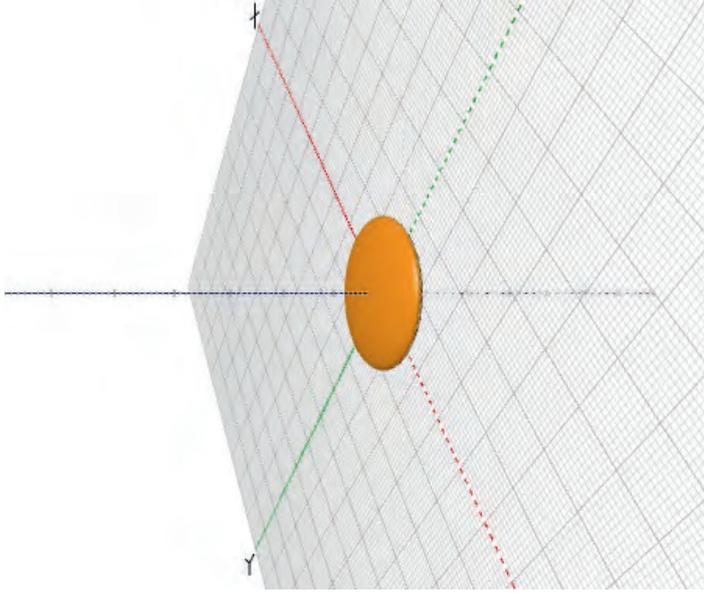
Wenn dieser Wert **größer** ist **als 1** wird die Kugel also **gestreckt**.

Hier findet ihr den *skalieren*-Block:

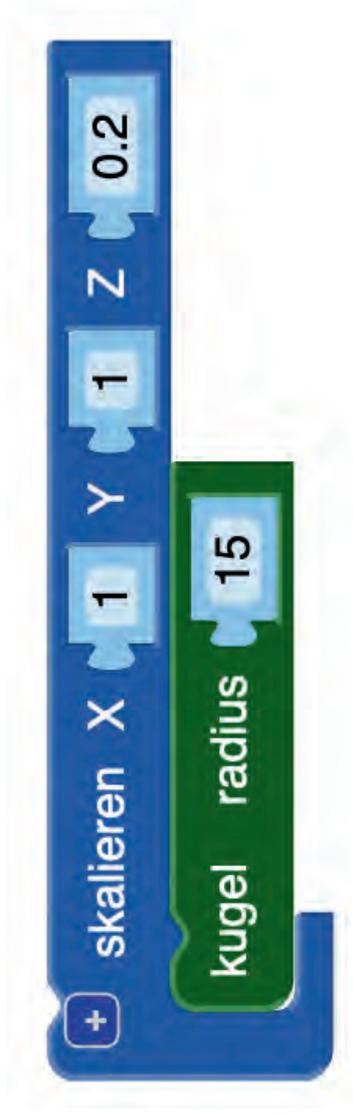


## 2 b) Transformationen: Skalieren

So kann es aussehen:



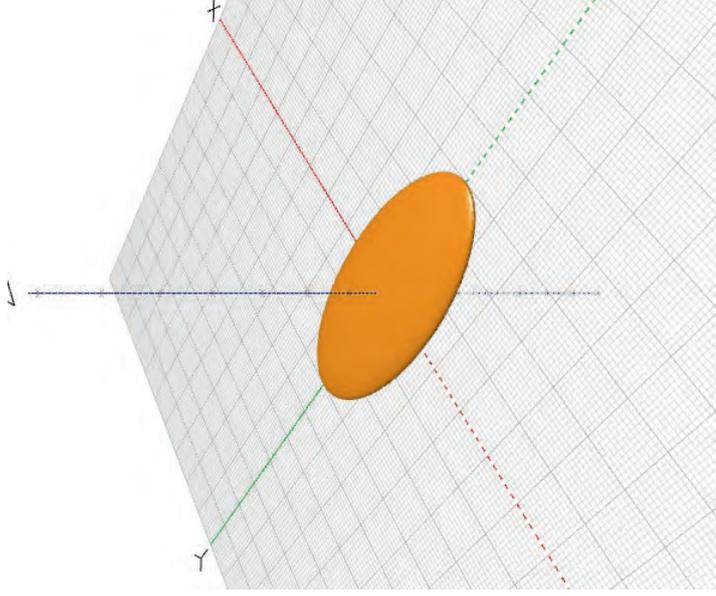
Folgende Blöcke benötigt ihr dafür:



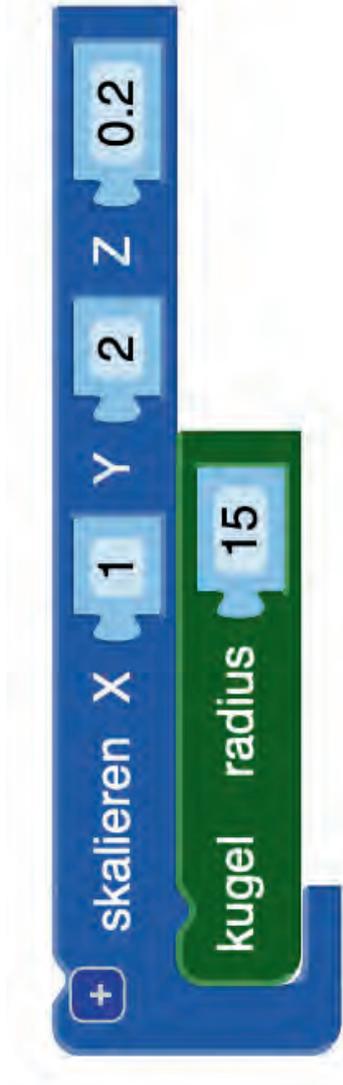
Wird für den z-Parameter ein Wert eingesetzt, der **kleiner** ist als **1** (also z. B. 0.2), dann wird die Kugel **gestaucht**.

## 2 c) Transformationen: Skalieren

So kann es aussehen:



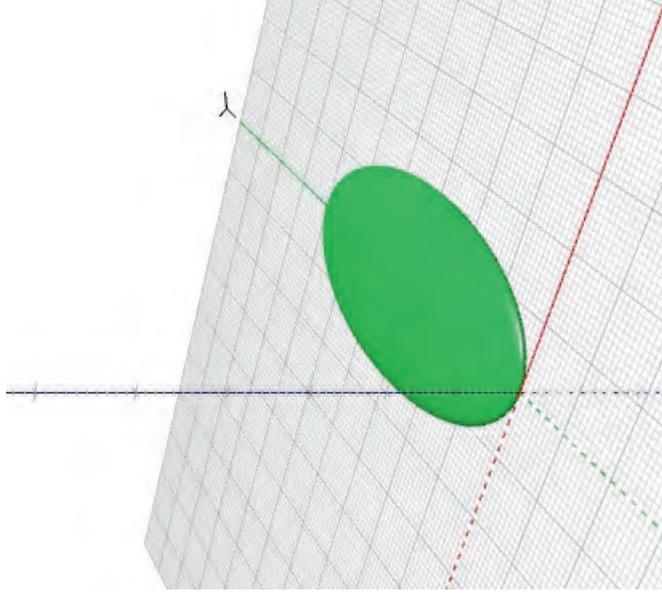
Folgende Blöcke benötigt ihr dafür:



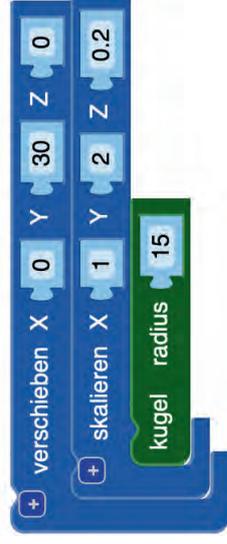
Körper lassen sich also auch entlang mehrerer Achsen strecken bzw. stauchen.

## 2 d) Transformationen: Skalieren

So kann es aussehen:



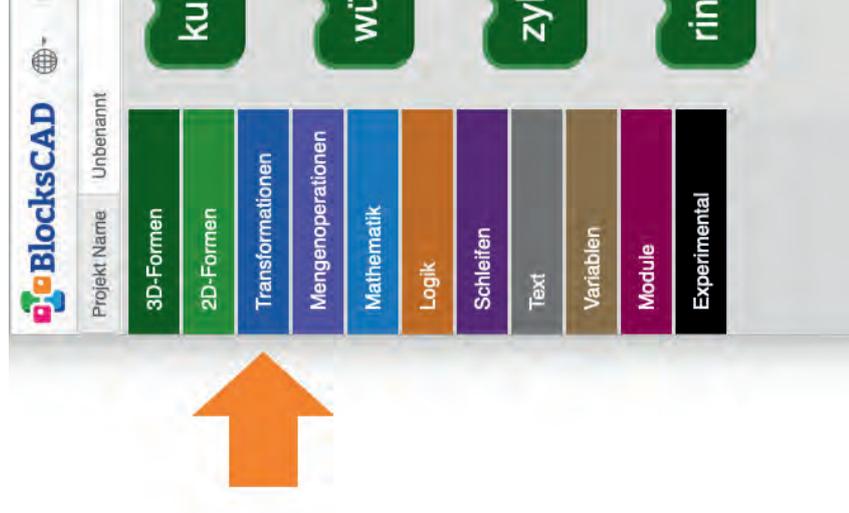
Folgende Blöcke benötigt ihr dafür:



Erklärung des Beispiels:

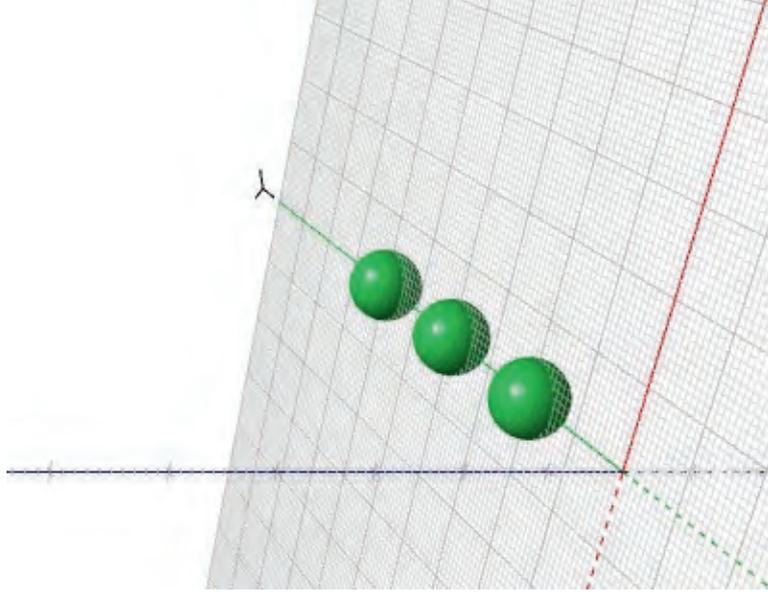
Nachdem die Kugel mit dem Radius 15 um den Faktor  $2 = 30$  gestreckt wurde, beträgt ihr Radius  $15 \cdot 2 = 30$  Längeneinheiten (LE). Sie hat also einen Durchmesser von 60 LE (sie ist 60 LE „lang“). Damit ihr Ende an den Koordinatenursprung ansteckt, muss sie also um  $\frac{60}{2}$  LE entlang der y-Achse verschoben werden.

Hier findet ihr den verschoben-Block:

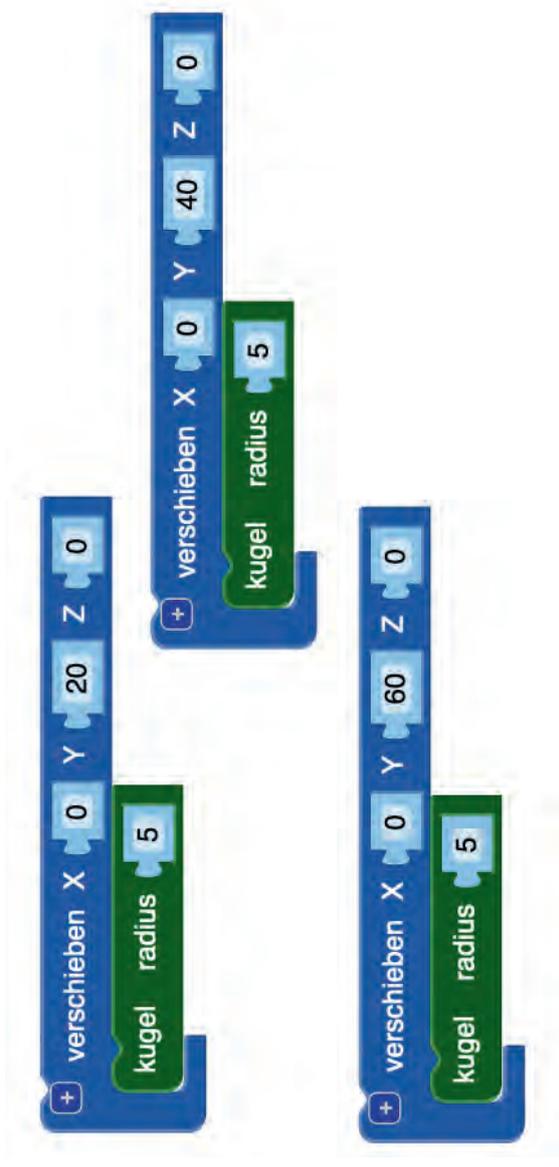


### 3 a) Schleifen und Variablen: Kugeln einzeln

So soll es aussehen:

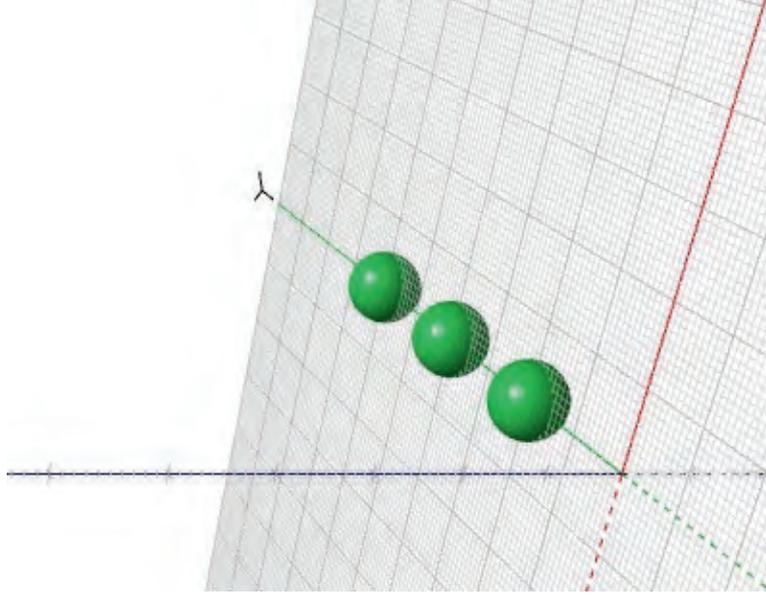


Folgende Blöcke benötigt ihr dafür:

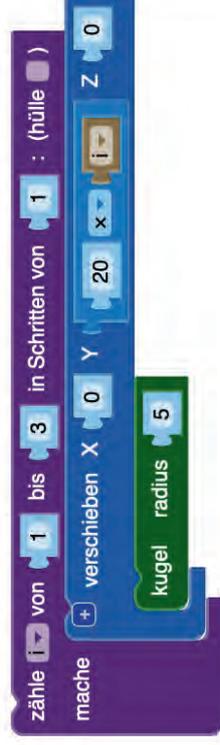


### 3 b) Schleifen und Variablen: Verschieben

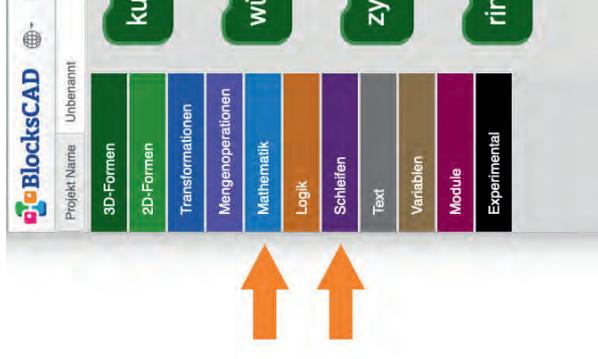
So soll es aussehen:



Folgende Blöcke benötigt ihr dafür:

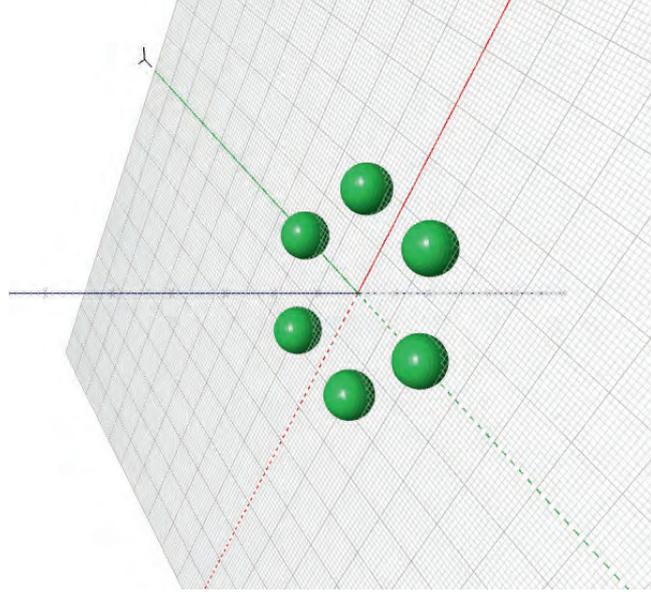


Hier findet ihr den *verschieben*-Block:

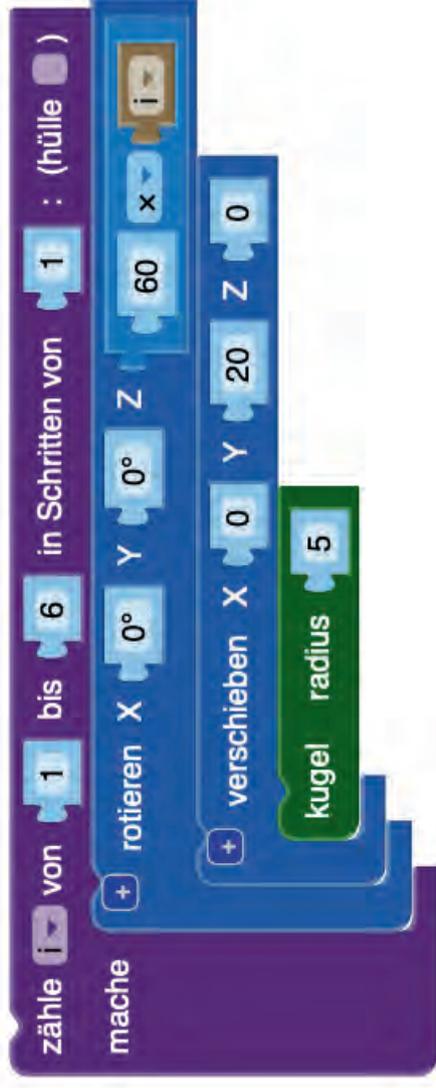


### 3 d) Schleifen und Variablen: Rotieren

So soll es aussehen:



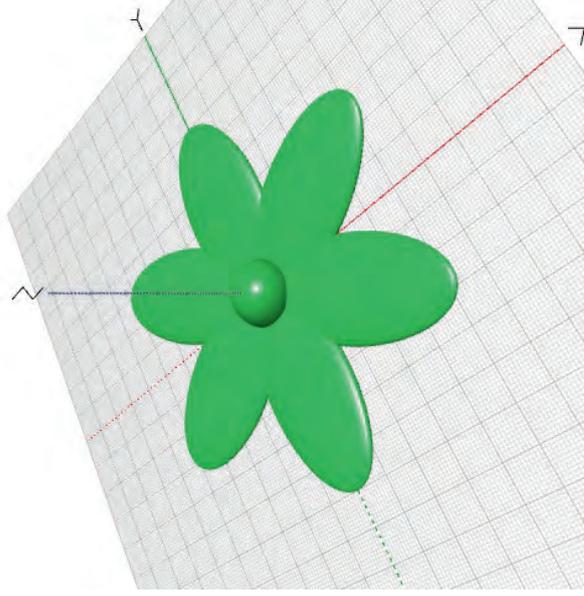
Folgende Blöcke benötigt ihr dafür:



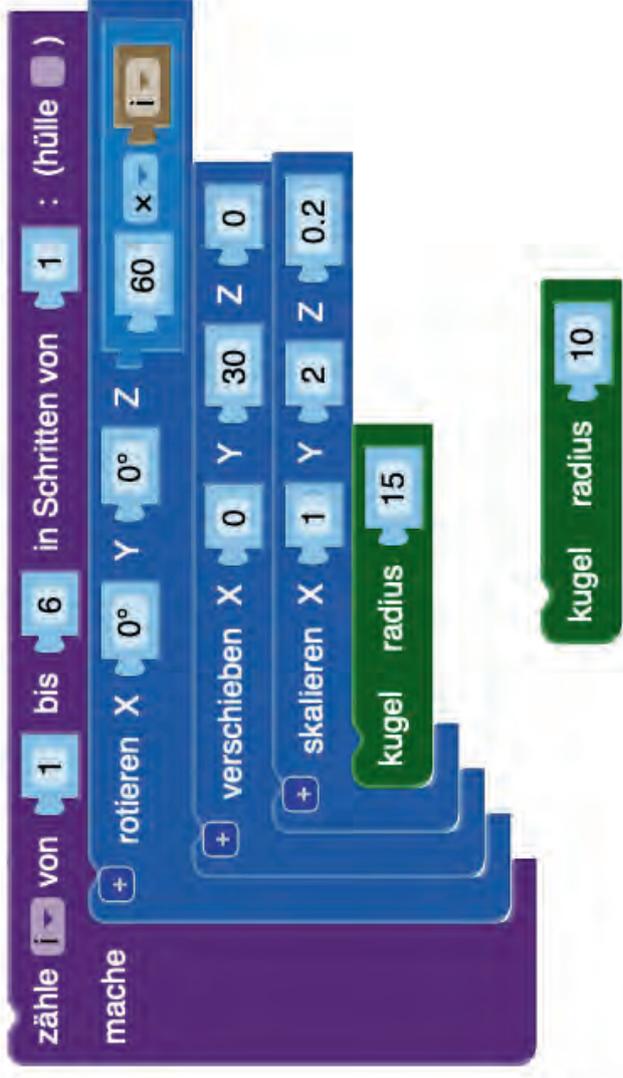
Ein Kreis hat einen Vollwinkel von  $360^\circ$ . Bei 6 Blättern (wie in der Abbildung) haben die einzelnen Blätter also einen Abstand von  $\frac{360^\circ}{6} = 60^\circ$  voneinander.

### 3 e) Schleifen und Variablen: Blume

So kann es aussehen:

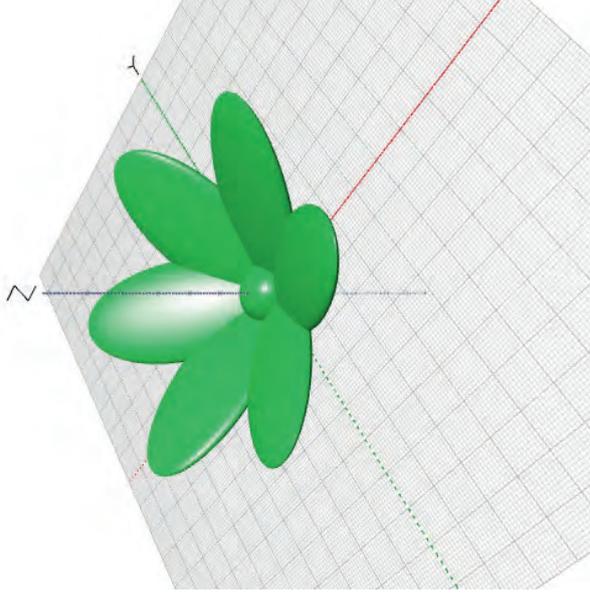


Folgende Blöcke benötigt ihr dafür:

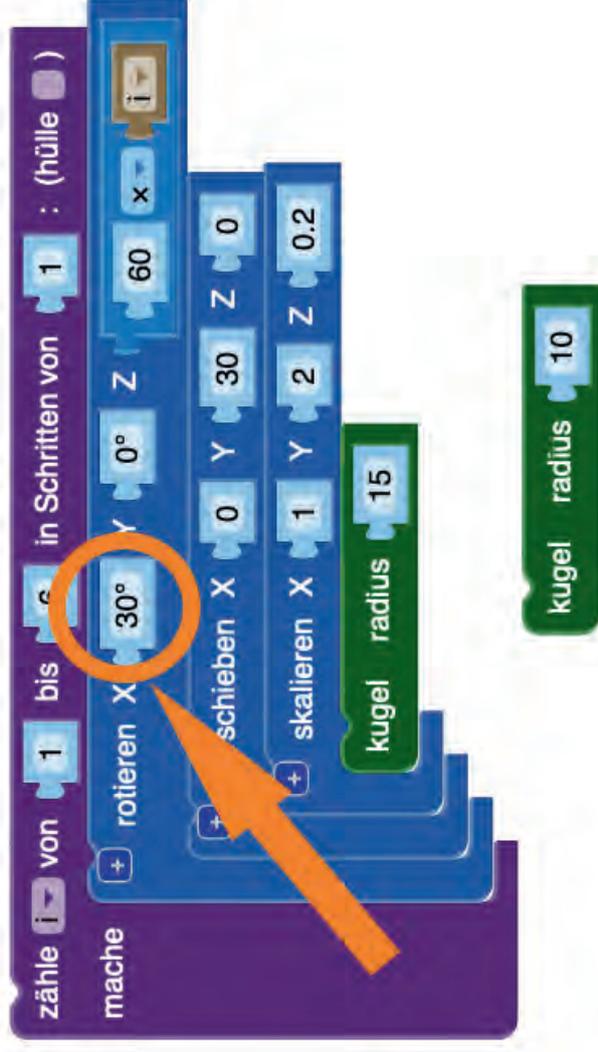


### 3 f) Schleifen und Variablen: Blume

So kann es aussehen:

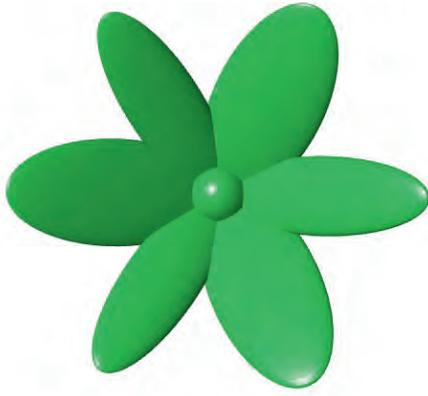


Folgende Blöcke benötigt ihr dafür:

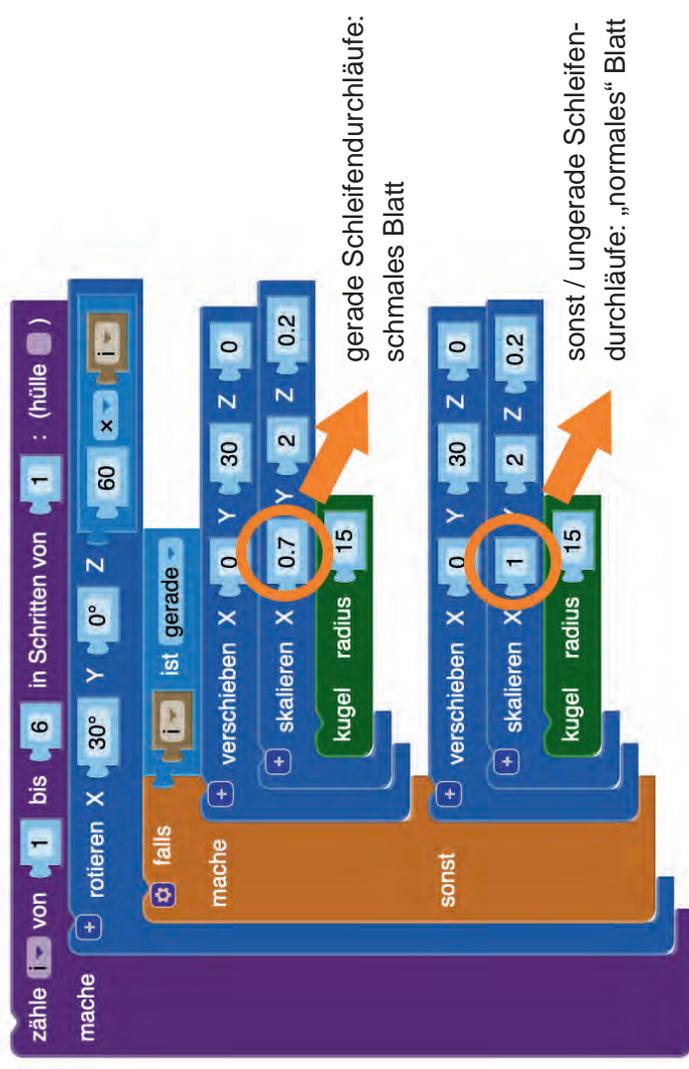


## 4 a) Verzweigungen

So kann es aussehen:



Folgende Blöcke benötigt ihr dafür:

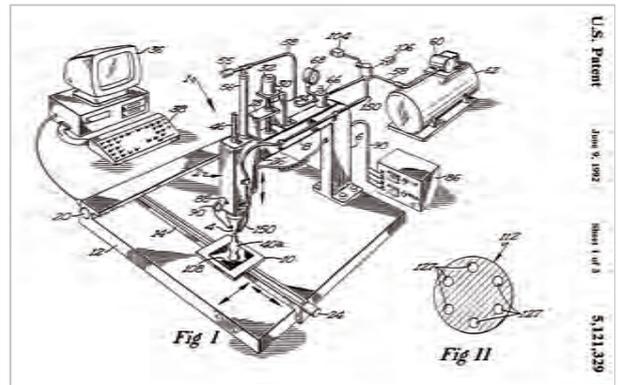


# Aufbau und Funktionsweise von 3D-Druckern

Was ist 3D-Druck und was kann man damit machen

Mit dem Begriff „3D-Drucken“ bezeichnet man Fertigungsverfahren von dreidimensionalen Werkstücken, die durch das schichtweise Auftragen von Material entstehen.

Auch wenn diese Technologie erst jetzt Einzug in die Klassenzimmer findet, sind die Grundlagen und Patente bereits über 50 Jahre alt. In der rechten Abbildung ist ein Ausschnitt aus einem Patent aus dem Jahr 1992 dargestellt. Es wird eine Maschine beschrieben, die mit Hilfe eines Computers einen Druckkopf über einem Druckbett bewegt und geschmolzenen Kunststoff Schicht für Schicht aufträgt. Erst als im Jahr 2009 dieses Patent abgelaufen ist, wurde der Weg für die heutigen Hobby 3D-Drucker geebnet.



U.S. Patent  
 5,121,329  
 Filed 1/19/92  
 Inventor: Charles W. Hull

Während in den 1980er ein 3D-Drucker noch über 300.000\$ gekostet hat, sind heutige Einsteiger Modelle bereits für unter 200€ verfügbar.

## Aufgabe 1: Verschiedene Anwendungen und dem Nutzen von 3D-Druck

In der folgenden Tabelle sind vier Personen in vier unterschiedlichen Berufen aufgelistet. Beschreibe welche Aufgaben die Personen in ihren Berufen erfüllen und inwiefern 3D-Druck sie dabei unterstützen kann.

Name	Peter	Eva	Julian	Nephele
Beruf	Modellbauer	Medizintechnikerin	Maschinenbauingenieur	Architektin
Aufgaben				
Interessiert, Folgendes zu drucken				

Eine Entwicklung von OFFIS e.V. in Kooperation mit der Universität Oldenburg  
 im Auftrag der Wissensfabrik – Unternehmen für Deutschland e.V.

## Aufgabe 2: Funktionsweise von 3D-Druckern

Vervollständige den folgenden Lückentext.

Es gibt viele verschiedene Arten von 3D-Druckern und Fertigungsverfahren. In einigen Verfahren wird eine harzähnliche Flüssigkeit mit UV-Licht (DLP-Verfahren) oder Lasern (SLA-Verfahren) gezielt ausgehärtet oder Pulver durch Laserstrahlen (SLS-Verfahren) geschmolzen. Andere Verfahren schmelzen \_\_\_\_\_ und pressen diese dann durch eine schmale Düse, um so aus dünnen Schichten Stück für Stück ein 3D-Modell herzustellen. Viele der Verfahren werden im Ingenieursbereichen, wie zum Beispiel \_\_\_\_\_ oder Maschinenbau, verwendet, um schnell \_\_\_\_\_ zu fertigen und zu testen. In den letzten Jahren wurde das sogenannte \_\_\_\_\_ (oder kurz FFF) vor allem im Hobby- und Bildungsbereich immer beliebter und die Geräte auch für Privatpersonen erschwinglich. Aber wie funktioniert so ein 3D-Drucker, der Kunststoffe schmelzen kann, eigentlich?

### Vorbereitung:

Um etwas drucken zu können, muss zunächst ein 3D-Modell erstellt (oder aus dem Internet heruntergeladen) und abgespeichert werden. Gängige Dateiformate sind STL oder 3MF. Diese Dateien werden mit einem speziellen Programm, dem sogenannten \_\_\_\_\_, geöffnet. Das Programm wandelt das 3D-Modell in Steuerbefehle für den 3D-Drucker um und fügt unter Umständen Stützstrukturen für Überhänge hinzu. Diese Befehle werden als G-code bezeichnet. Im Gegensatz zu Dateiformaten, wie STL und 3MF, werden G-code Befehle schon seit den 1950er Jahren zur Steuerung von Fertigungsmaschinen genutzt. Mit Hilfe einer \_\_\_\_\_ oder einem USB-Stick kann die G-code-Datei auf den Drucker übertragen werden. Moderne 3D-Drucker können die Datei auch über das heimische Netzwerk direkt aus dem Slicer empfangen. Mit Hilfe eines Bedienfeldes kann die Datei ausgewählt und der Druck gestartet werden. In der G-code Datei steht unter anderem, welche mit welchen Temperaturen gedruckt werden soll und wohin sich der Druckkopf bewegt.

### Druck:

Statt Tinte verwenden FFF-Drucker \_\_\_\_\_ zum Drucken. Je nach Anforderung können verschiedene Kunststoffe verwendet werden. Besonders beliebt ist dabei das Material PLA (Polylactide), weil es sich bei vergleichsweise einfach und bei relativ niedrigen Temperaturen (200°C bis 220°C) drucken lässt. PLA ist eine Polymilchsäure, die aus Maisstärke gewonnen werden kann und unter bestimmten Bedingungen auch biologisch abbaubar ist.

Das Filament wird aufgewickelt in Spulen geliefert. Der sogenannte \_\_\_\_\_ befördert (oder extrudiert) das Material von der Spule zum Druckkopf. Der Druckkopf wird mit Hilfe von zwei Motoren, Riemen und Schlitten, die sich auf stabilen \_\_\_\_\_ befinden in \_\_\_\_\_-Richtung bewegt. Im Druckkopf befindet sich neben den Lüftern ein zentrales Bauteil eines 3D-Druckers: das \_\_\_\_\_. Im Hotend wird das Material erhitzt und durch eine \_\_\_\_\_ (auch Nozzle genannt) gepresst. Am unteren Ende des Hotends befinden sich ein Heizelement zum Erhitzen der Düse und ein \_\_\_\_\_ (Thermistor) zur Überwachung der Temperatur. Ein typischer Durchmesser einer Nozzle sind 0,4mm.

Das geschmolzene Material wird dann Schicht für Schicht auf dem \_\_\_\_\_ aufgetragen. Zum Aushärten wird das frisch aufgetragene Filament mit einem \_\_\_\_\_ abgekühlt. Sobald eine Schicht fertig ist, bewegt ein Schrittmotor das Druckbett in \_\_\_\_\_-Richtung mit einer \_\_\_\_\_ nach unten. Das Auftragen der Schichten und Absenken der Druckplatte wird so lange wiederholt, bis das Modell fertig ist. Je nach Modell, Düse, Schichthöhe, Füllung und Druckgeschwindigkeit kann dies wenige Minuten, aber auch mehrere Tage dauern.

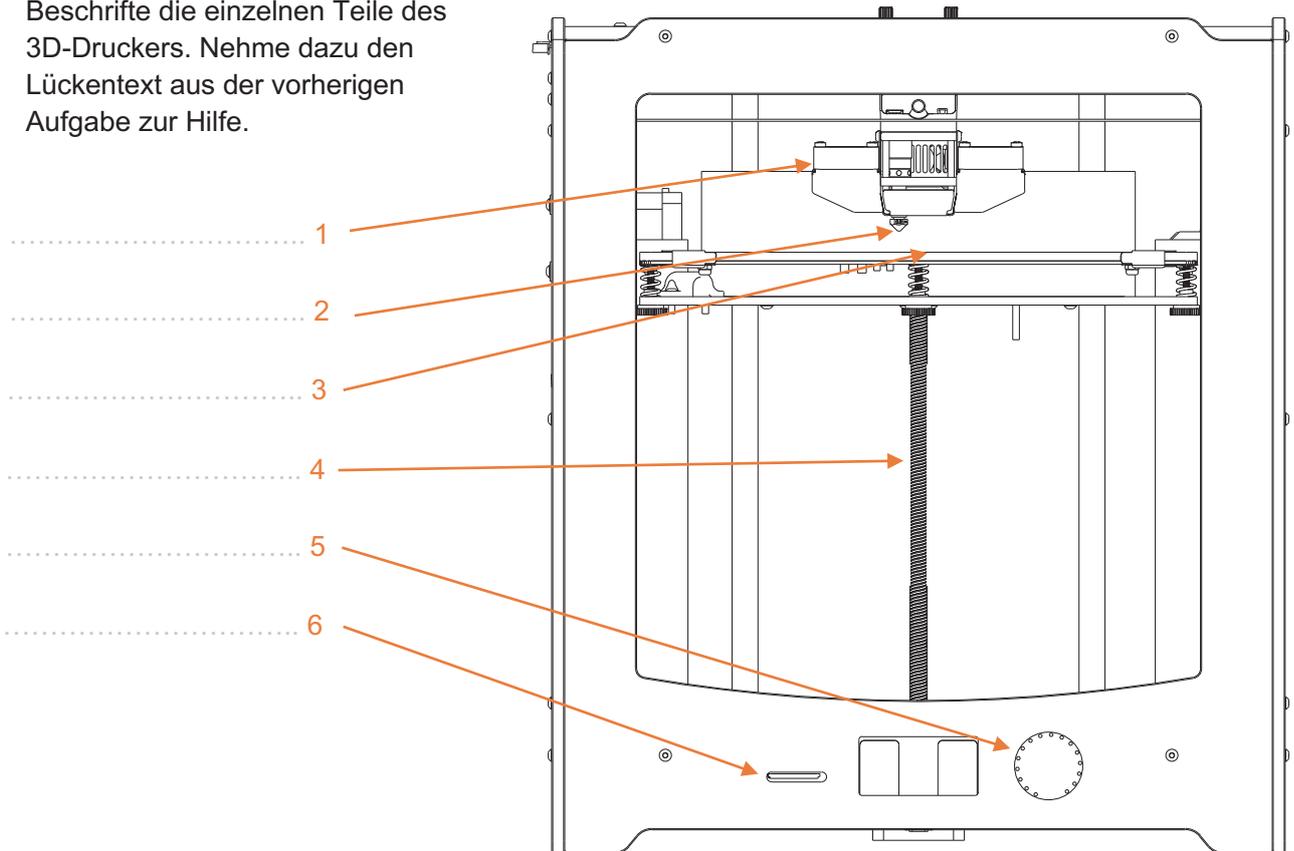
### Nachbearbeitung:

Im letzten Schritt wird der Druck vorsichtig vom Druckbett gelöst und ggf. Stützstrukturen vom Modell heruntergebrochen. Bei Bedarf kann das Modell noch nachgeschliffen werden.

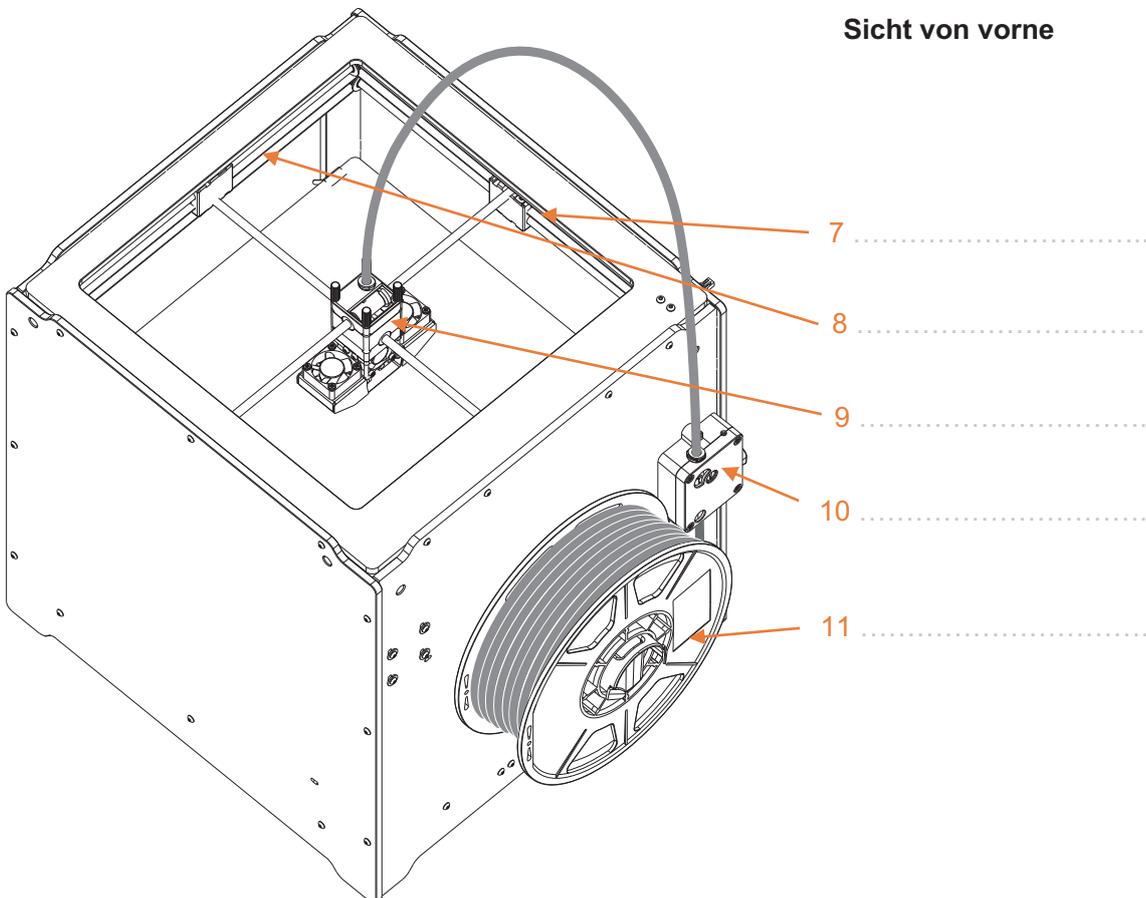
Temperaturfühler	Gewindestange	Fused Filament Fabrication	Bauteillüfter			
Filament	Rundstäben	Druckbett	Extruder	Hotend	Z	Düse
Medizintechnik	Speicherkarte	XY	Slicer	Prototypen	Kunststoffe	

## Aufgabe 3: Aufbau von 3D-Druckern

Beschrifte die einzelnen Teile des 3D-Druckers. Nehme dazu den Lückentext aus der vorherigen Aufgabe zur Hilfe.



Sicht von vorne



Sicht von oben/hinten

## Aufgabe 4: Ablauf eines 3D-Drucks

Das Slicing Programm „schneidet“ das 3D-Modell in einzelne Scheiben und erzeugt im Anschluss die notwendigen Steuerbefehle für den 3D-Drucker. Dies umfasst die Bewegung in XY-Richtung für den Druckkopf, aber auch in Z-Richtung für das Druckbett und den Extruder.

In der folgenden Aufgabe siehst du einen Ausschnitt aus einer G-code Datei. Der Befehl G1 bedeutet für die Steuerungssoftware des Druckers, dass eine lineare Bewegung zu einem Punkt ausgeführt werden soll.

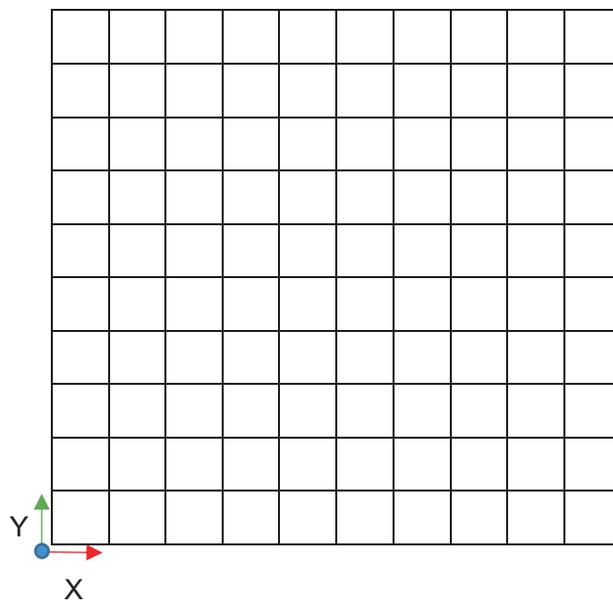
- a) Betrachte den G-code Ausschnitt aus Aufgabe b: Beschreibe wofür die Parameter X, Y und E stehen könnten?
- b) Zeichne die Bewegungen des Druckkopfes auf dem Druckbett anhand des G-code Ausschnittes nach. Welches 3D-Modell wird gedruckt?

Tipp: Die blaue Markierung zeigt den Nullpunkt an.

Ausschnitt aus der G-code Datei

```
G1 X50.000 Y28.000 E1.70284
G1 X70.000 Y18.500 E2.40565
G1 X67.500 Y41.000 E3.10840
G1 X83.000 Y57.000 E3.81116
G1 X61.000 Y61.500 E4.51386
G1 X50.000 Y81.000 E5.21666
G1 X40.000 Y61.000 E5.91946
G1 X17.000 Y57.000 E6.62218
G1 X32.500 Y41.000 E7.32494
G1 X30.000 Y18.500 E8.02581
```

Druckbett



- c) Erstelle deinen eigenen G-code und tausche diesen mit deinem Sitznachbarn aus.

---

---

---

---

---

---

---

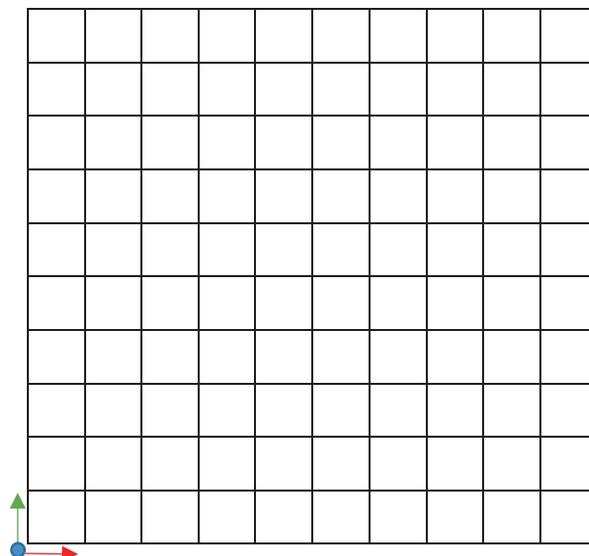
---

---

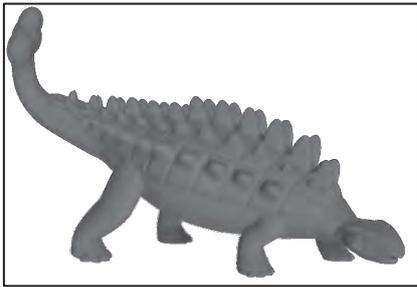
---

---

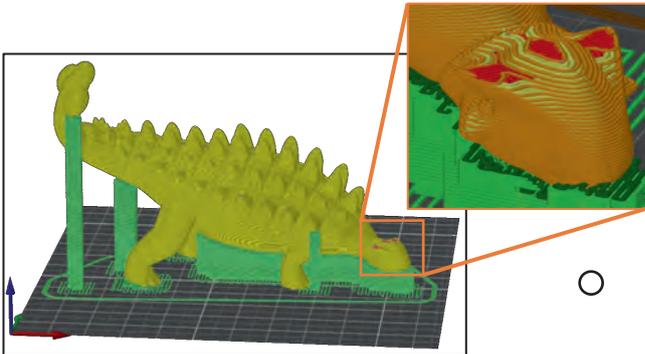
---



d) Ordne die folgenden Abbildungen den zugehörigen Begriffen zu.



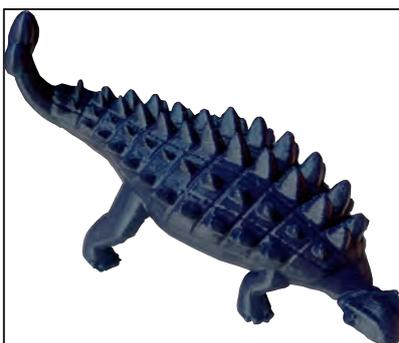
Physisches 3D-Modell aus PLA



3D-Druck



PLA Abfall



Modell nach dem Slicing



Digitales 3D-Modell

## Musterlösungen

### Arbeitsblatt B4.9: Aufbau und Funktionsweise von 3D-Druckern

#### Aufgabe 1:

<b>Name</b>	Eva	Julian	Nephele
<b>Beruf</b>	Medizintechnikerin	Maschinenbauingenieur	Architektin
<b>Aufgaben</b>	Forschung, Planung und Entwicklung neuer oder verbesserter Verfahren, Materialien, Geräte und Ausrüstungen im biomedizinischen Bereich	Planung, Konstruktion und Fertigung von Maschinen	Gebäudebauplanung Zeichnerische Darstellung von Entwürfen Objektüberwachung und -betreuung
<b>Interessiert, Folgendes zu drucken</b>	<p><u>Patientenorientierte Operationsmodelle:</u> 3D-gedruckte Anatomiemodelle aus Scandaten der Patientien als Anschauungsmodelle für ein besseres Verständnis</p> <p><u>Medizinprodukte und Instrumente:</u> Eva kann beispielsweise innerhalb weniger Tage komplexe Entwürfe testen, anstelle auf die Fertigung durch externe Anbieter zu warten.</p> <p><u>Prothesen</u></p> <p><u>Bioprinting:</u> Aktuell wird an der Fertigung von Organersatz geforscht. Dazu zählen Knochen, Herzklappen oder Blutgefäße aus dem 3D-Drucker</p>	<p><u>Ersatz und Sonderteilherstellung on-demand</u></p> <p><u>Schnelles Entwickeln und Testen von Prototypen für eine spätere Serienproduktion</u></p>	<p><u>Architekturmodelle zur Visualisierung</u></p> <p><u>Einstöckige Häuser aus Beton</u></p>

#### Vorteile:

- ∞ Individualisierung
- ∞ Hohe Designfreiheit
- ∞ Rapid Prototyping
- ∞ Umweltfreundlich durch Einsparung langer Transportwege
- ∞ Materialvielfalt
- ∞ Günstig bei kleinen Stückzahlen: gängige Spritzgußverfahren lohnen sich aufgrund der Kosten für die Werkzeuge erst ab einer hohen Stückzahl

Weiterführende Links:

<https://medizin-und-technik.industrie.de/medizintechnik-studium/faszination-medizintechnik/was-der-3d-druck-in-der-medizin-ausrichten-kann/>

<https://www.3dnatives.com/de/3d-druck-architektur-121120201/#>

## Aufgabe 2:

Es gibt viele verschiedene Arten von 3D-Druckern und Fertigungsverfahren. In einigen Verfahren wird eine harzähnliche Flüssigkeit mit UV-Licht (DLP-Verfahren) oder Lasern (SLA-Verfahren) gezielt ausgehärtet oder Pulver durch Laserstrahlen (SLS-Verfahren) geschmolzen. Andere Verfahren schmelzen **Kunststoffe** und pressen diese dann durch eine schmale Düse, um so aus dünnen Schichten Stück für Stück ein 3D-Modell herzustellen. Viele der Verfahren werden im Ingenieursbereichen, wie zum Beispiel **Medizintechnik** oder Maschinenbau, verwendet, um schnell **Prototypen** zu fertigen und zu testen. In den letzten Jahren wurde das sogenannte **Fused Filament Fabrication** (oder kurz FFF) vor allem im Hobby- und Bildungsbereich immer beliebter und die Geräte auch für Privatpersonen erschwinglich. Aber wie funktioniert so ein 3D-Drucker, der Kunststoffe schmelzen kann, eigentlich?

### Vorbereitung:

Um etwas drucken zu können, muss zunächst ein 3D-Modell erstellt (oder aus dem Internet heruntergeladen) und abgespeichert werden. Gängige Dateiformate sind STL oder 3MF. Diese Dateien werden mit einem speziellen Programm, dem sogenannten **Slicer**, geöffnet. Das Programm wandelt das 3D-Modell in Steuerbefehle für den 3D-Drucker um und fügt unter Umständen Stützstrukturen für Überhänge hinzu. Diese Befehle werden als G-code bezeichnet. Im Gegensatz zu Dateiformaten, wie STL und 3MF, wurden G-code Befehle schon seit den 1950er Jahren zur Steuerung von Fertigungsmaschinen genutzt. Mit Hilfe einer **Speicherkarte** oder einem USB-Stick kann die G-code-Datei auf den Drucker übertragen werden. Morderne 3D-Drucker können die Datei auch über das heimische Netzwerk direkt aus dem Slicer empfangen. Mit Hilfe eines Bedienfeldes kann die Datei ausgewählt und der Druck gestartet werden.

In der G-code Datei (rechts im Bild) steht unter anderem, welche mit welchen Temperaturen gedruckt werden soll und wohin sich der Druckkopf bewegt.

### Druck:

Statt Tinte verwenden FFF-Drucker **Filament** zum Drucken. Je nach Anforderung können verschiedene Kunststoffe verwendet werden. Besonders beliebt ist dabei das Material PLA (Polylactide), weil es sich bei vergleichsweise einfach und bei niedrigen Temperaturen (200°C bis 220°C) drucken lässt. PLA ist eine Polymilchsäure, die aus Maisstärke gewonnen werden kann und unter bestimmten Bedingungen auch biologisch abbaubar ist.

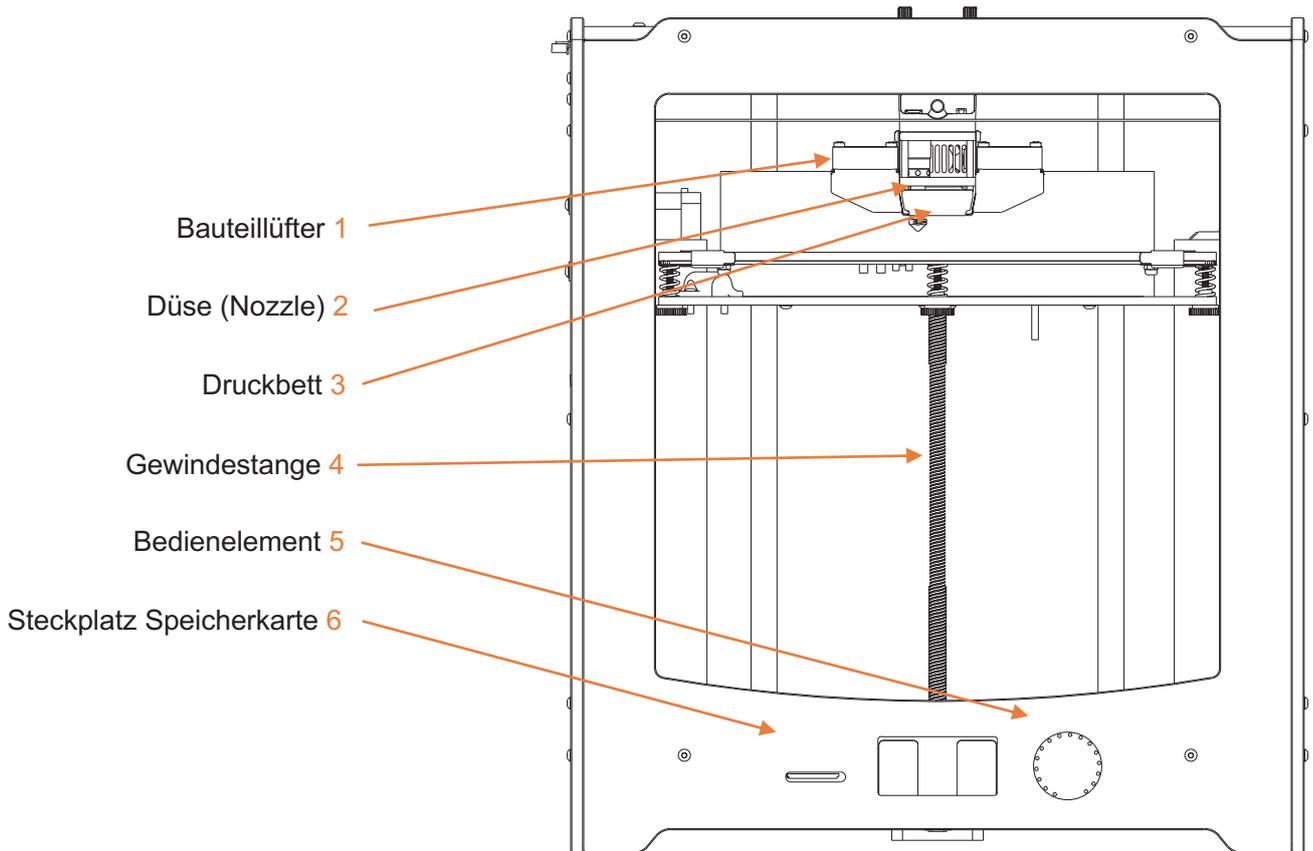
Das Filament aufgewickelt in Spulen geliefert. Der sogenannte **Extruder** befördert (oder extrudiert) das Material von der Spule zum Druckkopf. Der Druckkopf wird mit Hilfe von zwei Motoren, Riemen und Schlitten, die sich auf stabilen **Rundstäben** befinden in **XY**-Richtung bewegt. Im Druckkopf befindet sich ein zentrales Bauteil eines 3D-Druckers: das **Hotend**. Im Hotend das Material erhitzt und durch eine **Düse** (auch Nozzle genannt) gepresst. Am unteren Ende des Hotends befinden sich ein Heizelement zum Erhitzen der Düse und ein **Temperaturfühler** (Thermistor) zur Überwachung der Temperatur.

Das geschmolzene Material wird dann Schicht für Schicht auf dem **Druckbett** aufgetragen. Zum Aushärten wird das frisch aufgetragene Filament mit einem **Bauteillüfter** abgekühlt. Sobald eine Schicht fertig ist, bewegt ein weiterer Schrittmotor das Druckbett (ein typischer Wert sind 0,2mm) in **Z**-Richtung mit einer **Gewindestange** nach unten. Dieser Vorgang wird solange wiederholt, bis das Modell fertig ist. Je nach Modell, Düse, Schichthöhe und Druckgeschwindigkeit kann dies wenige Minuten, aber auch mehrere Tage dauern.

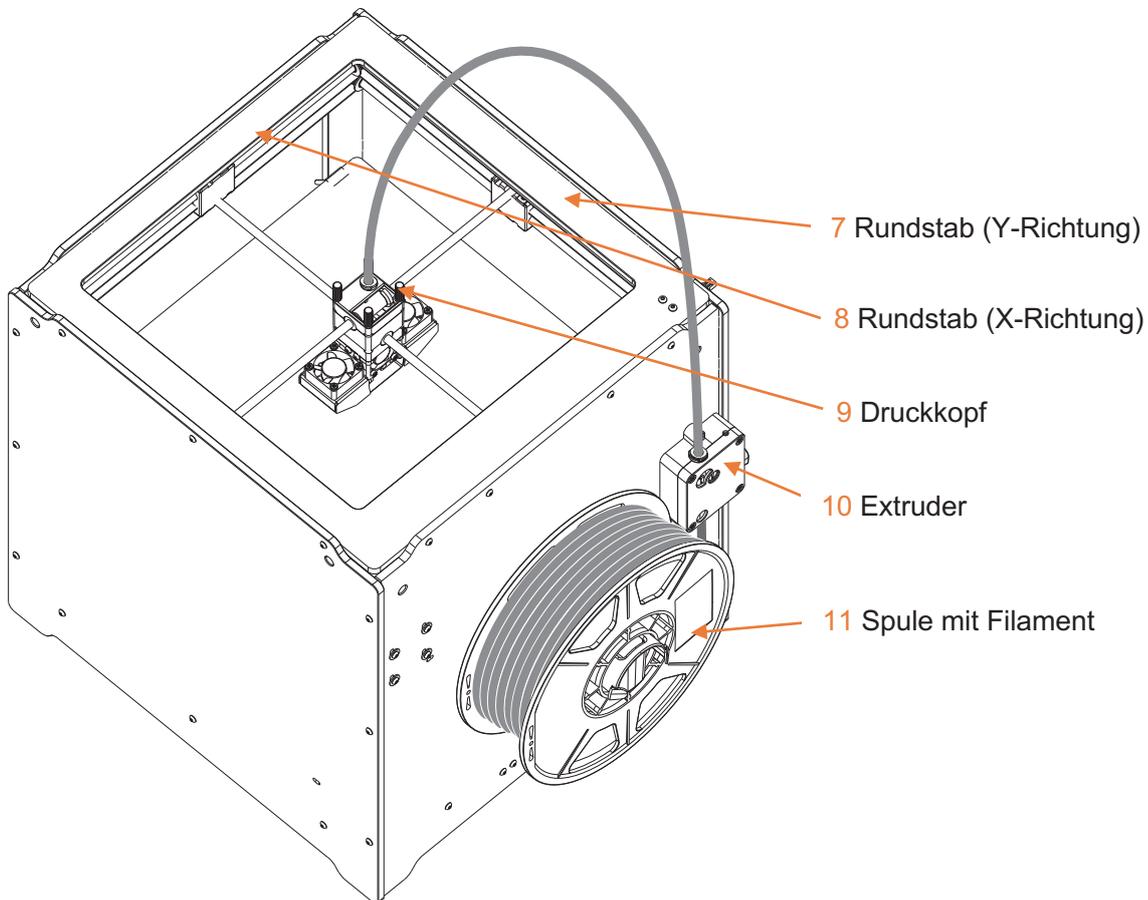
### Nachbearbeitung:

Im letzten Schritt wird der Druck vorsichtig vom Druckbett gelöst und ggf. Stützstrukturen vom Modell heruntergebrochen. Bei Bedarf kann das Modell noch nachgeschliffen werden.

## Aufgabe 3:



Sicht von vorne

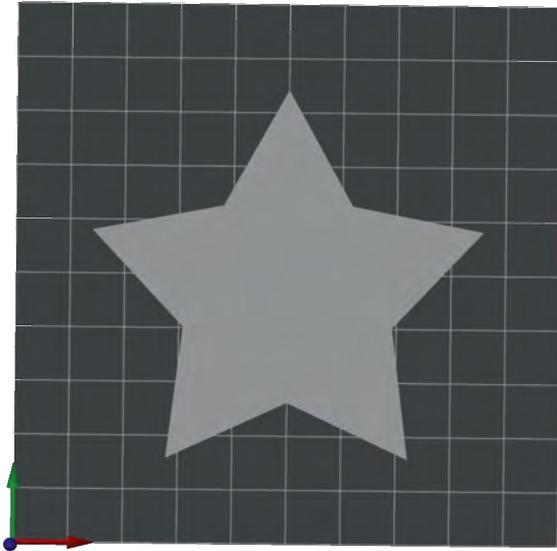


Sicht von oben/hinten

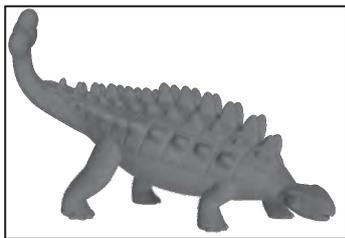
## Aufgabe 4:

a.) Die Parameter X und Y stehen für die Position der Nozzle. Der Parameter E steht für die zurückgelegte Stecker des Extruders.

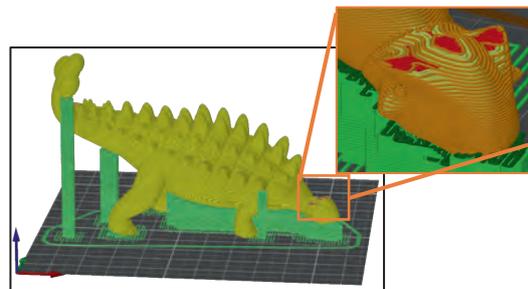
b.)



d.)



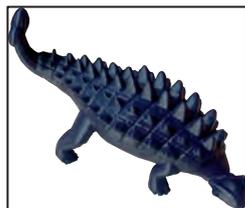
Digitales 3D-Modell



Modell nach dem Slicing



3D-Druck



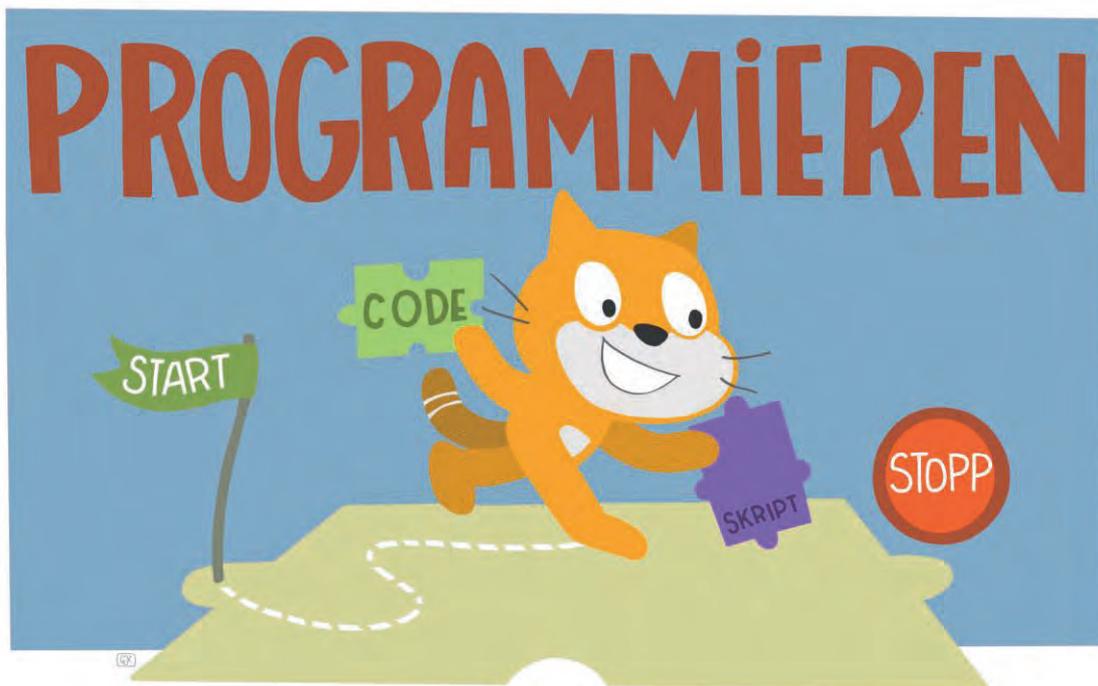
Physisches 3D-Modell aus PLA



PLA Abfall

# IT2School

Gemeinsam IT entdecken



## Modul B5 – Programmieren

Leichter ProgrammierEinstieg

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Inhalt

1	Leichter ProgrammierEinstieg.....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	5
4	Rolle der Unternehmensvertreter*innen.....	5
5	Inhalte des Moduls.....	5
5.1	Die Entwicklungsumgebung von Scratch .....	5
5.2	Programmieren mit Scratch .....	7
6	Unterrichtliche Umsetzung.....	9
6.1	Grober Unterrichtsplan.....	9
6.2	Stundenverlaufsskizzen .....	10
6.2.1	Einführung in Scratch .....	10
6.2.2	Umsetzung eines eigenen Projekts.....	11
7	Einbettung in verschiedene Fächer und Themen .....	13
8	Anschlussthemen.....	14
9	Literatur und Links .....	15
10	Arbeitsmaterialien .....	15
11	Glossar.....	16
12	FAQs und Feedback.....	18

# 1 Leichter ProgrammierEinstieg

In diesem Modul sorgt eine einfache Entwicklungsumgebung dafür, dass die Schülerinnen und Schüler erste Erfahrungen in der Programmierung machen können. Als Hilfsmittel hierfür dient die grafische blockbasierte Programmiersprache Scratch.

Eine grafische blockbasierte Programmierung gestaltet den Einstieg für die Schülerinnen und Schüler einfacher als eine textuelle Programmiersprache und erlaubt es trotzdem, selbst komplizierte Anwendungen zu erstellen. Aufgrund der Interaktionsmöglichkeiten verschiedener Objekte (z.B. Unterhaltung zwischen verschiedenen Figuren mittels Sprechblasen) können auch schon sehr junge Schülerinnen und Schüler einen Einstieg in die Programmierung finden. Das schnelle Erfolgserlebnis und das eigenständige und intuitive Zusammensetzen der Programme stärken zudem das Selbstvertrauen.



<b>Lernfeld/Cluster:</b>	IT spielend entdecken	
<b>Zielgruppe/Klassenstufe:</b>	<b>X</b>	4. bis 5. Klasse
	<b>X</b>	6. bis 7. Klasse
	<b>X</b>	8. bis 10. Klasse
	<b>X</b>	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	6 bis 7 Einzelstunden (zusätzliche Unterrichtsstunden zur Vertiefung werden empfohlen!)	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Entwickeln eines grundlegenden Programmierverständnisses</li> <li>• Eigenschaften von Programmen beschreiben</li> <li>• Programme selbst gestalten</li> <li>• Algorithmische Grundbausteine zur Programmierung verwenden</li> <li>• Entwerfen, Implementieren und Testen eigener Programme</li> </ul>	
<b>Vorkenntnisse der Schüler*innen:</b>	<p>Erforderlich:</p> <ul style="list-style-type: none"> <li>• Programme aufrufen und bedienen</li> <li>• Eingaben mit Maus und Tastatur</li> <li>• Laden und Speichern von Dateien</li> </ul> <p>Empfohlen:</p> <ul style="list-style-type: none"> <li>• Internetkenntnisse (zum Hochladen der Programme auf die Scratch-Website)</li> <li>• Grundlegende Erfahrungen mit der Bildbearbeitung (hilfreich bei der Erstellung eigener Figuren und Hintergründe)</li> </ul>	
<b>Vorkenntnisse der Lehrkraft:</b>	<p>Erforderlich:</p> <ul style="list-style-type: none"> <li>• Grundlegende Erfahrungen im Programmieren (durch einmaligen Durchlauf dieses Moduls zu erlangen)</li> </ul>	



<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Grundlegende Erfahrungen im Programmieren, Ausprobieren des Scratch-Tutorials</li> </ul>
<b>Sonstige Voraussetzungen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Mindestens 1 Computer pro 2 Schülerinnen und Schüler</li> <li>• Scratch 2 ist in der Regel auf jedem Rechner lauffähig, trotzdem sollte dies vor Unterrichtsbeginn getestet werden</li> <li>• Seit dem Update auf Scratch 3 ist Scratch durch den Einsatz von HTML 5 auch im Browser von Tablets einsetzbar, da seit Scratch 3 kein Adobe Flashplayer mehr benötigt wird</li> </ul> Empfohlen: <ul style="list-style-type: none"> <li>• Beamer zur Präsentation sowie zum gemeinsamen Zeigen und Arbeiten mit Scratch</li> </ul>

## 2 Warum gibt es das Modul?

In diesem Modul wird den Schülerinnen und Schülern Raum für spannende eigenständige Projekte eröffnet. Der Fokus liegt dabei auf der kreativen Gestaltung von Programmen, die mit Hilfe der visualisierten Entwicklungsumgebung Scratch umgesetzt werden.

Durch das Erlernen von Programmiersprachen verstehen Kinder und Jugendliche nicht nur, wie die Geräte, die sie verwenden, funktionieren. Vielmehr erhalten sie die Möglichkeit, durch logische Abfolgen aktiv einzugreifen und Einfluss darauf zu nehmen. Dies könnte eine der wesentlichen Kompetenzen der Zukunft sein. So fasst auch der New Yorker Medienwissenschaftler Douglas Roushko<sup>1</sup> es mit wenigen Worten zusammen: „Program or be programmed“ – was auf Deutsch so viel heißt wie: „Programmiere selbst oder du wirst programmiert“.

Ziel des Moduls ist es daher, das Interesse an Informationstechnik bei Kindern und Jugendlichen zu fördern und daraus folgend auch ihre gesellschaftliche Mündigkeit zu unterstützen. Sie sollen die Grundlagen der Technik verstehen, einfache Softwarekomponenten selbst programmieren und eigene innovative Lösungen entwickeln. Dadurch werden auch das logische Denken sowie Problemlösekompetenz gefördert.

Während der aktiven Auseinandersetzung und Entwicklung eigener Projekte erfahren die Kinder und Jugendlichen etwas über moderne Arbeitsprozesse und bekommen die Möglichkeit, die digitale Gesellschaft mitzugestalten. Zu Beginn dieses Moduls steht die Heranführung an die Programmierung im Vordergrund. Es wird dabei auch aufgezeigt, wie Aspekte des Problemlösens und der Algorithmisierung, also einer eindeutigen Handlungsvorschrift zur Lösung von Problemen, vertieft werden können.

---

<sup>1</sup> <http://www.rushkoff.com/program-or-be-programmed/>

### 3 Ziele des Moduls

- Kennenlernen der Grundkonzepte der Computerprogrammierung
- erleben, wie (leicht) kleine Programme gestaltet werden können
- Softwareerstellung als kreativen Prozess erkennen und erleben
- Beschreiben der Eigenschaften von Programmen
- Verwendung von algorithmischen Grundbausteinen zur Programmierung

### 4 Rolle der Unternehmensvertreter\*innen

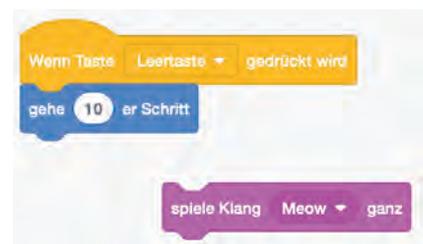
Im *Modul B5 – Programmieren* hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten aktiv mitzuwirken. Hier einige Anregungen:

- Unterstützung der Lehrkraft, z. B. bei der Einführung in die Scratch-Programmier-Oberfläche
- Unterstützung der Schülerinnen und Schüler bei Umsetzung ihrer eigenen kreativen Projekte
- Sie oder er kann zur Abschlusspräsentation der Projektergebnisse als Special-Guest eingeladen werden.

### 5 Inhalte des Moduls

Scratch ist eine visuelle blockbasierte Programmier-sprache, die von der Lifelong Kindergarten Group des MediaLab am Massachusetts Institute of Technology (MIT) entwickelt wurde, um insbesondere Kinder und Jugendliche mit der Computerprogrammierung vertraut zu machen.

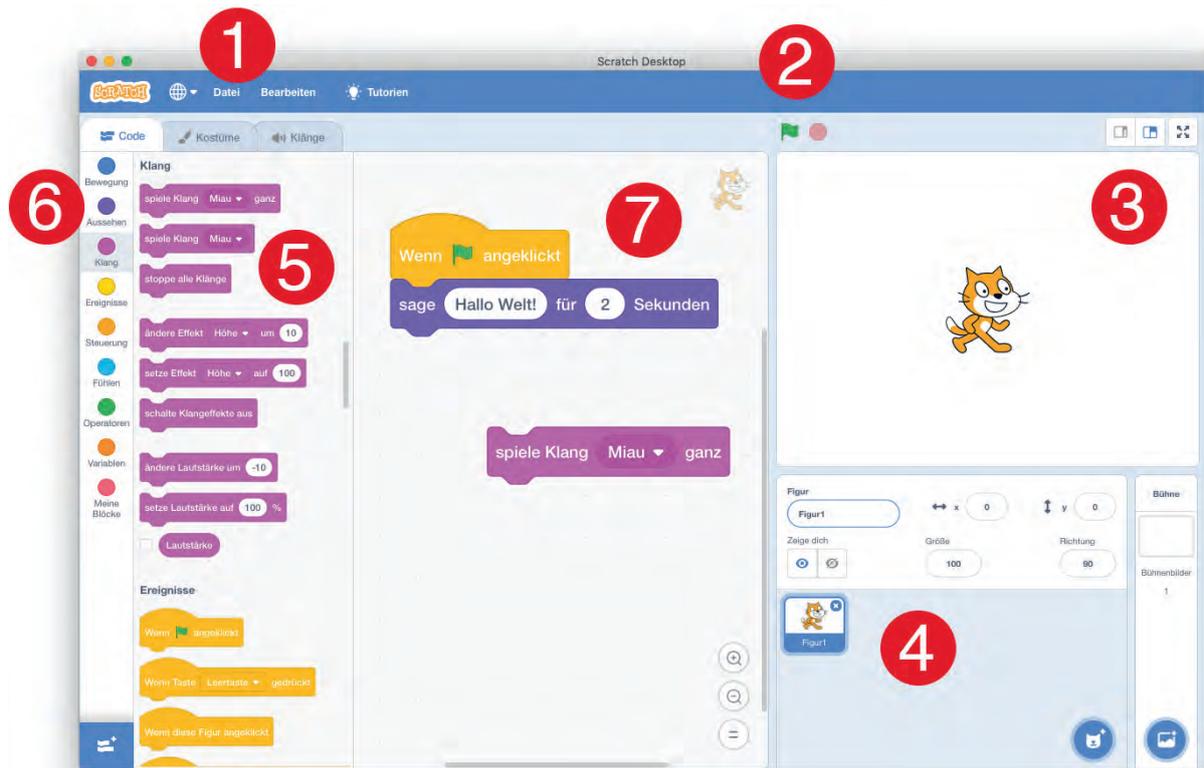
In dieser Entwicklungsumgebung werden Befehle in Form von Blöcken beziehungsweise Bausteinen dargestellt und können ähnlich wie Puzzleteile „aneinandergesteckt“ werden. Auf diese Weise müssen Schülerinnen und Schüler keine Programmierbefehle auswendig lernen oder Kommata und sonstige Sonderzeichen beachten, wie es sonst bei textuellen Programmiersprachen (z. B. C+, Java, Visual Basic) nötig ist. Die Programmierung wird auf das Wesentliche reduziert, wodurch auch Anfänger sehr schnell interessante Ergebnisse erzielen. Diese schnellen Erfolgserlebnisse fördern die Motivation sowie das Selbstvertrauen der Schülerinnen und Schüler. Durch die freie Verfügbarkeit von Scratch ist es außerdem möglich, Freunden und Verwandten die selbstgestellten Programme zu Hause vorzuführen und das Neuerlernte so zu teilen.



#### 5.1 Die Entwicklungsumgebung von Scratch

Für die Entwicklung von Programmen mit Scratch kann ein Offline- oder ein Online-Editor genutzt werden. Das bedeutet, Sie können ohne aufwendiges Installieren von Programmen online direkt im Browser (Online-Editor) oder unabhängig von der Internetverbindung offline mit einem Programm (Offline-Editor *Scratch Desktop*) programmieren. Für die Schule ist der Off-

line-Editor zu empfehlen, um nicht von der Internetverbindung der Schule abhängig zu sein und das Netz weniger zu belasten.



Im Folgenden werden die nummerierten Bereiche der Arbeitsumgebung beziehungsweise des Editors genauer erklärt:

1. Die verschiedenen Reiter, wie zum Beispiel **Datei** oder **Bearbeiten**, stellen das Menü des Programms dar. Diese Menüs bieten unter anderem Optionen zum Speichern und Laden von Programmen.
2. Mit der *grünen Fahne* werden üblicherweise Programme in Scratch gestartet und mit dem *roten Sechseck* gestoppt. So lässt sich ansehen, was man bisher programmiert hat.
3. Auf der *Bühne* (Stage) werden die *Figuren* (auch Objekte oder Sprites genannt) positioniert und führen die Befehle aus, die man programmiert hat. Alle sichtbaren Befehle der Scratch-Programme werden hier dargestellt.
4. In diesem Bereich werden alle Figuren, die im Programm enthalten sind, aufgelistet. Nicht alle Figuren müssen immer auf der Bühne sichtbar sein, sie können sich auch verstecken oder erst zu einem späteren Zeitpunkt erscheinen. Links daneben sind alle Hintergründe für die Bühne aufgelistet.
5. Anweisungen beziehungsweise *Befehle*, die der Computer für die Ausführung von Programmen benötigt, sind hier in Form von *Bausteinen* dargestellt. Der Computer arbeitet diese Schritt für Schritt ab.
6. Die Programmieranweisungen sind in Kategorien zusammengefasst. Unter **Bewegung** sind alle Programmierbefehle zu finden, um einzelne Figuren zu bewegen, unter **Aussehen** findet man Kostümwechsel für die Figuren, aber auch Sätze, die gesagt werden können.

7. Der große weiße Bereich in der Mitte des Fensters wird als das Skriptfenster bezeichnet. Um ein Programm zu schreiben, werden die einzelnen Bausteine mit gedrückter Maustaste in diesen Bereich gezogen. Wenn also der Baustein „**wenn Leertaste gedrückt – gehe 10er-Schritt**“ hier positioniert wird, geht die Katze bei entsprechendem Tastendruck vorwärts. Mehrere solcher zusammengesetzter Bausteine nennt man *Skript*, wohin gegen die Gesamtheit aller Figuren, Bühnen und Skripte als *Programm* bezeichnet wird.

## 5.2 Programmieren mit Scratch

Wie bereits im Abschnitt 3.1.1 erwähnt, werden Programme in Scratch in der Regel durch das Klicken der grünen Fahne gestartet. Bei „Wenn [grüne Fahne] angeklickt“ handelt es sich um ein Ereignis. So findet man auch diesen Ereignisbaustein in der Kategorie „Ereignisse“.



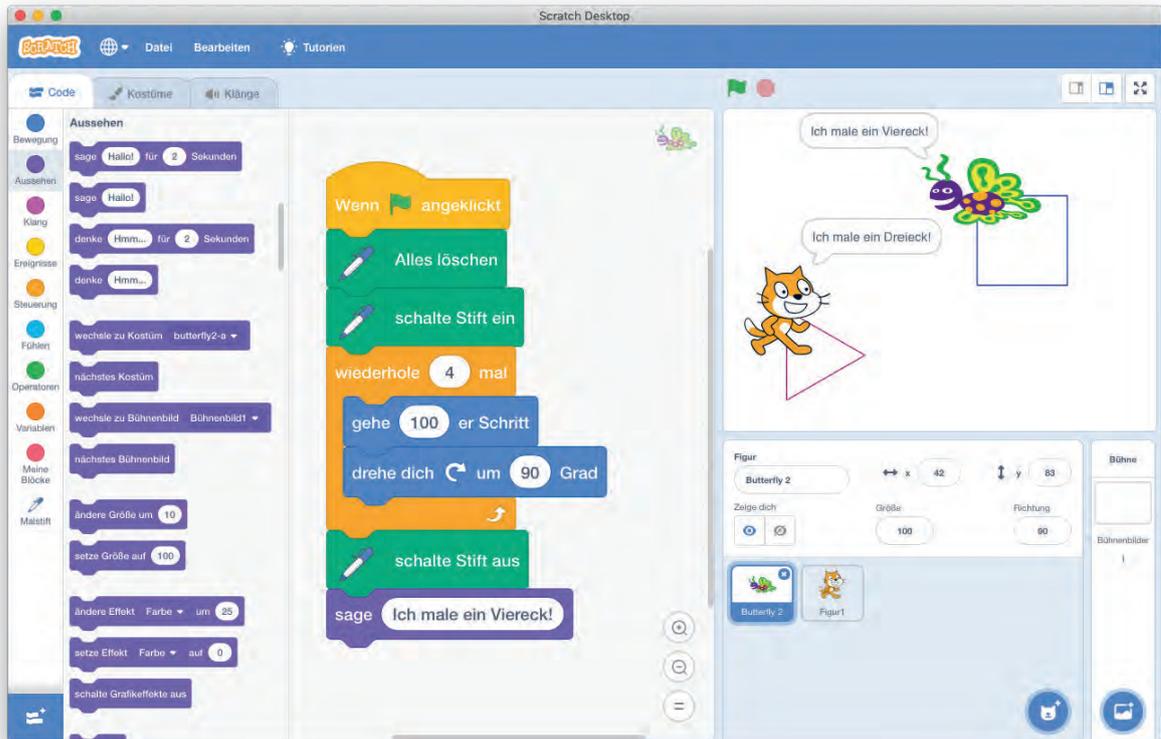
An das Ereignis können weitere Bausteine angeordnet werden. Dazu werden sie einfach an die gewünschte Position gezogen. Ein so geschaffenes Skript wird immer von oben nach unten Schritt für Schritt abgearbeitet. Ein Skript kann wie folgt aussehen:



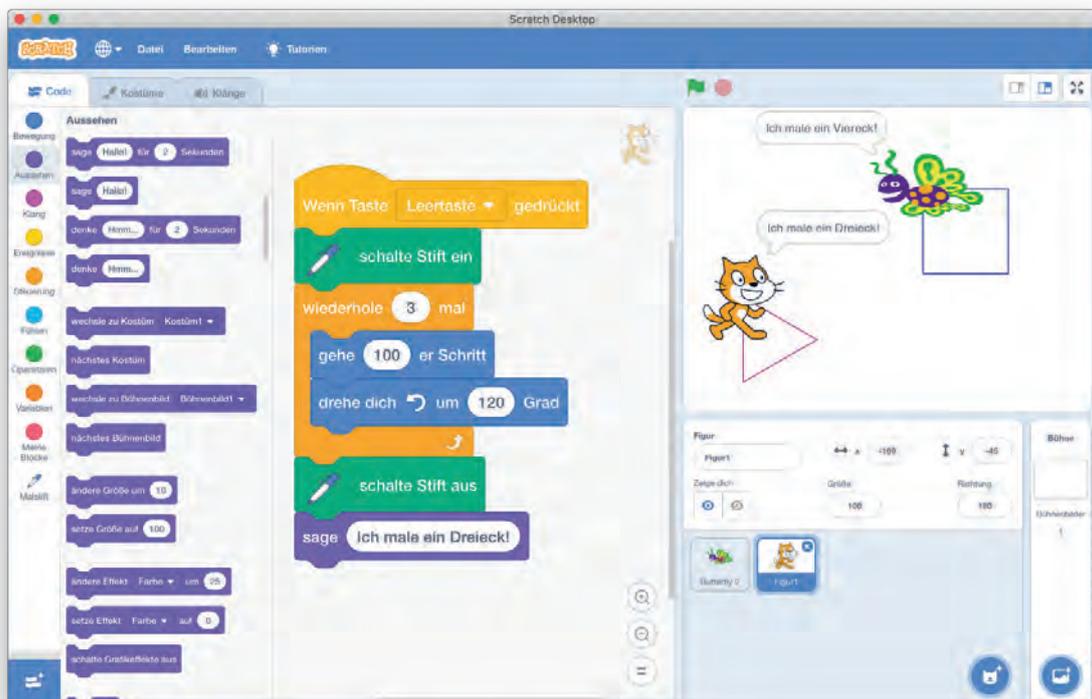
Dieses Beispiel würde dafür sorgen, dass die Figur zu Beginn nach rechts schaut und dann einmal ein Sechseck („6 mal“ „drehe dich um 60 Grad“) mit einer Schrittweite von 100 („6 mal“ „gehe 100er-Schritt“) abläuft.

So würde sich aber nur die Figur verhalten, der dieses Skript zugeordnet wurde. Fügt man in den Bereich 4 (siehe Abschnitt 3.1.1) nun eine weitere Figur hinzu und klickt diese an, dann leert sich Bereich 7 und man sieht nur die Skripte dieser Figur. Dies kann wie folgt aussehen<sup>2</sup>:

<sup>2</sup> Hinweis: Der Malstift ist seit Scratch 3 unter den Erweiterungen zu finden.



Dasselbe Scratch-Projekt sieht bei Auswahl der Katze wie folgt aus:



## 6 Unterrichtliche Umsetzung

Auch auf der Webseite von Scratch finden sich zahlreiche Animationen, Spiele und kleine interaktive Geschichten, die einen großen Aufforderungs- und Nachahmungscharakter besitzen. Sie sollen genutzt werden, um bei den Schülerinnen und Schülern Neugierde und die Lust zu wecken, selbst mithilfe von Scratch solche Filme und Spiele zu programmieren. Danach werden die Schülerinnen und Schüler schrittweise mit der Entwicklungsumgebung von Scratch und den einzelnen Bausteinen vertraut gemacht. Dabei sollte darauf geachtet werden, dass sie schnell erste Erfolge durch das Erstellen eigener Programme erreichen. Hierfür wurde ein Tutorial entwickelt, das die Schülerinnen und Schüler schrittweise durchgehen können, um kleine Programmiersequenzen zu erstellen. Zum Abschluss können sie das Gelernte in ein größeres Projekt einfließen lassen und selbst kreativ werden.

Als Ergänzung zu Scratch bietet sich das Modul B6 Mein Anschluss an. Hier bietet der MocoMoco als Eingabegerät die Möglichkeit noch kreativere und interaktivere Programme und Projekte zu entwickeln und zu gestalten.

### 6.1 Grober Unterrichtsplan

Unterrichtsszenarien	Kurze Zusammenfassung
Einführung und erste Schritte mit Scratch	Die Schülerinnen und Schüler werden für Scratch motiviert und die Entwicklungsumgebung wird ihnen anhand eines Tutorials und mit Unterstützung der Lehrkraft nähergebracht.
Einarbeitung und Vertiefung in Scratch	Die Schülerinnen und Schüler entwickeln eine eigene Idee und setzen diese um.
Präsentation	Die programmierten Geschichten und Computerspiele werden präsentiert.

## 6.2 Stundenverlaufsskizzen

### Abkürzungen/Legende

AB = Arbeitsblatt/Arbeitsblätter; L = Lehrkraft; MuM = Mitschüler\*innen; SuS = Schüler\*innen; UV = Unternehmensvertreter\*in

#### 6.2.1 Einführung in Scratch

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
5 Min.	Vorbereitung		<p>Es sollte sichergestellt sein, dass Scratch entweder als Offline- oder Online-Editor auf den Computern der SuS funktionstüchtig ist. Die Editoren können unter folgenden URLs aufgerufen werden:</p> <ul style="list-style-type: none"> <li>• <a href="https://scratch.mit.edu/scratch2download/">https://scratch.mit.edu/scratch2download/</a></li> <li>• <a href="https://scratch.mit.edu/projects/editor/">https://scratch.mit.edu/projects/editor/</a></li> </ul>	
5 Min.	Einstieg	Lehrkraftpräsentation	<p>Begrüßung SuS; ggf. Vorstellung UV; als Einstieg werden ein bis zwei Programme mit Scratch vorgeführt, um Neugierde zu wecken und die Lust, es selbst auszuprobieren. Beispiele sind auf der Scratch-Internetseite zu finden: <a href="https://scratch.mit.edu/starter_projects/">https://scratch.mit.edu/starter_projects/</a></p> <p>Beispielprojekte:</p> <ul style="list-style-type: none"> <li>• <a href="https://scratch.mit.edu/projects/1192/">https://scratch.mit.edu/projects/1192/</a></li> <li>• <a href="https://scratch.mit.edu/projects/13701368/">https://scratch.mit.edu/projects/13701368/</a></li> </ul>	Computer, Beamer, Internetverbindung
15 Min.	Hinführung	Lehrkraftvortrag	<p>Nach dem Einstieg führt L kurz in die Entwicklungsumgebung ein. Folgende Fragestellungen sollten dabei beantwortet werden:</p> <ul style="list-style-type: none"> <li>• Was ist Scratch?</li> <li>• Wie kann/soll das Programm aufgerufen werden?</li> <li>• Wie werden Programme, die mit Scratch geschrieben wurden, aufgerufen?</li> <li>• Wie ist die Entwicklungsumgebung in groben Zügen aufgebaut?</li> </ul>	

70 Min.	Erarbeitung	Einzel- oder Gruppenarbeit	SuS erhalten B5.1, arbeiten es durch und lösen die darin enthaltenen Aufgaben.	B5.1 (Einstiegtutorial)
20 Min.	Erarbeitung	Tandemarbeit	Im Anschluss vertiefen SuS ihre Kenntnisse mithilfe der Scratch-Würfel. (Wenn diese noch nicht fertig zusammengebaut sind, muss ggf. mehr Zeit eingeplant werden!) Jedes Team erhält ein Set Scratch-Würfel (B5.2 ist die Bauvorlage) sowie B5.3 zur Bearbeitung. Es wird dann geschaut, was sich aus den durch den Würfel vorgegebenen Bausteinen programmieren lässt. Dadurch entstehen verschiedene Zufallsergebnisse.	B5.2 (Bauvorlage), B5.3 (Arbeitsauftrag)
10 Min.	Sicherung	Präsentation	Die Gruppen präsentieren und erklären die entstandenen Skripte. Gegebenfalls können die Ergebnisse in einem Wiki oder auf einer Lernplattform festgehalten werden.	Computer, Beamer

Zur Einführung finden Sie weitere Aufgabenstellungen und Arbeitsblätter unter „Zusatzmaterial“ (B5.5 – B5.8). Diese Arbeitsblätter behandeln Schritt für Schritt den Aufbau von Scratch sowie den Funktionsumfang der Entwicklungsumgebung. Dies ist vor allem für Schülerinnen und Schülern interessant, die Schwierigkeiten mit der offenen Projektarbeit haben oder mehr Hilfestellungen benötigen. Im weiteren Verlauf dieses Unterrichtsszenarios wird ein eigenes Programm der Schülerinnen und Schüler umgesetzt. Es ist aber genauso denkbar B3.5 – B3.8 zu bearbeiten.

### 6.2.2 Umsetzung eines eigenen Projekts

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Plenum	Begrüßung, L klärt über den Stundenverlauf auf, SuS setzen ein eigenes Programm in Teams oder Kleingruppen (unter Verwendung von B5.4) um. SuS können wählen zwischen: <ul style="list-style-type: none"> <li>• Computerspiel</li> <li>• animierte Geschichte</li> </ul> L fungiert als Lernbegleiter*in und unterstützt die Gruppen nach Bedarf. Zu	B5.4 (Projektaufgabe)

30–45 Min.	Projektplanung	Gruppenarbeit	<p>Beginn können Internetlinks, beispielsweise zum Scratch-Wiki, an die SuS werden, damit sie auftretende Probleme eigenständig lösen können.<sup>3</sup></p> <p>Aufteilung in Gruppen, die Gruppen entscheiden sich für ein Thema und gehen es wie folgt an:</p> <ul style="list-style-type: none"> <li>• Entwicklung einer Idee</li> <li>• Storyboard für Geschichte oder Konzept und Design für das Spiel entwickeln (alles wird mit Stift und Papier festgehalten)</li> <li>• Aufgaben in der Gruppe werden bei Bedarf verteilt</li> </ul>	Stifte, Papier zur Ideenfindung
60–120 Min.	Durchführungsphase	Gruppenarbeit	<p>SuS setzen ihre Ideen mit Scratch um:</p> <ul style="list-style-type: none"> <li>• Programmierung</li> <li>• Ausprobieren (Testing)</li> <li>• Fertigstellen</li> </ul>	Computer mit Internetverbindung
10 Min.	Präsentation	Plenum	SuS präsentieren ihre Animationen und Spiele und laden sie gegebenenfalls auch auf die Scratch-Webseite hoch.	Computer, Beamer
15 Min	Reflexion	Plenum	Auseinandersetzung mit dem Prozess: Was hat geklappt? Was hat nicht geklappt? Wie war das Arbeiten im Team? Gab es Probleme? Sollte man öfter Projektarbeit einsetzen?	

**Hinweis:** Vergessen Sie nicht, auch die Möglichkeiten des MocoMoco aus dem Modul B6 in den Folgeunterricht einfließen zu lassen.

<sup>3</sup> Weitere Informationen zu der Projektmethode finden Sie im Methodenmodul M2.

## 7 Einbettung in verschiedene Fächer und Themen

Die Erstellung von Filmen und Geschichten sowie deren Planung mithilfe eines Storyboards stellt Bezüge zum Fach Deutsch her. Im Einzelnen hervorzuheben sind hier die prozessbezogene Kompetenz „Schreiben“ sowie die inhaltsbezogene Kompetenz „Texte und Medien“ mit dem Schwerpunkt Medien (Kompetenzen nach KMK).

Die grafische Gestaltung ebenso wie die Planung eines möglichen Films lassen sich auch mit dem Unterrichtsfach Kunst verbinden.

Durch die Bezüge zu unterschiedlichen Fächern kann das gesamte Modul oder können einzelne Teile in verschiedenen Fächern eingesetzt werden. Die folgenden Kompetenzen finden sich entweder in den Bildungsstandards der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

### Deutsch

Die Schülerinnen und Schüler ...

- können handelnd mit Texten umgehen, mit Sprache experimentell und spielerisch umgehen z. B. illustrieren, inszenieren, umgestalten, ... (Grundschule).
- können verschiedene Medien für Präsentationen nutzen (Grundschule).
- Texte (medial unterschiedlich vermittelt) szenisch gestalten (Sek I).
- Texte mit Hilfe von neuen Medien verfassen (Sek I).

### Kunst

Die Schülerinnen und Schüler ...

- setzen digitale Medien für eine Gestaltungsaufgabe ein.
- setzen designorientierte Findungsprozesse und Lösungsstrategien ein.
- entwickeln verschiedene Ideen in einem Entwurfsprozess und stellen den Entwurf angemessen dar.
- realisieren filmische Projekte (Stopmotion, Video, Computeranimationen).

### Informatik

Die Schülerinnen und Schüler ...

- erstellen Produkte unter Anwendung fortgeschrittener Techniken von Standardsoftware; falls es das Produkt erfordert, arbeiten sie sich in geringem Umfang in Spezialsoftware ein.
- erwerben beim Bearbeiten von Softwareprojekten in angemessenem Umfang Kenntnisse über Analyse- und Modellierungsverfahren sowie Projektmanagement.
- verstehen Programmabläufe und die Arbeitsweise von Schnittstellen.
- können erdachte Systeme in technische Systeme übertragen.

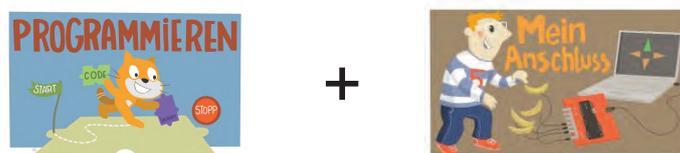
- kennen sich in Entwicklungsumgebungen/Programmierungsumgebungen aus.
- eine Spielumgebung entsprechend der Spielidee gestalten.
- eine Spielvariante mit informatischen Werkzeugen entwerfen und implementieren.

## 8 Anschluss Themen

Als Anschluss Themen im Zusammenhang mit IT2School bieten sich folgende Bausteine an:

### Beispiel: Mein Anschluss

Wenn Sie das Gesamtpaket von IT2School in Form einer Bildungspartnerschaft erhalten haben, so besitzen Sie auch einen Klassensatz von MocoMocos. Dies bietet die Möglichkeit die Programmierung durch weitere Projekte mit Einbindung des MocoMocos auszubauen.



### Beispiel: Programmieren

Eine zu Scratch ähnliche Entwicklungsumgebung stellt auch der App Inventor dar. Dieser erlaubt die Gestaltung und Programmierung eigener Apps für Android Smartphones. Im Anschluss an die visuelle Programmierung kann die textuelle Programmierung zur Vertiefung herangezogen werden.



Tipp: Darüber hinaus bietet Scratch noch viele weitere Möglichkeiten zur Erweiterung und Vertiefung:

- **Das Pico-Board**  
An die Computer kann ein sogenanntes Pico-Board (Mikrocontroller-Board) angeschlossen werden. Das Pico-Board ist speziell für Scratch entwickelt worden und integriert einen Drucktaster, einen Lichtsensor, einen akustischen Sensor sowie vier weitere Anschlüsse für Sensoren (z. B. Bewegungssensor). Durch das Pico-Board wird Scratch interaktiv und eröffnet viele Projektideen für den Unterricht.
- **Angebote von Makeblock**  
Die Firma Makeblock bietet ein vielfältiges Baukastensystem aus elektronischen und mechanischen Teilen an, aus denen Roboter, Maschinen und andere Konstruktionen gebaut werden können. Alles lässt sich mit Scratch programmieren. Insbesondere das „Makeblock Inventor Electronic Kit“ ist interessant. Es beinhaltet zwölf Sensoren und Aktoren und bietet alles für schnelles Prototyping.

- **Squeak und BYOB**

Auf Scratch aufbauend kann es sich anbieten, mit den Open-Source-Entwicklungsumgebungen Squeak und Snap/BYOB (Build Your Own Blocks) weiterzuarbeiten. Squeak unterstützt den Textmodus und weitere Konzepte, die nicht in Scratch vorhanden sind. In BYOB können sogar eigene Bausteine erstellt werden und Netzwerkkommunikation ist möglich.

## 9 Literatur und Links

- Glöde, Martina & Reit, Birgit (Hg.) (2015): **Programmieren supereasy. Einfacher Einstieg in Scratch und Python**. München: Dorling Kindersley.
- Bartmann, Erik (2015): **Faszinierende Elektronik-Projekte mit Scratch, Arduino und Raspberry Pi**. 1. Aufl. Beijing: O'Reilly (O'Reillys basics).
- Engeln, Frank (2010): **Scratch me if you can – Programmieren mit Scratch, Lego® WeDo und Picoboard**. Online: <https://collection.switch.ch/objects/LOR:6384/datastreams/DS5>
- **Offizielle Scratch-Website**, Beispiele, Tipps – online: <https://scratch.mit.edu>
- Deutsches **Scratch-Wiki** – online: <http://www.scratch-dach.info/>
- Konstruktionsplattform **Makeblock**. Baukastensystem mit Sensoren und Aktoren zum Programmieren mit Scratch. Online: <http://www.makeblock.cc/>
- **Pico-Board**. Mikrocontroller speziell für Scratch. Online: <http://scratch-dach.info/wiki/PicoBoard>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B5.1	Die ersten Schritte mit Scratch	Arbeitsblatt mit Einstiegstutorial zum Kennen- und Anlernen der Entwicklungsumgebung Scratch.
 B5.2	Scratch-Würfel	Bau-/Kopiervorlage für ein Set Scratch-Würfel mit Bausteinen aus Scratch zum Programmieren von zufälligen Programmen.
 B5.3	Ein Scratch-Programm erwürfeln	Arbeitsblatt für das Scratch-Würfel-Set. Nutzung für den kreativen Entwurf eines zufälligen Programms mit Scratch.
 B5.4	Projektarbeit	Arbeitsblätter mit Hinweisen und Aufgaben für eine Projektarbeit mit Scratch.
 B5.5	Scratch kennenlernen	Arbeitsblatt als Alternativeinstieg. Analyse und Vergleich verschiedener Programme.
 B5.6	Bühnenbild und Bewegungen	Arbeitsblatt als Alternativeinstieg. Umgang mit dem Bühnenbild und Bewegungen.

 B5.7	Animationen und Geschichten erzählen	Arbeitsblatt als Alternativeinstieg. Erstellung von Animationen mit Scratch und Programmieren einer Kurzgeschichte.
 B5.8	Nachrichten und Variablen	Arbeitsblatt als Alternativeinstieg. Umgang mit Nachrichten und Variablen.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Algorithmus	Als Algorithmus wird eine Handlungsvorschrift bezeichnet, deren einzelnen Handlungsanweisungen eindeutig und deterministisch (endlich; zeitlich begrenzt) sind. Algorithmen beschreiben meist, wie gegebene Problemstellungen gelöst werden oder bestimmte Tätigkeiten durchzuführen sind.
Attribut	Ein Attribut ist eine zu einem Objekt (in Scratch eine Figur) gehörende Variable, mit deren Hilfe die Datenwerte einer bestimmten Eigenschaft verwaltet werden. Solche Datenwerte nennt man auch Attributwerte.
Attributwerte	vgl. Attribut.
(Scratch-)Bausteine	Bausteine werden genutzt, um Programme in Scratch zu erzeugen. Die Bausteine lassen sich wie Puzzleteile miteinander verbinden – es passt nur zusammen, was einen syntaktisch richtigen Quellcode ergibt, dadurch werden Syntaxfehler verhindert. (Scratch-)Bausteine werden auch <i>Blöcke</i> genannt.
Blöcke	vgl. (Scratch-)Bausteine.
Bühne	Die Bühne ist der Bereich des Scratch-Programms, den der Nutzer sieht. Auf ihr sind die verschiedenen Aktionen der einzelnen Figuren sichtbar. Analog zu den Figuren kann auch der Hintergrund der Bühne nach Belieben angepasst und innerhalb der Programmlaufzeit verändert werden.
Entwicklungsumgebung	vgl. IDE.
Figur	Figuren werden in Scratch auf der → Bühne (Stage) platziert und können dort Aktionen wie Bewegungen und Aussagen (Sprechblase, Gedankenblase) ausführen. Sie müssen nicht immer auf der Bühne sichtbar sein. Während der Programmlaufzeit kann die Darstellungen (deren Bilder) verändert werden.

IDE	Ist eine Abkürzung und steht für „Integrierte Entwicklungsumgebung“; abgeleitet von englisch „integrated development environment“. Gemeint ist hier die Umgebung beziehungsweise das Programm, mit dem in Scratch programmiert wird.
(Scratch-) Kategorien	Die Scratch-Programmoberfläche ist in drei Hauptbereiche (→ Bühne, Liste der → Bausteine und → Skriptfenster) eingeteilt. Im Bereich Kategorien sind ähnliche Bausteine zusammengefasst und farbkodiert. Kategorien von Scratch 2: Bewegung, Ereignisse, Aussehen, Steuerung, Klang, Fühlen, Malstift, Operatoren, Daten, weitere Bausteine. Die Kategorien der Programmoberfläche sind farbcodiert.
Kontrollstruktur	Eine Kontrollstruktur stellt einen Scratch-Baustein dar, mit dem der Ablauf des Programms gesteuert werden kann. Beispiele hierfür sind die [Wenn]-, [Wenn, Sonst]- oder [Wiederhole]-Bausteine.
Methode	vgl. Skript.
Objekt	vgl. Figur.
Programm	Bezeichnet die Gesamtheit aller verwendeten → Figuren, → Bühnen und → Skripte in Scratch. Programme können mit der grünen Fahne ausgeführt und dem roten Punkt gestoppt werden.
Programmierungsumgebung	vgl. IDE.
Sequenz	Unter einer Sequenz wird eine Abfolge von Anforderungen verstanden, die nacheinander abgearbeitet werden. Ähnlich aufzufassen wie → Skript.
Skript	Ein Skript besteht aus mehreren Handlungsanweisungen beziehungsweise Bausteine und wird einer Figur zugeschrieben.
Skriptfenster	Einer der Hauptbereiche der Scratch-Entwicklungsumgebung. Die Bausteine werden per Drag-and-Drop“ in das Skriptfenster hineingezogen und können dort zu Skripten zusammengesetzt werden.
Sprite	vgl. Figur.
Stage	vgl. Bühne.
Storyboard	Ein Storyboard bildet die Inhalte und Handlungsverläufe eines Drehbuchs visuell ab und dient so als Vorlage für die Erstellung eines Films. Die Handlungen aller Akteure werden hierbei auf einem Zeitstrahl dargestellt.

**Anmerkung:** Bezeichnungen in der Programmoberfläche können lokalisiert beziehungsweise individuell angepasst werden. Darum sind die Bezeichnungen nicht immer einheitlich.

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

## Die ersten Schritte mit Scratch

Dieses Tutorial hilft dir bei den ersten Schritten mit Scratch. Nachdem du die einzelnen Übungen durchgearbeitet hast, kannst du selber kreativ werden.

### Aber was ist Scratch?

Mit der Programmiersprache Scratch kann man eigene Computerprogramme schreiben. Hierfür werden farbige Programmier-Bausteine verwendet, die man wie Puzzleteile zusammensetzt. Mehrere zusammengesetzte Bausteine nennt man Skript. Sie werden vom Computer Schritt für Schritt abgearbeitet.

### Aufgabe

Arbeite das Tutorial von Scratch sorgfältig durch. Dabei wirst du viele Möglichkeiten von Scratch entdecken und dich mit der Entwicklungsumgebung vertraut machen. Zum Schluss wirst du sogar dein erstes Spiel mit Scratch programmiert haben.

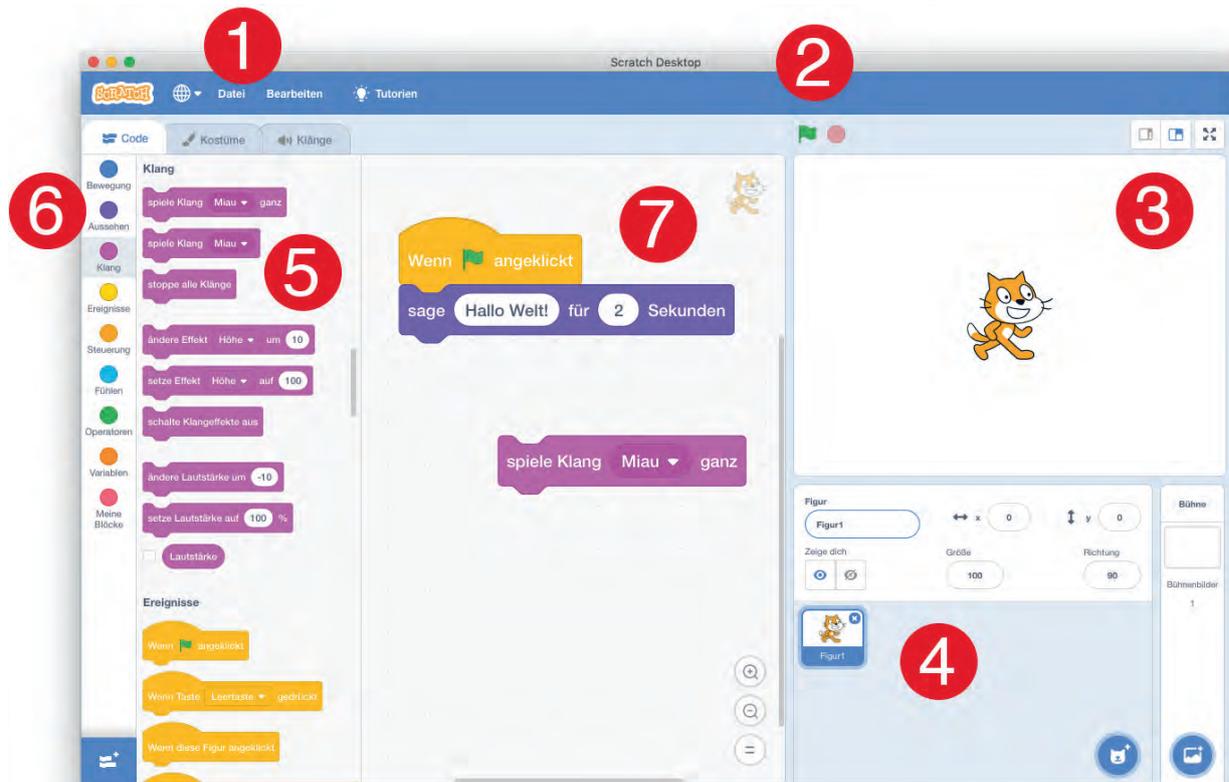
### Wichtiger Hinweis

Achte darauf, dass du deinen Fortschritt ausreichend abspeicherst. Lasse deiner Kreativität genügend Platz: Es darf auch einmal eine andere Figur als die vorgeschlagene ausgewählt werden.



## Scratch-Tutorial I

## Entwicklungsumgebung



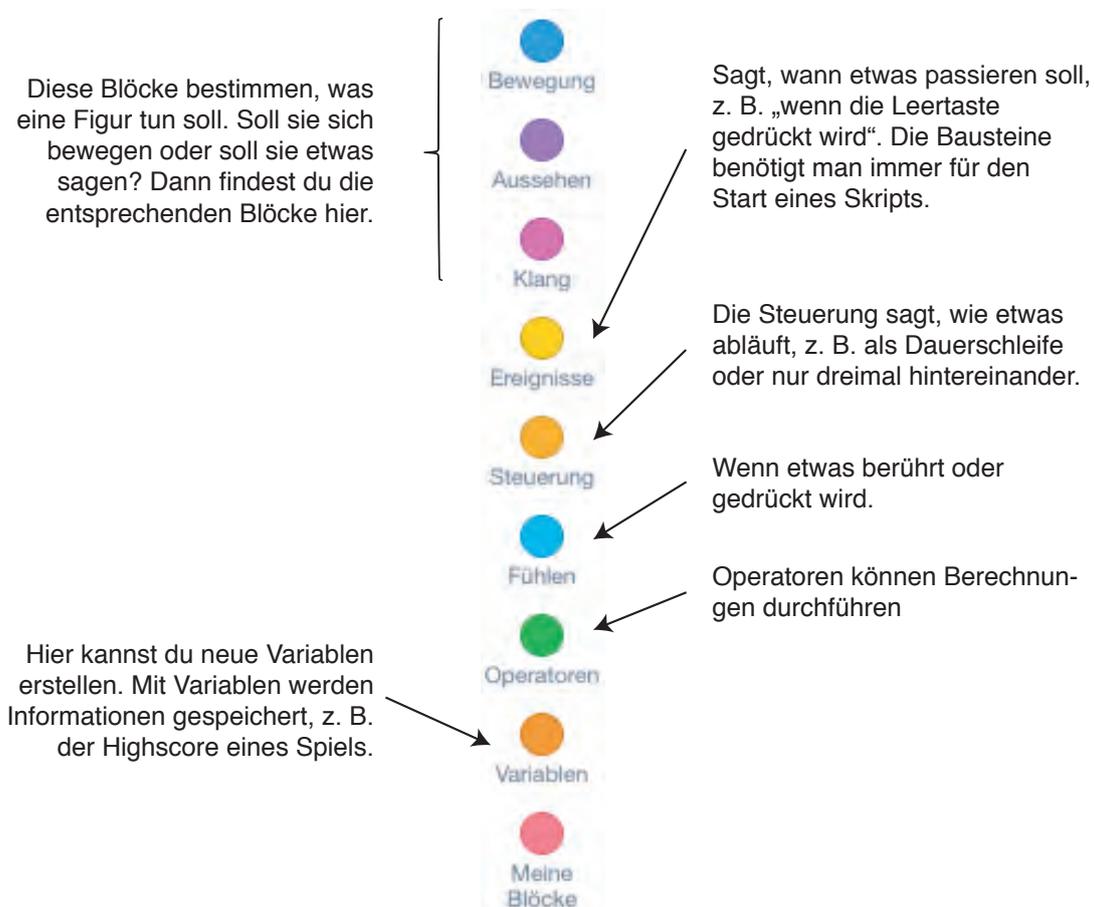
Die folgenden Erläuterungen beziehen sich auf die mit den Zahlen gekennzeichneten Bereiche:

1. In diesem Bereich findest du verschiedene Reiter wie „Datei“ oder „Bearbeiten“, das sind sogenannte Menüs. Man kann sie durch Anklicken „ausklappen“ (engl.: drop down = herunterfallen). Dadurch findest du zum Beispiel Möglichkeiten zum Speichern und Laden von Programmen.
2. Mit der grünen Fahne werden Programme in Scratch gestartet und mit dem roten Sechseck gestoppt.
3. Dieser Bereich stellt die Bühne (Stage) dar. Hier werden Figuren (Objekte oder auch Sprites genannt) positioniert und führen Aktionen aus. Alles was die Nutzer sehen und durch die Programme von Scratch ausgeführt wird, ist hier dargestellt.
4. Figuren müssen nicht immer auf der Bühne sichtbar sein (sie können sich auch verstecken). Damit man weiß, dass diese im Programm enthalten sind, werden alle Figuren in diesem Bereich aufgelistet. Links davon wird die Bühne in Klein angezeigt.
5. Anweisungen beziehungsweise Befehle, die der Computer ausführen soll, haben in Scratch die Form von Bausteinen. In diesem Bereich sind die Befehle/Bausteine aufgelistet.

- Um nicht immer eine RIESIGE Liste an Bausteinen angezeigt zu bekommen, sind die Bausteine in Scratch in Kategorien zusammengefasst. Sie sind hier zu finden. Wenn man eine andere Kategorie auswählt, dann erscheinen andere Bausteine im Bereich 5.
- Um mit Scratch zu programmieren, werden einzelne Bausteine in das Skriptfenster gezogen. Werden mehrere Bausteine miteinander verbunden, spricht man von einem Skript. In diesem Fenster werden immer nur die Skripte der ausgewählten Figur oder der Bühne angezeigt. Mehrere Figuren können also verschiedene Skripte haben. Du kannst dir die Skripte ähnlich vorstellen wie kleine Unterprogramme. Als Programm wird alles bezeichnet, was du verwendet hast, also die Bühne sowie alle Figuren und Skripte.

## Wie funktioniert Scratch

Scratch besteht aus einer Vielzahl an farbigen Bausteinen, die zu Kategorien zusammengefasst wurden. Schau dir mal den Bereich 6 genauer an. Mit einigen Bausteinen kann man Bewegung und Aussehen ändern oder zeichnen.



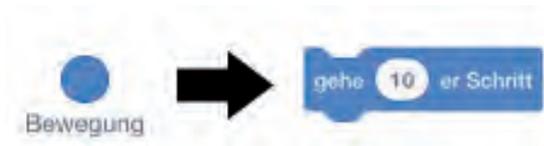
## Das erste Programm: Bewegungen

Zu Beginn wirst du den Umgang mit den Bewegungsfunktionen kennenlernen. Verwende dazu die beim Programmstart dargestellte Katze Scratchy und versuche, ihr einige Bewegungen beizubringen.

## Aufgaben

Versuche zunächst, die Katze von links nach rechts laufen zu lassen.

1. Ziehe aus der „Bewegung“-Kategorie folgenden Baustein in das Skriptfenster:



2. Klicke mit der Maus auf den gerade hinzugefügten Baustein im Skriptfenster. Die Katze Scratchy sollte sich in 10er-Schritten von links nach rechts bewegen. Die Zahl gibt an, um wie viele Pixel sich die Katze vorwärtsbewegt. Je größer die Zahl, desto schneller ist die Geschwindigkeit.
3. Damit sich die Katze bewegt, sobald die grüne Fahne geklickt wird, brauchst du eine bestimmte Startbedingung. Suche den folgenden Baustein in der „Ereignisse“-Kategorie und verbinde ihn durch Ziehen ins Skriptfeld mit dem anderen Baustein.



4. Wenn du alle Schritte richtig befolgt hast, dann sollte dein Skript so aussehen:



Wenn du jetzt auf die grüne Flagge klickst, wirst du sehen, wie sich die Katze einmal vorwärts bewegt.

### Zusatzaufgabe

Probiere auch die rechts abgebildeten Startbedingungen mit der Vorwärtsbewegung zu verknüpfen, und teste, wie sie funktionieren.



Vielleicht fragst du dich ja: Warum bewegt sich die Katze eigentlich von links nach rechts? Unter der Bühne siehst du Folgendes:



Der „x“- und „y“-Wert gibt die Position auf der Bühne an. Diese Werte nennt man auch Koordinaten. Wenn du mit dem Mauszeiger über die Bühne fährst, siehst du direkt unterhalb der Bühne die x- und y-Koordinaten des Mauszeigers auf der Bühne.

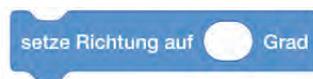
Im Bereich der „Bewegen“-Kategorie ist dir sicherlich schon Folgendes aufgefallen:



Wenn du dort die Haken setzt, dann kannst du dir die Position und Richtung der Katze auf der Bühne anzeigen lassen.

## Aufgabe

1. Finde heraus, welche Koordinaten die Mitte der Bühne hat. Kannst du auch sagen, welche Koordinaten die rechte obere Ecke hat?
2. Versuche, das Skript so zu verändern, dass sich die Katze nach oben bewegt, wenn du auf die grüne Flagge drückst. Dieser Baustein sollte dir dabei helfen:



## Schleifen

Bis jetzt musstest du, damit die Katze sich einmal bewegt, immer wieder auf die grüne Flagge drücken. Nun soll die Katze aber mit einmal Drücken immer weiter gehen, bis sie ganz am rechten Rand angekommen ist. Dafür benötigt man eine Wiederholungsanweisung, diese nennt man Schleife. Ein Beispiel dafür ist der „wiederhole fortlaufend“-Baustein:



Alles was sich innerhalb dieses Bausteines befindet, wird immer wieder ohne ein definiertes Ende ausgeführt. Damit die Katze nun also mit einmal Drücken der grünen Flagge ganz nach rechts läuft, muss das Skript so aussehen:



Übrigens kannst du durch Drücken des roten Sechsecks die Schleife abbrechen.

## Aufgabe

1. Baue die „wiederhole fortlaufend“-Schleife in dein Programm ein. Ändere dazu das bisherige Skript und probiere aus, was passiert, nachdem die Katze ganz rechts angekommen ist.

Du kannst die Katze mit der Maus wieder an eine andere Position auf der Bühne ziehen, um sie erneut, nach Drücken der grünen Flagge, in Bewegung zu sehen.

2. Praktisch ist es, wenn die Katze automatisch an eine Startposition geht, bevor sie mit der Bewegung beginnt. Ergänze dafür dein Skript entsprechend. Verwende dazu folgenden Baustein:



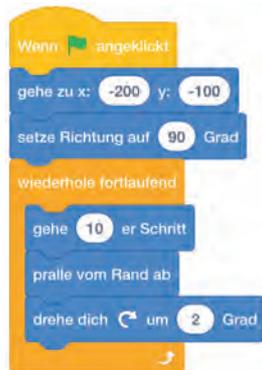
3. Nun soll die Katze, wenn sie auf den Rand der Bühne trifft, in eine andere Richtung weiterlaufen. Am besten so lange, bis das rote Sechseck gedrückt wird. Ändere dein Skript entsprechend. Der folgende Baustein könnte dir behilflich sein:



4. Damit die Katze nicht nur hin und her läuft, kannst du sie sich drehen lassen:



Alles zusammen, kann dann so aussehen:



## Zusatzaufgabe

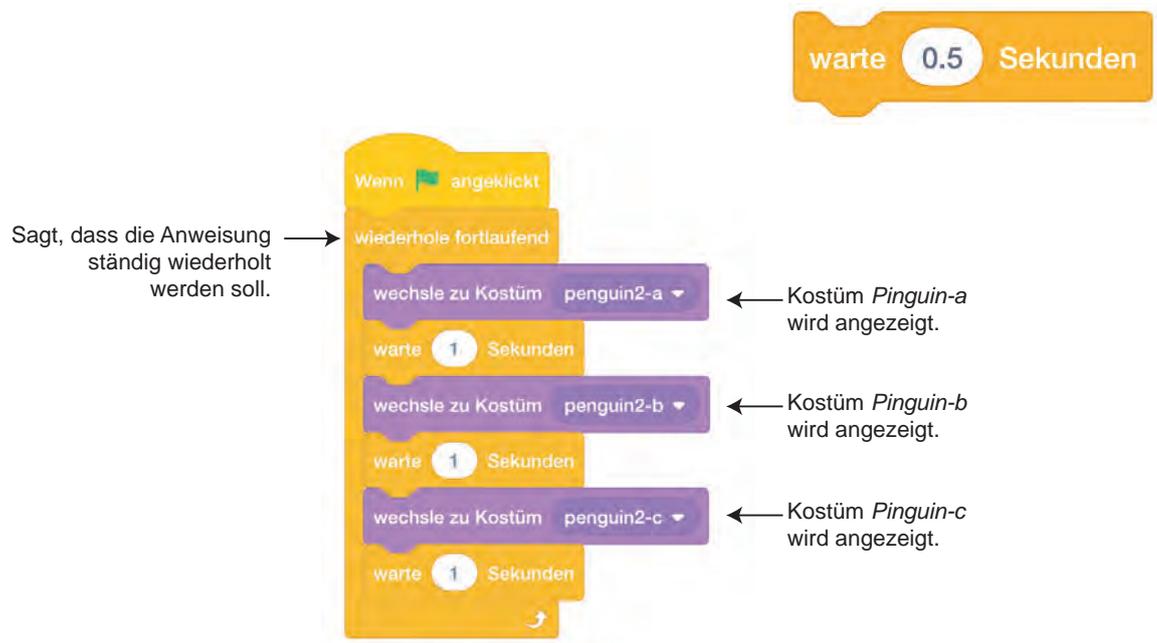
Probiere aus, den einen oder anderen Baustein wegzulassen, und beobachte, was passiert. Welche Bausteine sind unbedingt notwendig und welche kann man auch weglassen?

## Aussehen/Kostüme

Versuch nun etwas Neues. Speichere dazu dein Programm ab und klicke im Menü auf „Datei“ und dann auf „Neu“ um ein neues Programm zu erstellen. Anschließend kannst du die Katze mit Rechtsklick auf die Figur von der Bühne löschen. Danach musst du eine neue Figur hinzufügen, wähle hierfür den Pinguin aus. Wenn du nun auf den Karteireiter „Kostüme“ klickst, dann siehst du, dass es den Pinguin in mehreren Varianten gibt.



Man kann programmieren, dass der Pinguin sein „Kostüm“ wechselt. So bekommt man ziemlich schnell Bewegung rein. Baue Folgendes nach:

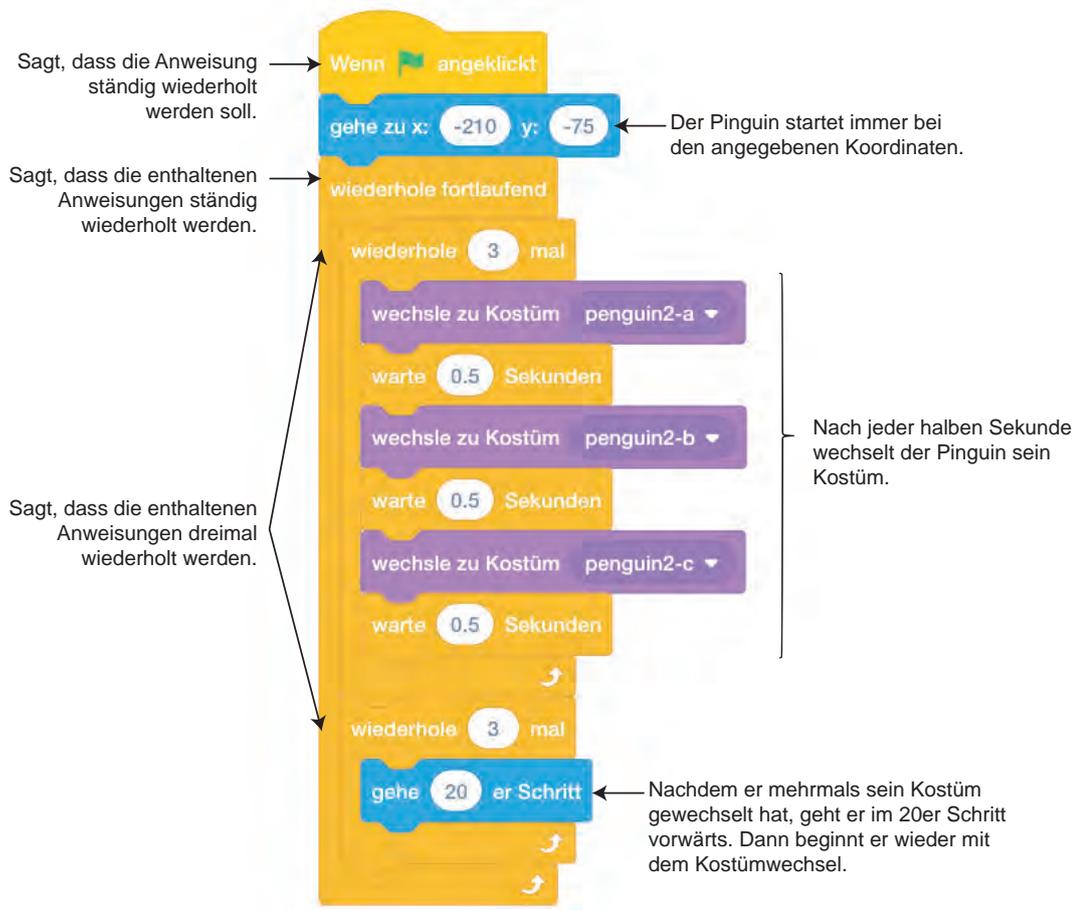


## Aufgabe

Was passiert, wenn du keinen „Warte-Block“ einfügst oder du die Zeit im „Warte-Block“ veränderst?

**Tipp:** Wenn der Pinguin sein Kostüm schneller wechseln soll, muss das Warten kürzer werden. Verwendet wird hier aber die englische Schreibweise, bei der man einen **Dezimalpunkt** verwendet und kein **Dezimalkomma**.

Du kannst nun auch programmieren, dass sich der Pinguin nicht nur auf der Stelle bewegt, sondern auch von links nach rechts:



## Hex-Hex – vom Verschwinden und wieder Auftauchen

Es gibt viele verschiedene Möglichkeiten, Figuren unsichtbar, größer oder kleiner zu machen oder sogar die Farbe zu wechseln.

Suche dir wieder eine Figur aus, vielleicht eine kleine Hexe oder einen Geist. Die nötigen Bausteine findest du in der Kategorie „Aussehen“. Hinter den Bausteinen verstecken sich viele interessante Möglichkeiten.



Die Programmierung beginnt mit dem Baustein „Schalte Grafikeffekte aus“. Dies macht man, damit die Figur erst einmal ohne einen Effekt startet. Andernfalls kann es passieren, dass man vielleicht schon irgendwo einen Effekt programmiert hat und die Figur zu Beginn unsichtbar ist.

Versuche einmal Folgendes:

The script consists of the following blocks:

- Wenn angeklickt** (When clicked) block containing:
  - schalte Grafikeffekte aus** (Turn off all effects) block. Annotation: "Wenn das Skript startet, dann werden alle Effekte gestoppt."
  - wiederhole fortlaufend** (Repeat forever) loop containing:
    - wiederhole 10 mal** (Repeat 10 times) loop containing:
      - ändere Effekt Durchsichtigkeit um 10** (Change opacity by 10) block. Annotation: "Die Figur wird immer blasser und verschwindet dann vollkommen."
    - gehe zu x: Zufallszahl von -200 bis 200 y: Zufallszahl von -150 bis 150** (Go to random x and y) block. Annotation: "Die Figur geht zu einer beliebigen und zufälligen Position auf der Bühne."
    - wiederhole 10 mal** (Repeat 10 times) loop containing:
      - ändere Effekt Durchsichtigkeit um -10** (Change opacity by -10) block. Annotation: "Die Figur erscheint langsam wieder an der neuen Position."
    - warte 1 Sekunden** (Wait 1 second) block.

## Aufgabe

- Verändere die Zahlen: zum einem im „Wiederholen-Baustein“ und zum anderem beim „Durchsichtigkeit-Effekt“. Welche Auswirkungen hat das?
- Probiere doch nun einmal die anderen Effekte aus: Farbe, Wirbel, Pixel und so weiter. Welcher gefällt dir am besten?

**Tipp:** Vielleicht ist dir aufgefallen, dass die grünen Bausteine nicht wie die anderen aneinandergereiht werden. Es sind sogenannte Operatoren, sie werden immer in andere Bausteine eingesetzt. Das gilt auch für einige andere Bausteine aus dem Bereich „Fühlen“.

## Scratch-Tutorial II

In diesem zweiten Teil wirst du ein eigenes Computerspiel programmieren. Dafür brauchst du drei Darsteller: einen, der etwas wirft, einen, der abgeworfen wird, und ein Objekt, das geworfen wird. Als Beispiel wurden in diesem Tutorial ein großer Seestern, ein kleiner Seestern und ein



Tintenfisch ausgewählt.



Wähle für dein Computerspiel drei Figuren und einen passenden Hintergrund aus. Du kannst unsere Vorschläge übernehmen oder dir etwas Eigenes aussuchen.

1. Zu Beginn wählst du die Figur aus, die werfen soll. In diesem Beispiel ist das der pinkfarbene Seestern. Der Seestern soll am Anfang immer an einem bestimmten Startpunkt im Koordinatensystem stehen und er soll mit den Pfeiltasten bedient werden können. Die Programmierung dazu sieht folgendermaßen aus:



Der Seestern startet immer bei den Koordinaten  $x=-15$  und  $y=-130$ .

Nach dem Start soll die Bewegung mit Hilfe der Pfeiltasten durchgängig durchgeführt werden.

Damit der Seestern beim Richtungswechsel nicht überkopf steht, wird der Drehtyp auf *links-rechts* gesetzt.

Dies ist die Programmierung für die Steuerung mit den Pfeiltasten:  
 Falls die Pfeiltaste nach links gedrückt wird, dann soll der Stern nach links laufen (Richtung  $-90$ ) und zwar in 10er-Schritten  
 Dadrunter sind die Bausteine für das Laufen in die rechte Richtung.

```

  Wenn grünes Flagge angeklickt
  -> gehe zu x: -15 y: -130
  -> setze Drehtyp auf links-rechts
  -> wiederhole fortlaufend
    -> falls Taste Pfeil nach links gedrückt? dann
      -> setze Richtung auf -90 Grad
      -> gehe 10 er Schritt
    -> falls Taste Pfeil nach rechts gedrückt? dann
      -> setze Richtung auf 90 Grad
      -> gehe 10 er Schritt
  
```

Die Programmierung für unseren ersten Darsteller ist nun fertig. Wenn du auf das grüne Fähnchen klickst, musst du ihn mit Hilfe der Pfeiltasten nach links oder rechts laufen lassen können.

2. Wähle jetzt die Figur aus, die abgeworfen werden soll, in unserem Fall ist das der Tintenfisch. Dir fällt sicher auf, dass das Skriptfenster für die neue Figur leer ist, dies ist so, weil dort immer nur die Skripte der ausgewählten Figur angezeigt werden. Der Tintenfisch soll sich immer hin und her bewegen und das in unterschiedlichen Geschwindigkeiten. Das soll er so lange machen, bis er vom Wurfgeschoss getroffen wird – dann soll er herunterfallen. Bevor du damit starten kannst, musst du eine neue Variable erstellen.

Variablen sind Platzhalter, in denen Informationen und Werte gespeichert werden. Meistens verwendet man sie, um Punkte zu zählen oder um eine Stoppuhr laufen zu lassen. Um eine Variable zu erstellen, klickst du auf „Variablen“ und dann auf „Neue Variable“. Daraufhin sollte folgendes Fenster erscheinen:



Name der Variable:  
Nenne sie Tempo oder Geschwindigkeit.

Hier kann man bestimmen, ob die Variable auf alle Figuren angewendet wird oder nur für diese Figur. In unserem Fall wird die Variable nur von dem Tintenfisch verwendet.

Weiter geht es mit der Programmierung:

Hier beginnt die Schleife.

Der Tintenfisch dreht sich nach rechts.

Der Tintenfisch wird wieder sichtbar.

Das Tempo wird auf einen Zufallswert zwischen 3 und 15 gesetzt.

Das *sende getroffen an alle* wird für einen späteren Highscore benötigt.

Damit der Tintenfisch beim Richtungswechsel nicht überkopf steht, wird der Drehtyp auf *links-rechts* gesetzt.

Der Tintenfisch erscheint auf der linken Seite der Bühne ( $x=-300$ ), aber auf einer zufälligen Höhe ( $y$ =Zufallszahl von 50 bis 150)

Diese Programmierung sagt, dass der Tintenfisch sich so lange hin und her bewegen soll, bis er getroffen wird.

Diese Teil wird ausgeführt, nachdem der Tintenfisch getroffen wurde: Zuerst wird eine Nachricht an alle anderen Objekte geschickt. Dann soll er in 10er-Schritten nach unten fallen und durchsichtiger werden. Dies soll 60-mal wiederholt werden, damit er auch unten ankommt.

Übrigens kannst du beim „senden an alle“-Baustein auch den Text der Nachricht beispielsweise auf „getroffen“ ändern. Dazu musst du einfach im Dropdown-Menü „neue Nachricht...“ auswählen. Es sollte sich dann folgendes Fenster öffnen:



Wenn du dein Spiel jetzt testest, dann kannst du deinen Werfer mit den Pfeiltasten hin und her bewegen und deine Figur, die abgeworfen wird, bewegt sich selbstständig in unterschiedlicher Geschwindigkeit hin und her.

Du solltest bei der Programmierung darauf achten, dass die Skripte für die verschiedenen Figuren auch im richtigen Skriptfenster sind. Sonst kann es zu Fehlern im Programm kommen.

3. Nun benötigst du noch etwas zum Werfen. In unserem Beispiel ist das der kleine Seestern. Die Programmierung für ihn sieht wie folgt aus:

Der kleine Seestern erscheint zu Beginn auf der angegebenen Position.

Der kleine Seestern zeigt sich (er wird in einer anderen Programmierung noch auf „verstecken“ gestellt).

Der Seestern wartet, bis ihn der große Seestern berührt, danach geht er zum großen Seestern, bis die Leertaste gedrückt wird.

Diese Programmierung sagt, was passiert, wenn die Leertaste gedrückt wird: Dann fliegt der Seestern nach oben.

Die Programmierung sagt, wo der kleine Seestern nach dem Wurf wieder auftauchen soll. Je nach Zufallszahl (1 oder 2), erscheint der Stern bei  $x=123$ ,  $y=-140$  oder  $x=-180$ ,  $y=-140$ .

4. Deine Figur sollte nun eine andere Figur aufnehmen und eine dritte Figur damit bewerten können. Damit das Ganze noch interessanter wird, kannst du noch Punkte zählen. Das geht so:

Zuerst erstellst du eine neue Variable für alle Figuren, diese nennst du „**Punkte**“.

**Zur Erinnerung:** Dem Tintenfisch hast du gesagt, er soll „getroffen“ senden, wenn er vom kleinen Seestern berührt wird. Das kannst du nun nutzen und dem kleinen Seestern Folgendes hinzufügen:

# Arbeitsmaterial B5.1



Wenn du „getroffen“ empfangst, dann verstecke dich, spiele einen Klang und ändere die Punkte um 1. Da der Seestern sich hier versteckt, mussten wir in der vorherigen Programmierung das „zeige dich“

Damit der Punktestand bei mehrmaligem Spielen immer bei „0“ beginnt, müssen wir das dem Computer auch noch sagen.



5. Zum Schluss legst du fest, dass das Computerspiel nach einer bestimmten Zeit zu Ende ist. Dafür wählst du diesmal keine Figur aus, sondern den Hintergrund/die Bühne. Darüber hinaus brauchst du wieder eine neue Variable, diesmal für die Zeit. Klicke dafür wie gewohnt auf „Daten“ und erstellst eine neue Variable.

Danach erstellst du folgende Programmierung:

Das Spiel endet mit einem Grafikeffekt, daher müssen wir dem Computer sagen, dass zu Beginn alle Effekte ausgeschaltet werden.

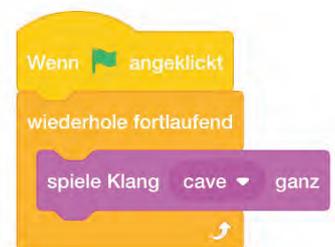


Dieser Block sagt, dass zu Beginn die Zeit auf „30“ gestellt werden soll.

Danach wird die Zeit jede Sekunde um 1 verringert, bis die Zeit auf „0“ steht.

Nachdem die Zeit abgelaufen ist, gibt es einen Grafikeffekt, sozusagen als „Game over“ – du kannst dir aber auch ein anderes Ende überlegen.

Wenn du willst, kannst du nun deinem Computerspiel noch eine Hintergrundmusik geben. Das programmierst du auch auf dem Hintergrund.



Und nun bist du an der Reihe. Entwickle doch mal ein eigenes Spiel oder produziere einen Trickfilm!

# Scratch-Würfel

Es folgen zunächst alle für Bausteine aus Scratch, die sich unserer Ansicht nach für einen Würfel eignen. Anschließend folgen die Bastelvorlagen für die Scratch-Würfel.

**Die Bausteine in den Bauvorlagen der Würfel können auf Wunsch mit den hier folgenden Bausteinen ersetzt werden!**

## Ereignis-Bausteine

Wenn das Bühnenbild zu Bühnenbild1 wechselt	Wenn angeklickt wird
Wenn diese Figur angeklickt wird	Wenn ich empfangt
Wenn Taste gedrückt wird	Wenn Stoppuhr > 10
sende an alle	

## Steuerungs-Bausteine

warte Sekunden	wiederhole fortlaufend
wiederhole mal	wiederhole bis
falls , dann	falls , dann sonst

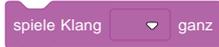
## Fühlen-Bausteine

Taste gedrückt?	wird berührt?
frage und warte	Entfernung von
Antwort	

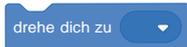
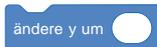
## Operatoren-Bausteine

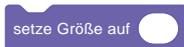
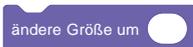
## Klang-Bausteine

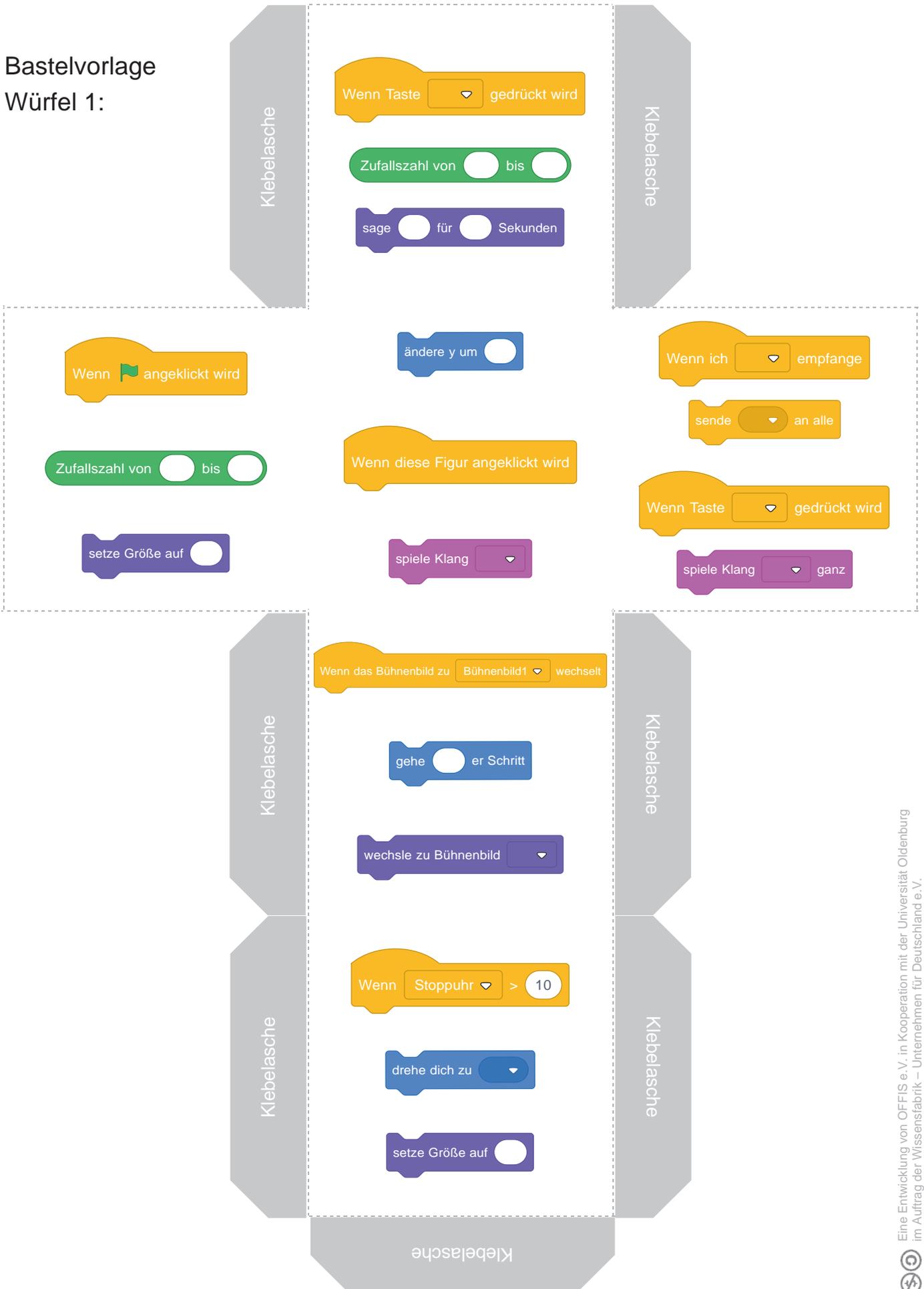
## Bewegungs-Bausteine

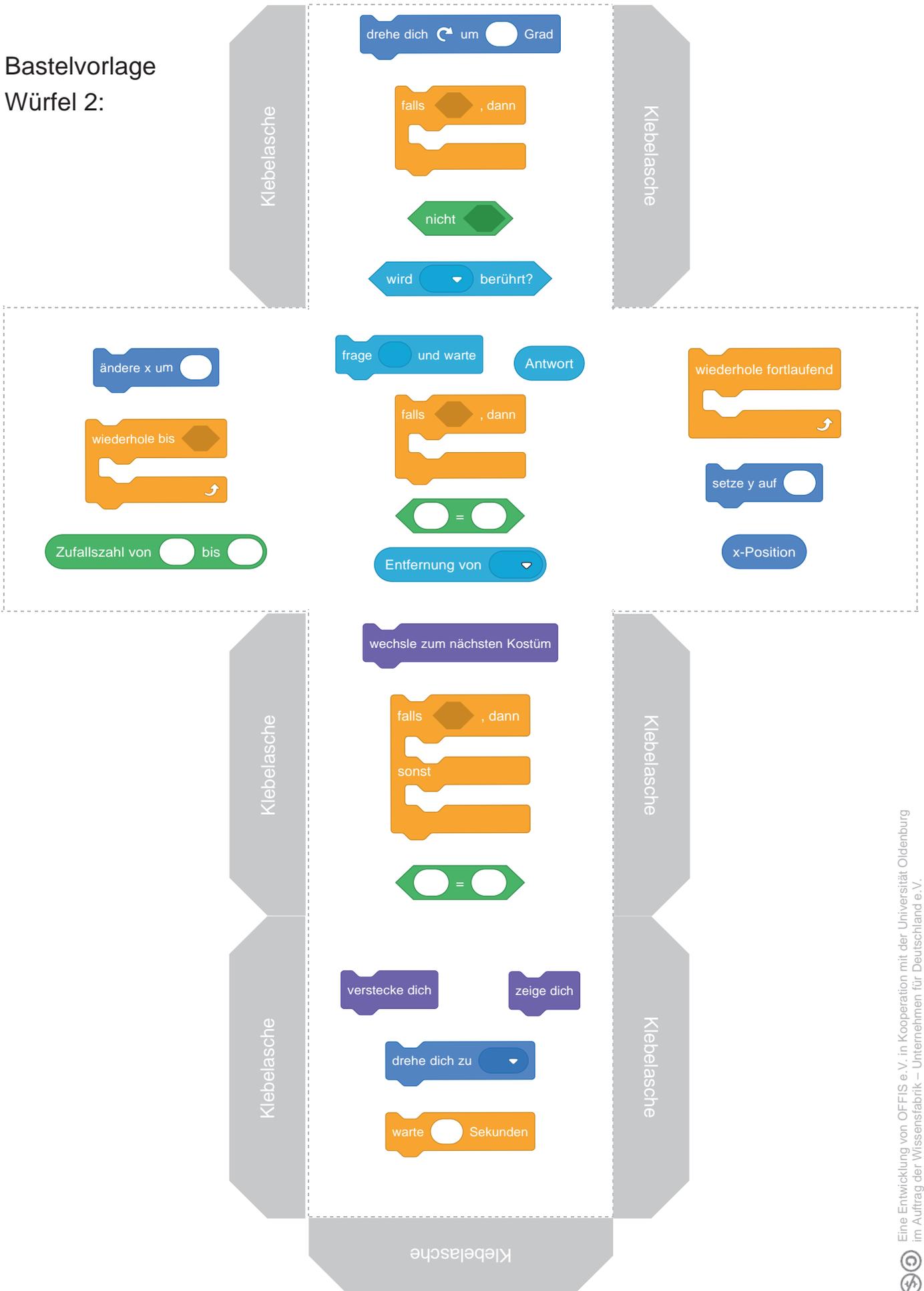
## Aussehen-Bausteine

# Bastelvorlage Würfel 1:



# Bastelvorlage Würfel 2:



# Ein Scratch-Programm erwürfeln

Falls nicht mitgeliefert, bastle dir zusammen mit deiner Sitznachbarin oder deinem Sitznachbarn die Scratch-Würfel aus dem Material 5.2 und rufe dann an einem Computer das Programm Scratch auf.

## Aufgaben

1. Nimm dir zusammen mit deiner Sitznachbarin oder deinem Sitznachbarn ein Set an Scratch-Würfeln und würfelt sie.
  - a. Nehmt pro Würfel einen Baustein, sucht ihn in Scratch und zieht ihn aus der Befehlsliste in das Skriptfenster. Ihr solltet nun also zwei Bausteine im Skriptfenster haben. Verwendet die gewürfelten Bausteine in einem beliebigen Skript, das ihr sinnvoll durch weitere Scratch-Bausteine ergänzen könnt. Was beobachtet ihr?  
  
Beschreibt nun gemeinsam in wenigen Sätzen, was euer erwürfeltes Programm tut und vergleicht euer Programm mit dem Programm eurer Nachbargruppe.
  - b. (*Fortgeschritten*) Nehmt alle Programmierbausteine, die ihr gewürfelt habt, und geht wie unter Aufgabenteil a vor.



## Projektaufgabe

Entwickelt euer eigenes Computerspiel oder eine interaktive Geschichte mit Scratch. Arbeitet dafür zusammen im Team.

**Teamgröße:** 2er- oder 3er-Teams

**Zeit:** 3 Unterrichtsstunden



### Aufgabe

1. Überlegt euch, was ihr gerne machen wollt (Geschichte oder Computerspiel).
2. Entwickelt ein Storyboard für die Geschichte beziehungsweise überlegt euch ein Konzept und ein Design für das Computerspiel.
3. Programmiert mit Scratch eure Animation oder euer Computerspiel. Nutzt das Scratch-Wiki zur Hilfe:

[www.scratch-dach.info/wiki/Hilfe:Inhaltsverzeichnis](http://www.scratch-dach.info/wiki/Hilfe:Inhaltsverzeichnis)

4. Testphase: Nachdem ihr alles fertig habt, bittet eine Mitschülerin oder einen Mitschüler aus einer anderen Gruppe, alles einmal zu testen. Ihr habt dann die Möglichkeit, Fehler zu beheben und Feedback einzuarbeiten.
5. Am Ende gibt es eine Präsentation aller Ergebnisse.

## Scratch kennenlernen

Mit Scratch lassen sich ganz einfach tolle Programme selbst kreieren. Doch bevor du richtig loslegen und ein eigenes Spiel oder Ähnliches entwickeln kannst, solltest du Scratch etwas näher kennenlernen.

Bearbeite dafür die folgende Aufgabenstellung.

### Aufgabe

Unter folgenden Pfaden/URLs findest du Beispiele für Programme, die mit Scratch erstellt wurden:

- ∞ <https://scratch.mit.edu/projects/1192/>
- ∞ <https://scratch.mit.edu/projects/13701368/>

Teste die aufgeführten Programme und notiere dir, welches Programm dir besonders gefallen hat. Versuche, die Abläufe des Programms nachzuvollziehen. Schaue dir dazu auch den Quellcode der Programme an (klicke dazu auf „Schau hinein“).

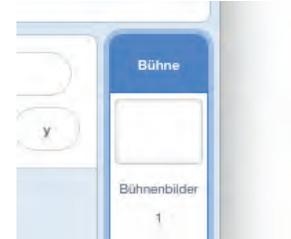
# Bühnenbild und Bewegungen

Jetzt kannst du zeigen, was du gelernt hast und selbst kreativ werden.

## Aufgabe 1

Die Bühne dient als Hintergrund für alle Figuren, die sich in deinem Programm bewegen. Gestalte eine passende Bühne zum Thema Ferien und setze ausgewählte Figuren auf die Bühne.

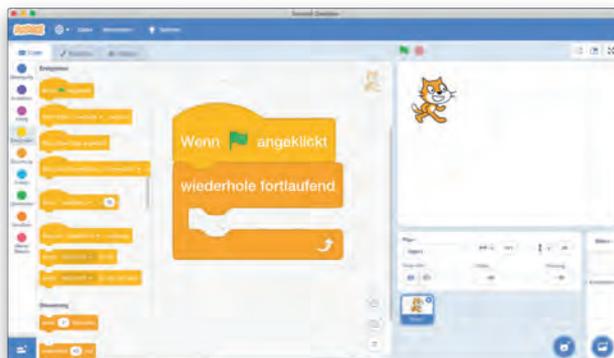
1. Dazu öffnest du ein neues Projekt und klickst auf das Symbol für die Bühne neben den Objekten (siehe Bild 1).
2. Unter dem Karteireiter Bühnenbilder können neue Hintergründe gezeichnet oder importiert werden (siehe Bild 2).
3. Gestalte mindestens einen neuen Hintergrund. Der erste, leere Hintergrund kann gelöscht werden.
4. Lösche Sprite1 (Katze) und füge eigene Figuren durch Auswahl aus bestehenden Figuren hinzu oder kreierte eine neue Figur, die auf deine Bühne passt (siehe Bild 3).
5. Am Ende sollte eine Ferienlandschaft mit ausgewählten Objekten entstehen.



## Aufgabe 2

Erstelle in Scratch ein neues Programm. Führe dann folgende Schritte aus:

1. Positioniere Scratchy ganz oben links auf der Bühne und lasse ihn mit Klick auf die Flagge über die Bühne nach rechts unten „laufen“ (das Kostüm von Scratchy sollte sich für diesen „Lauffeffekt“ bei jedem Schritt ändern).
2. Schreibe ein Skript, so dass Scratchy beim Klicken auf die Leertaste im Viereck läuft und an jeder Ecke etwas sagt.



# Animationen und Geschichten

## Aufgaben

Ziel ist es diesmal, eine kleine Animation mit Scratch zu erstellen. Dabei gilt natürlich, dass jede Animation nur so gut sein kann wie die Geschichte, die sie erzählt. Folge den Anweisungen:

1. Öffne das folgende Projekt mit Scratch und spiele die Geschichte durch:  
<https://scratch.mit.edu/projects/1758469/>
2. Mache dir nun selbst Gedanken und schreibe eine kurze Geschichte zum Thema Winter oder Sommer. Du könntest zum Beispiel von einem Urlaub erzählen.
3. Erstelle ein neues Projekt in Scratch, gestalte den Hintergrund und füge benötigte Figuren hinzu.
4. Versuche nun, die Figuren auf der Bühne zu bewegen und gezielt Dialoge zu führen.

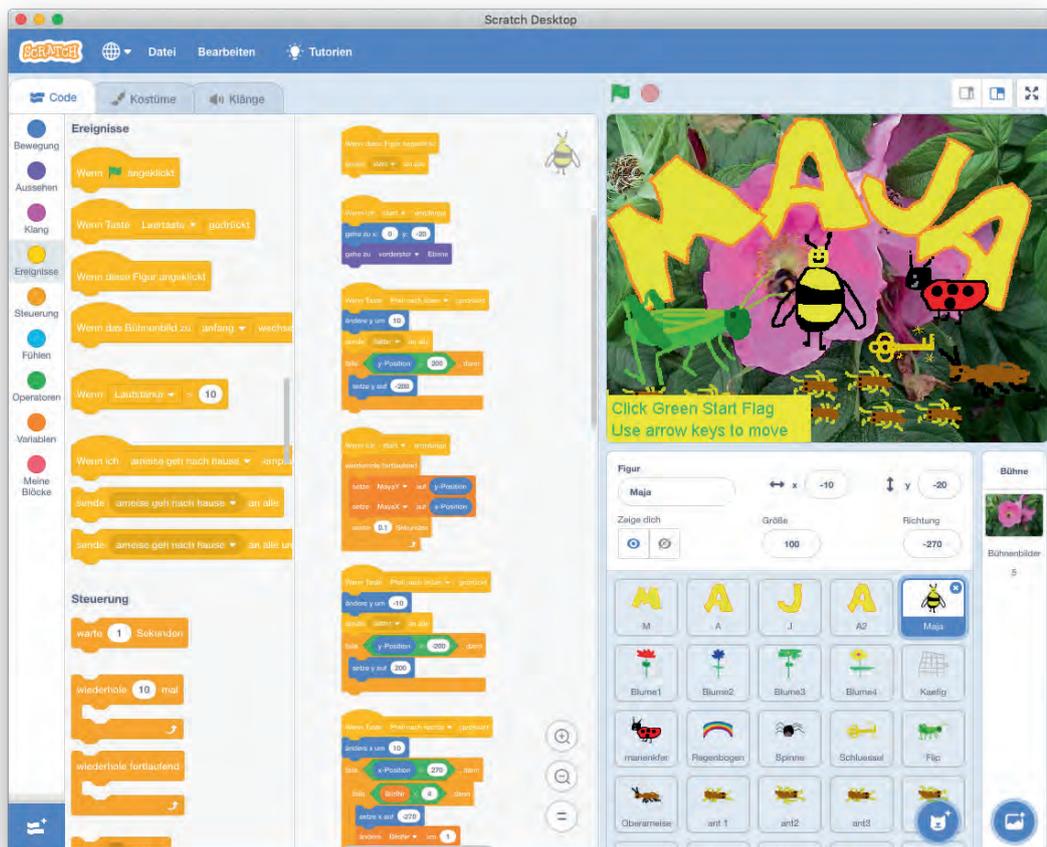


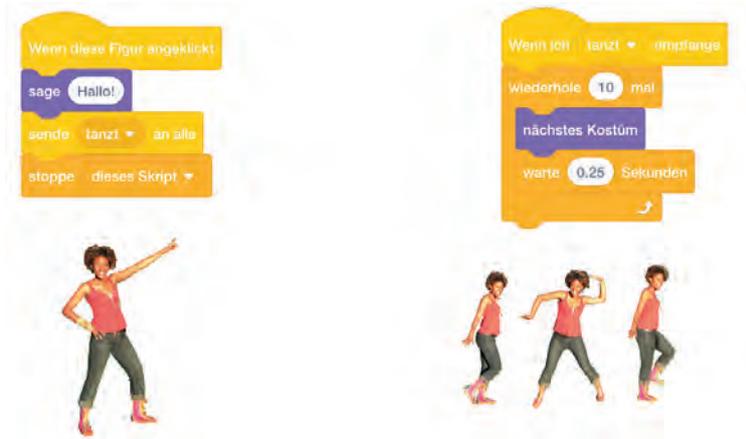
Abbildung: Ooreily Bee Story - <https://scratch.mit.edu/projects/1758469/>

# Nachrichten und Variablen

In Scratch können sich die einzelnen Figuren untereinander Nachrichten senden. Das sieht dann aber nicht so aus wie die Animationen mit den Sprechblasen. Dies kann manchmal sehr nützlich sein! Ebenso nützlich ist es, wenn du beispielsweise Spielstände abspeichern möchtest – dafür benötigst du Variablen.

## Aufgabe 1 – Nachrichten

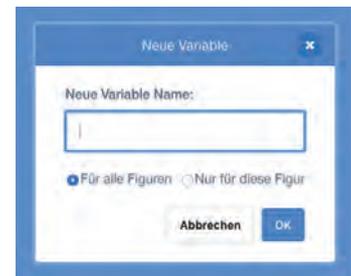
Erstelle die Animation einer Tanzgruppe. Die Tänzer sollen auf verschiedene Botschaften reagieren und unterschiedliche Tanzbewegungen vollziehen. Beispiele dazu siehst du hier:



## Aufgabe 2 – Variablen

Öffne ein neues Scratch-Programm und erstelle ein Reaktionsspiel.

1. Definiere eine neue Variable mit dem Namen „Punkte“ und zeige die Punkte auf der Bühne an. (Um die Punkte anzeigen zu lassen musst du die Kategorie „Daten“ öffnen und einen Haken bei der Variable setzen.)
2. Füge nun der Katze das Skript mit folgenden Anweisungen hinzu.



# Musterlösungen

## Tutorial II

Im Folgenden werden alle Skripte zu den einzelnen Figuren und der Bühne aufgelistet.

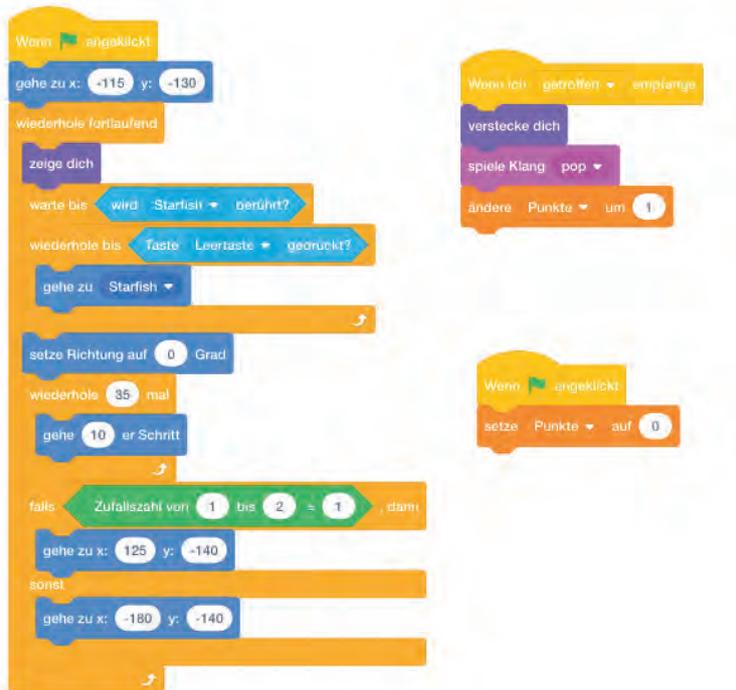
### Bühne



### Großer Seestern



## Kleiner Seestern



## Tintenfisch



# IT2School

Gemeinsam IT entdecken



## Modul B6 – Mein Anschluss

MocoMoco – Mein besonderer Anschluss

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Inhalt

1	Mein besonderer Anschluss .....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Rolle der Unternehmensvertreter*innen.....	4
5	Inhalte des Moduls.....	5
5.1	Anschlüsse des MocoMoco .....	5
5.2	Funktionsweise des MocoMoco.....	6
6	Unterrichtliche Umsetzung.....	6
6.1	Grober Unterrichtsplan.....	6
6.2	Stundenverlaufsskizzen .....	7
6.2.1	Erste Doppelstunde .....	7
6.2.2	Zweite bis etwa sechste Doppelstunde.....	8
7	Einbettung in verschiedene Fächer und Themen .....	10
8	Anschlussthemen.....	11
9	Literatur und Links .....	11
10	Arbeitsmaterialien .....	12
11	Glossar.....	12
12	FAQs und Feedback.....	13

# 1 Mein besonderer Anschluss

Was haben Bananen und Knete mit Tastaturen und Joysticks gemeinsam? Das können Schülerinnen und Schüler in diesem Modul erfahren. Mithilfe des mitgelieferten Mikrocontrollers können in Sekundenschnelle alle leitenden Alltagsgegenstände ganz ungefährlich als Eingabegeräte an den Computer angeschlossen werden – ob Bananen, Aluminiumfolie, Blumen oder sogar die Mitschülerinnen und Mitschüler selbst. Dadurch ergeben sich viele kreative Möglichkeiten, Eingabegeräte selbst zu gestalten und dafür auch Programme zu schreiben, die diese Eingaben zur Steuerung eines Spiels oder zur Kommunikation nutzen.



<b>Lernfeld/Cluster:</b>	IT spielend entdecken	
<b>Zielgruppe/Klassenstufe:</b>	<b>X</b>	4. bis 5. Klasse
	<b>X</b>	6. bis 7. Klasse
	<b>X</b>	8. bis 10. Klasse
	<b>X</b>	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	ca. 3 bis 6 Doppelstunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Funktionsweise von Eingabegeräten wie Tastatur, Maus, Joystick verstehen</li> <li>• <b>Eingabe-Verarbeitung-Ausgabe(=EVA)-Prinzip</b> kennenlernen</li> <li>• Automatisierte Prozesse im Alltag entdecken</li> <li>• Ein eigenes Programm entwerfen und programmieren</li> </ul>	
<b>Vorkenntnisse der Schüler*innen:</b>	<ul style="list-style-type: none"> <li>• Zur Entwicklung eines eigenen Projekts sind Kenntnisse in Scratch notwendig (vgl. B5, Leichter ProgrammierEinstieg)</li> <li>• Grundkenntnisse Elektrizitätslehre (Leitfähigkeit, Spannung etc.) sind hilfreich</li> </ul>	
<b>Vorkenntnisse der Lehrkraft:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Zur Entwicklung eines eigenen Projekts sind Kenntnisse in Scratch notwendig (vgl. B5, Leichter ProgrammierEinstieg)</li> </ul>	
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	<ul style="list-style-type: none"> <li>• Zur Entwicklung eines eigenen Projekts, sind Kenntnisse in Scratch notwendig (vgl. B5, Leichter ProgrammierEinstieg)</li> <li>• Das Material sollte gesichtet (speziell Präsentation der Programmieraufträge) und der Mikrocontroller ausprobiert werden</li> </ul>	
<b>Sonstige Voraussetzungen:</b>	<ul style="list-style-type: none"> <li>• Ein Desktop-Computer oder Laptop pro Mikrocontroller</li> </ul>	

## 2 Warum gibt es das Modul?

Dieses Modul bietet Informatik zum Anfassen und Selbermachen. Durch die beigefügten Mikrocontroller erhalten die Schülerinnen und Schüler einen kreativen Zugang zu den Grundlagen der Informatik. Das Prinzip **Eingabe-Verarbeitung-Ausgabe**, kurz EVA-Prinzip, wird anschaulich anhand alltagsnaher Physik erklärt. Auf diese Weise werden die technischen Vorgänge bei der Informationsverarbeitung des Computers erkennbar.

Darüber hinaus können die Schülerinnen und Schüler eigene Informatiksysteme in Form eines Programms nicht nur entwerfen, sondern auch selbst programmieren und mithilfe des Mikrocontroller bedienen. Das können zum Beispiel Musikinstrumente, Zeichenprogramme, Computerspiele oder interaktive Geschichten sein. Hierfür empfiehlt es sich, das Modul B5 entweder im Vorfeld durchzunehmen oder in das vorliegende Modul zu integrieren.

Zudem entdecken die Schülerinnen und Schüler die Mensch-Maschine-Interaktion als technologischen Schlüsselbereich und erleben Technologie als etwas Kreatives und Veränderbares.

## 3 Ziele des Moduls

Grundlegend können folgende Ziele verfolgt werden:

- Funktionsweise von Eingabegeräten wie Tastatur, Maus, Joystick verstehen
- Grundlagen der Informationsverarbeitung verstehen (EVA-Prinzip)
- automatisierte Prozesse im Alltag entdecken
- Sensor-Aktor-Prinzip verstehen
- gegebenenfalls ASCII-Codierung kennenlernen

Durch Einbindung einer Programmiersprache/-umgebung (z. B. Scratch mittels Modul B5) lassen sich auch folgende Ziele verfolgen:

- ein eigenes Programm unter festgelegten Parametern im Team planen und programmieren (Analyse, Entwurf, Implementierung, Test)
- verschiedene Operatoren, Anweisungen und Eigenschaften innerhalb der Programmierumgebung zielgerichtet nutzen
- logische Abläufe innerhalb der Programme erstellen und erklären
- eigene Ideen und Vorgehensweisen im Plenum vorstellen und verteidigen

## 4 Rolle der Unternehmensvertreter\*innen

Im *Modul B6 – Mein Anschluss* hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten aktiv mitzuwirken. Hier einige Anregungen:

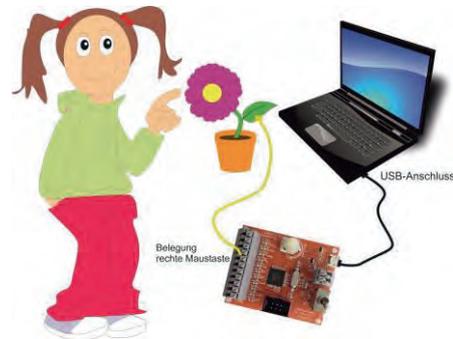
- Unterstützung der Lehrkraft, zum Beispiel bei der Einführung in die Scratch-Programmier-Oberfläche
- Auftraggeber\*in für die Programmieraufträge (B6.4)

- Unterstützung der Schülerinnen und Schüler bei der Umsetzung ihrer eigenen kreativen Projekte
- Gast(-Juror\*in) bei der Abschlusspräsentation der Projektergebnisse

## 5 Inhalte des Moduls

Über die üblichen Arbeitsblätter hinaus gehört zu diesem Modul der MocoMoco in mehrfacher Ausführung. Dabei handelt es sich um einen Mikrocontroller, der vom Makey Makey<sup>1</sup> inspiriert wurde.

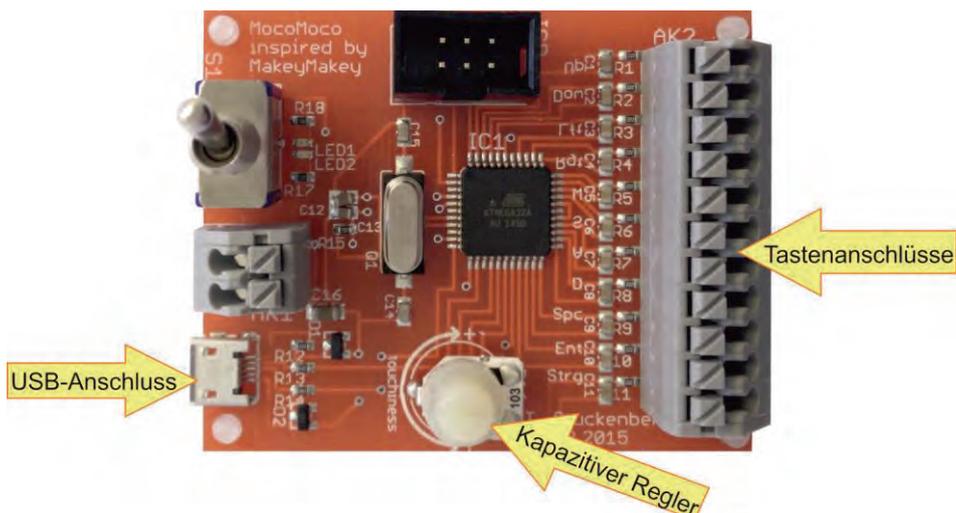
Für die Bedienung eines PC gibt es heute viele verschiedene Eingabegeräte, von der klassischen Tastatur über Joysticks bis zu Mundmäusen. Der Mikrocontroller MocoMoco macht es möglich, auch Alltagsgegenstände als Eingabegeräte zu nutzen. Er besteht aus einer Platine, die per USB an den Computer angeschlossen wird. Der Computer erkennt diese Platine als angeschlossene Tastatur. An die Platine können nun mit einem Draht verschiedene leitfähige Gegenstände wie Obst, Blumen oder Gefäße mit Wasser angeschlossen und zur Bedienung etwa von Computerspielen oder Musikprogrammen verwendet werden.



Mithilfe der Programmiersprache Scratch können die Schülerinnen und Schüler leicht eigene kleine Anwendungen oder Spiele entwickeln, die dann mit dem Mikrocontroller bedient werden können.

Dieses einfache Konzept eröffnet Kindern und Jugendlichen einen altersgerechten Zugang zur Informatik und lässt sie eigene spannende Projekte entwickeln.

### 5.1 Anschlüsse des MocoMoco



<sup>1</sup>Copyright JoyLabz LLC © 2012-2015. Makey Makey is a registered trademark of JoyLabz LLC. Dieser Controller kann alternativ zum MocoMoco mit diesem Modul verwendet werden. Es bestehen aber Unterschiede in der Funktionsweise: Beim Makey Makey muss für eine Signalübertragung ein Stromkreis zur Erde (Earth) geschlossen werden, beim MocoMoco ist dies nicht erforderlich, da dieser wie ein Touch-Display kapazitiv funktioniert.

- USB-Anschluss:** Über den USB-Anschluss wird der MocoMoco mit einem Computer verbunden, er funktioniert dann wie eine externe Tastatur.
- Kapazitiver Regler:** Mit diesem Regler kann die Empfindlichkeit des MocoMoco eingestellt werden.
- Tastenanschlüsse:** An den MocoMoco können bis zu acht Gegenstände angeschlossen werden. Dabei können folgende Tasten belegt werden: W, A, S, D sowie vier Pfeiltasten.

## 5.2 Funktionsweise des MocoMoco

Wird ein Gegenstand an den MocoMoco angeschlossen (abgesehen vom USB-Anschluss), dann nutzt der Mikrocontroller diesen als kapazitiven Sensor beziehungsweise Schalter. Technisch bedeutet dies, dass der angeschlossene Gegenstand zu einem Kondensator wird – also einem passiven, elektrischen Baustein mit der Fähigkeit, eine elektrische Ladung zu speichern. Änderungen der Kapazität – der gespeicherten Ladung – können vom MocoMoco erfasst werden. Das geschieht beispielsweise, wenn jemand seine Hand auf einen angeschlossenen (leitenden) Gegenstand legt - das generiert einen „Tastendruck“.

# 6 Unterrichtliche Umsetzung

Dieses Modul zeichnet sich durch die spielerische Einführung in die Thematik der Eingabegeräte und die historische Komponente aus.

Im ersten Teil befassen sich die Schülerinnen und Schüler mit der Frage, wie man einen Computer bedient und welche Eingabegeräte es gibt. Im Anschluss wird ihnen der Mikrocontroller vorgestellt, sie können ihn dann mit verschiedenen Materialien ausprobieren.

Im zweiten Teil erhalten die Schülerinnen und Schüler einen Programmierauftrag mit Scratch, den sie eigenverantwortlich in einer Gruppe planen, umsetzen und präsentieren. Entsprechend sollte im Vorfeld sichergestellt werden, dass die Schülerinnen und Schüler ausreichend Vorkenntnisse besitzen, also etwa das Modul B5 absolviert haben. Im Rahmen der Gruppenarbeit sollte jede Schülerin und jeder Schüler selbst aktiv werden und einen Beitrag zum Ergebnis leisten.

## 6.1 Grober Unterrichtsplan

Unterrichtsszenarien	Kurze Zusammenfassung der Unterrichtsstunde
Einstieg	Einführung des Controllers
Vertiefung	Vorstellung Programmieraufträge, Gruppeneinteilung
Erarbeitung	Ideenfindung, Erstellung einer Programmskizze
Erarbeitung	Umsetzung der Ideen in Scratch
Ergebnissicherung	Umsetzung/Präsentationsplanung
Abschluss	Präsentation der Projektergebnisse

## 6.2 Stundenverlaufsskizzen

### Abkürzungen/Legende

AB = Arbeitsblatt/Arbeitsblätter; L = Lehrkraft; MuM = Mitschüler\*innen; SuS = Schüler\*innen; UV = Unternehmensvertreter\*in

#### 6.2.1 Erste Doppelstunde

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Lehrkraftvortrag/- demonstration	Begrüßung der SuS und ggf. Vorstellung des UV  Demonstration des Controllers  L holt Freiwillige* n nach vorne und erklärt die Anschlüsse. Ein einfaches Spiel kann gespielt und mittels eines Beamers übertragen werden.  Für einen alternativen Einstieg: siehe B6.5	Beamer, Computer, Controller, leitende Materialien, PC-Spiel
35 Min.	Praxisphase	Gruppenarbeit	Im Anschluss daran probieren SuS die Controller selbst aus (ideal im 2er-Team). <b>Aufgabe:</b> Die Controller selbst anschließen und verschiedene Materialien ausprobieren.  Zur Sicherung kann AB B6.1 verwendet werden: Welche Materialien funktionieren, was haben sie gemeinsam?	Controller, leitende Materialien, Computer, B6.1
10 Min.	Sicherung	Plenum	Erfahrungsberichte, Besprechen der Antworten von B6.1	
10 Min.	Hinführung	Plenum	Fragen an SuS: <b>1. Frage:</b> Mit welchen Eingabegeräten kann man den Computer noch bedienen und wozu?  (z. B. Tastatur oder Sprachsteuerung zur Bedienung von Schreibprogrammen, Maus zur Bedienung der Benutzeroberfläche)	

			che, Gamepad/Joystick für Spiele) <b>2. Frage:</b> Woher weiß der Computer eigentlich, was man mit den Eingabegeräten macht – wie funktioniert das? (Tastatur besteht aus elektrischer Matrix aus Reihen und Spalten; wird eine Taste gedrückt, schließt sich an dieser Stelle der Stromkreis und die Information wird über einen Mikrocontroller ausgewertet und an den PC gesendet – siehe B6.2.)	
20–25 Min.	Erarbeitung	Einzel-/Gruppenarbeit	B6.2 wird ausgeteilt und gemeinsam gelesen. Im Anschluss daran sollen SuS daraus ein Schaubild entwickeln und vorstellen sowie dazugehörige Fragen beantworten.	B6.2
10 Min.	Sicherung	Plenum	Schaubild und Antworten werden präsentiert und verglichen.	

#### 6.2.2 Zweite bis etwa sechste Doppelstunde

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Materialien
10 Min.	Einstieg	Lehrkraftvortrag	L oder auch UV fungiert als Auftraggeber*in, der SuS einen Programmierauftrag erteilt. SuS entwickeln unterschiedliche Programme, die mithilfe des Controllers bedient werden können. Für die Ideenfindung kann auch auf die Methode Design Thinking (vgl. Modul B4) zurückgegriffen werden.	B6.3, Beamer, Computer
5 Min.	Gruppenfindung	Gruppenarbeit	SuS werden in vier Gruppen unterteilt; Gruppeneinteilung abhängig von Lerngruppe (Mädchen/Jungen getrennt, lernheterogen oder -homogen, ...).	B6.4
5 Min.	Vorbereitung	Gruppenarbeit	Die vier Programmieraufträge weisen unterschiedliche Schwierigkeitsgrade auf, wodurch eine Binnendifferenzierung möglich ist. Die Arbeitsaufträge werden an die Gruppen ausgeteilt. SuS machen sich damit vertraut, im Anschluss können Fragen geklärt werden.	B6.4

30 Min.	Vorbereitung	Gruppenarbeit	Die Aufgaben 1–4 von B6.4 werden abgearbeitet: <ul style="list-style-type: none"> <li>Brainstorming, Ideen sammeln, wie das Projekt umgesetzt werden kann und mit welchen Materialien</li> <li>Erstellung einer Projektskizze</li> </ul>	B6.4
5 Min.	Vorbereitung	Präsentation	Die Gruppen geben in wenigen Worten ihre Projektidee wieder; MuM sowie L können Feedback und ggf. Verbesserungsvorschläge geben.	
90–120 Min.	Gruppenphase/Sicherung	Gruppenarbeit	Im Anschluss daran wird die Idee (z. B. in Scratch) realisiert. Weitere Anweisungen für SuS finden sich in B6.4.	Computer, B6.4.
15 Min.	Vorbereitung Präsentation	Gruppenarbeit	SuS bereiten die Abschlusspräsentation mit folgenden Inhalten vor: <ul style="list-style-type: none"> <li>Idee</li> <li>Graphik</li> <li>Umsetzung/Quellcode</li> <li>Demonstration/Controller</li> </ul>	Computer, Power-Point, Controller, leitende Materialien
5–10 Min./Gruppe	Präsentation	Schüler*in-Vortrag Plenum	Präsentation vor der Klasse und dem*der Auftraggeber*in (UV) Feedback von MuM, L und UV	Computer, Beamer Power-Point, Controller, leitende Materialien.

## 7 Einbettung in verschiedene Fächer und Themen

Dieses Modul weist Bezüge zu verschiedenen Fächern auf. In der Physik können der Stromkreis sowie elektrische Leiter und Nichtleiter thematisiert werden. Für das Fach Kunst bietet sich die Erstellung eines Zeichenprogramms oder einer Computeranimation an. Im Musikunterricht kann der Arbeitsauftrag für das Musikinstrumente-Programm verwendet und im Deutschunterricht eine interaktive Geschichte programmiert werden.

Durch diese Bezüge kann das gesamte Modul oder können einzelne Teile in verschiedenen Fächern eingesetzt werden. Die folgenden Kompetenzen finden sich entweder in den Bildungsstandards der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

### Deutsch

Die Schülerinnen und Schüler...

- können verschiedene Medien für Präsentationen nutzen (Grundschule).
- können Texte (medial unterschiedlich vermittelt) szenisch gestalten (Sek I).
- können Texte mit Hilfe von neuen Medien verfassen (Sek I).

### Physik

Die Schülerinnen und Schüler ...

- können nach Schaltskizzen Stromkreise aufbauen.
- kennen elektrische Leiter und Nichtleiter.
- beschreiben Phänomene und führen sie auf bekannte physikalische Zusammenhänge zurück.
- planen einfache Experimente, führen sie durch und dokumentieren die Ergebnisse.

### Musik

Die Schülerinnen und Schüler ...

- entwerfen unter einer leitenden Idee klangliche Gestaltungen auf der Grundlage ausgewählter Ordnungssysteme musikalischer Parameter und Formaspekte.
- realisieren einfache vokale und instrumentale Kompositionen und eigene klangliche Gestaltungen auch unter der Verwendung digitaler Werkzeuge und Medien.
- experimentieren mit Tönen, Klängen und Geräuschen.

### Kunst

Die Schülerinnen und Schüler ...

- setzen digitale Medien für eine Gestaltungsaufgabe ein.
- setzen designorientierte Findungsprozesse und Lösungsstrategien ein.

- entwickeln verschiedene Ideen in einem Entwurfsprozess und stellen den Entwurf angemessen dar.
- realisieren filmische Projekte (Stopmotion, Video, Computeranimationen).

## Informatik

Die Schülerinnen und Schüler...

- beschreiben die Funktionsweise von Informatiksystemen als Zusammenspiel von Hard- und Softwarekomponenten.
- erstellen Produkte unter Anwendung fortgeschrittener Techniken von Standardsoftware; falls es das Produkt erfordert, arbeiten sie sich in geringem Umfang in Spezialsoftware ein.
- erwerben beim Bearbeiten von Softwareprojekten in angemessenem Umfang Kenntnisse über Analyse- und Modellierungsverfahren sowie Projektmanagement.
- verstehen Programmabläufe und die Arbeitsweise von Schnittstellen.
- können erdachte Systeme in technische übertragen.
- kennen sich in Programmierumgebungen aus.

## 8 Anschluss Themen

Als Anschluss Themen im Zusammenhang mit IT2School bieten sich folgende Bausteine an:

### Beispiel: Programmieren

Das Modul B6 *Mein Anschluss* wird in der Regel zusammen mit B5 *Programmieren* eingesetzt, daher bietet es sich an, im nächsten Modul das Thema der Programmierung zu vertiefen:



## 9 Literatur und Links

- **Offizielle Seite der Makey Makeys.** Online: <http://www.makeymakey.com/>
- **Bauanleitungen für MakeyMakeys:**  
Online: <https://epic-stuff.de/bauanleitungen/bauanleitungen-makey-makey/>
- Hielscher, Michael & Döbeli Honegger, Beat (2015): **MaKey MaKey Projektideen.** Vielfältige Ideen für den Einsatz im Unterricht. Online: <http://ilearnit.ch/download/MakeyMakeyProjektideen.pdf>
- **Scratch-Karten.** Online: <https://scratch.mit.edu/help/cards>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B6.1	Der Mikrocontroller	Arbeitsblatt, das SuS mit dem Mikrocontroller vertraut macht. Sie prüfen, welche Materialien mit dem MocoMoco kompatibel sind.
 B6.2	Mensch-Maschine-Schnittstelle	Arbeitsblatt zu historischen Eckdaten von Eingabegeräten und Mensch-Maschine-Schnittstellen.
 B6.3	Powerpoint	Powerpoint kann zur Vorbereitung/als Einstieg auf B6.4 Programmierauftrag dienen.
 B6.4	Programmierauftrag	Arbeitsblätter umfassen 4 verschiedene Programmieraufträge: <ul style="list-style-type: none"> <li>• interaktive Geschichte</li> <li>• Musikprogramm</li> <li>• Spiel</li> <li>• Zeichenprogramm</li> </ul>
 B6.5	E-Mail	Vorlage dient als alternativer Einstieg für dieses Modul.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Mikrocontroller (kurz $\mu\text{C}$ )	Halbleiterchips, die mit mindestens einem Prozessor, Arbeits- und Programmspeicher ausgestattet sind. Zusätzlich verfügt der Mikrocontroller über programmierbaren Ein- und Ausgabe Schnittstellen.
EVA-Prinzip	Eingabe – Verarbeitung – Ausgabe, Grundprinzip der Datenverarbeitung.
Scratch	Visuelle Programmierumgebung für Anfänger, entwickelt von der Lifelong Kindergarten Group des Media Lab am MIT.

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



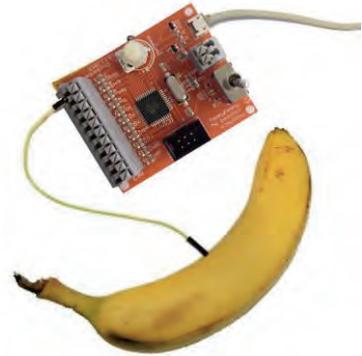
<https://tinyurl.com/IT2S-FAQ>

Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Der Mikrocontroller



## Aufgaben

- Überprüft, mit welchen Materialien ihr die Computer steuern könnt. Erstellt in der Tabelle eine Liste der Dinge, mit denen ihr den Mikrocontroller getestet habt und kennzeichnet, welche funktioniert haben und welche nicht (siehe Beispiel).

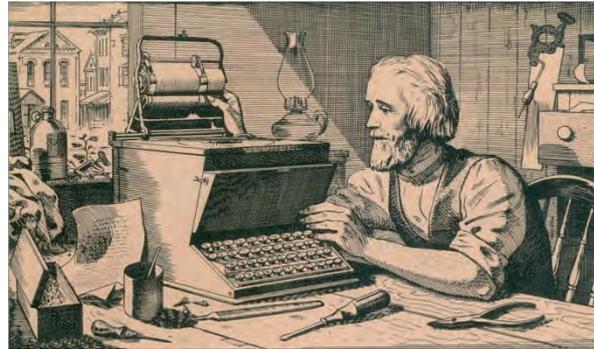
Material	funktioniert	funktioniert nicht
Banane	X	

- Vergleicht alle Materialien miteinander und beschreibt, was sie gemeinsam haben und was sie unterscheidet. Leitet daraus eine Regel ab: Was müssen Dinge erfüllen, damit sie mit dem MocoMoco benutzt werden können?

# Mensch-Maschine-Schnittstellen

## Die Tastatur

Die Tastatur wird vor allem benötigt, um zu „schreiben“. Als Eingabegerät sendet sie Signale an den Computer. Dieser wandelt sie in Zeichen um (Buchstaben, Zahlen, Symbole...), die z.B. danach auf dem Monitor zu sehen sind. Möglich wird dies durch ein der sichtbaren Tastatur unterlegtes Gitter, das aus Reihen und Spalten besteht, in dem jedes Feld



Christopher Latham Sholes, Artwork

eine spezifische Taste darstellt. Bei jedem Tastendruck wird eine elektrische Verbindung zwischen einer Reihe und Spalte hergestellt. Das so entstandene Signal wird an eine Steuereinheit gesendet, diese übergibt einen Code an den Computer und das passende Zeichen wird ausgeführt. Die Tastenbelegung „QWERTY“, die sich auf die Belegung der ersten sechs Buchstabentasten bezieht, wurde 1868 von Christopher Latham Sholes entwickelt. Er wollte verhindern, dass die Metallstifte der damals benutzten Schreibmaschinen sich beim Schreiben überkreuzen und verklemmen. Deswegen verteilte er die Buchstaben, die in der englischen Sprache am häufigsten ein Paar bilden, auf gegenüberliegende Seiten der Tastatur.

## Aufgaben

1. Vergleicht die Tastenbelegung bei euren Computern mit der im Text beschriebenen. Sieht die Tastatur anders aus? Woran könnte das liegen?
2. Erläutert für welchen Zweck die erste Tastatur entwickelt wurde und bezieht euch dabei auf den Text.

## Die Maus

Mit der Maus kann man einen Zeiger auf dem Monitor bewegen und mithilfe ihrer Tasten verschiedene Aktionen ausführen. Ein einfacher Linksklick markiert das ausgewählte Symbol, ein Doppelklick öffnet einen ausgewählten Ordner oder startet ein Programm. Mit der rechten Maustaste lassen sich zusätzliche Optionen anzeigen. Die erste Maus wurde 1963 von Douglas C. Engelbart und William English gebaut. Sie bestand aus Holz und besaß nur eine Taste. Bei ihrer offiziellen Vorstellung 1968 fand sie noch wenig Beachtung, da die damaligen Computer noch keine geeignete graphische Oberfläche besaßen. Das änderte sich 1984, als Apple die grafische Oberfläche einführte. Damit wurde die Maus zu einem unverzichtbaren Werkzeug, das bis heute stetig weiterentwickelt und verbessert wird.

## Aufgaben

1. Beschreibt wie die Mäuse heutzutage aussehen. Recherchiert, welche gängigen Technologien es gibt und wie diese funktionieren.
2. Fertigt einen Zahlenstrahl an, in den ihr die Daten aus dem Text einzeichnet.

Abbildung Christopher Latham Sholes - Artwork. Quelle: (Public Domain) [https://commons.wikimedia.org/wiki/File:Christopher\\_Latham\\_Sholes\\_artwork.png](https://commons.wikimedia.org/wiki/File:Christopher_Latham_Sholes_artwork.png) [17.11.2015]



IT2SCHOOL – GEMEINSAM IT ENTDECKEN

# PROGRAMMIERAUFTRAG

# VORGESCHICHTE

Unsere Firma benötigt neue Programme für die Pausengestaltung der Beschäftigten. Eine Pause dauert max. 15 Minuten.

Eine Umfrage ergab, dass diese möglichst kurzlebig und mit einem Mikrocontroller steuerbar sein sollten.

**Jetzt seid ihr die Softwareentwickler\*innen!**

# AUFTRÄGE

## 4 verschiedene Aufträge

Es sind vier verschiedene Programme gewünscht:

- ein Spiel
- eine interaktive Geschichte
- ein Musikprogramm
- ein Malprogramm

**Alle Programme sollen mit Scratch programmiert und mit einem Mikrocontroller bedient werden.**

# PHASEN

1. Gruppeneinteilung
  - pro Arbeitsauftrag eine Gruppe
2. Auftrag erhalten und besprechen
3. Brainstorming, Ideenfindung
  - Programmentwurf und Materialverwendung
4. Umsetzung mit Scratch und Mikrocontroller
5. Präsentation der Ergebnisse

# Programmierauftrag: Zeichnen

Einige Mitarbeiter\*innen unserer Firma haben sich ein witziges Zeichenprogramm gewünscht, mit dem sie in ihren Pausen etwas herumkritzeln können. Es sollte auf jeden Fall schön bunt und kreativ sein. Wichtig ist außerdem, dass es im Aufbau nicht zu kompliziert ist und man es leicht verstehen kann. Steuern wollen sie es mit dem Moco Moco.

## Aufgaben

1. Sammelt eure Ideen zu dem Programm schriftlich. Orientiert euch dabei an den Fragen:

- Was soll erreicht werden?
- Wie können wir es umsetzen?
- Womit soll unser Programm gesteuert werden?

Weitere Anregungen findet ihr auf den Scratchcards (von der Lehrkraft) oder auf:

<http://scratch.mit.edu/> und <http://www.epic-stuff.de/produkte/makey-makey/>

2. Stellt eure Ideen eurer Lehrkraft vor und notiert euch Verbesserungsvorschläge oder Fragen.
3. Formuliert in euren eigenen Worten den Ablauf eures Programms oder erstellt eine Skizze dazu.
4. Erstellt auf eurem Rechner bzw. im Schulnetzwerk einen gemeinsamen Ordner/Speicherort für eure Arbeitsgruppe und setzt euren Programmablauf in Scratch um. Vergesst nicht zwischendurch zu speichern!
5. Wenn euer Programm funktioniert, erstellt bitte eine Präsentation, in der ihr euer Programm vorstellt. Folgende Fragen helfen euch weiter:
  - Worum geht es in eurem Programm?
  - Warum sieht es so aus?
  - Welche Steuerung habt ihr gewählt und warum?
  - Was passiert im Quellcode?

# Programmierauftrag: Spiel

Einige Mitarbeiter\*innen unserer Firma haben sich ein Spiel gewünscht, mit dem sie sich in ihren Pausen die Zeit vertreiben können. Es sollte auf jeden Fall ein Spiel sein, bei dem man gewinnen oder verlieren kann. Vielleicht ist es auch möglich gegeneinander zu spielen. Wichtig ist außerdem, dass es leicht verständlich ist. Steuern wollen sie es mit dem Moco Moco.

## Aufgaben

1. Sammelt eure Ideen zu dem Programm schriftlich. Orientiert euch dabei an den Fragen:

- Was soll erreicht werden?
- Wie können wir es umsetzen?
- Womit soll unser Programm gesteuert werden?

Weitere Anregungen findet ihr auf den Scratchcards (von der Lehrkraft) oder auf:

<http://scratch.mit.edu/> und <http://www.epic-stuff.de/produkte/makey-makey/>

2. Stellt eure Ideen eurer Lehrkraft vor und notiert euch Verbesserungsvorschläge oder Fragen.
3. Formuliert in euren eigenen Worten den Ablauf eures Programms oder erstellt eine Skizze dazu.
4. Erstellt auf eurem Rechner bzw. im Schulnetzwerk einen gemeinsamen Ordner/Speicherort für eure Arbeitsgruppe und setzt euren Programmablauf in Scratch um. Vergesst nicht zwischendurch zu speichern!
5. Wenn euer Programm funktioniert, erstellt bitte eine Präsentation, in der ihr euer Programm vorstellt. Folgende Fragen helfen euch weiter:
  - Worum geht es in eurem Programm?
  - Warum sieht es so aus?
  - Welche Steuerung habt ihr gewählt und warum?
  - Was passiert im Quellcode?

# Programmierauftrag: Musikprogramm

Einige Mitarbeiter\*innen unserer Firma möchten gerne etwas Musik machen in ihrer Pause. Es sollten auf jeden Fall mehrere Instrumente zur Auswahl stehen. Wichtig ist außerdem, dass das Programm nicht zu kompliziert im Aufbau ist und man es leicht verstehen kann. Steuern wollen sie es mit dem Moco Moco.

## Aufgaben

1. Sammelt eure Ideen zu dem Programm schriftlich. Orientiert euch dabei an den Fragen:

- Was soll erreicht werden?
- Wie können wir es umsetzen?
- Womit soll unser Programm gesteuert werden?

Weitere Anregungen findet ihr auf den Scratchcards (von der Lehrkraft) oder auf:

<http://scratch.mit.edu/> und <http://www.epic-stuff.de/produkte/makey-makey/>

2. Stellt eure Ideen eurer Lehrkraft vor und notiert euch Verbesserungsvorschläge oder Fragen.
3. Formuliert in euren eigenen Worten den Ablauf eures Programms oder erstellt eine Skizze dazu.
4. Erstellt auf eurem Rechner bzw. im Schulnetzwerk einen gemeinsamen Ordner/Speicherort für eure Arbeitsgruppe und setzt euren Programmablauf in Scratch um. Vergesst nicht zwischendurch zu speichern!
5. Wenn euer Programm funktioniert, erstellt bitte eine Präsentation, in der ihr euer Programm vorstellt. Folgende Fragen helfen euch weiter:
  - Worum geht es in eurem Programm?
  - Warum sieht es so aus?
  - Welche Steuerung habt ihr gewählt und warum?
  - Was passiert im Quellcode?

# Programmierauftrag: Geschichte

Einige Mitarbeiter\*innen unserer Firma möchten sich in ihrer Pause eine Geschichte ansehen. Sie sollte auf jeden Fall interaktiv sein und spannend oder lustig. Wichtig ist außerdem, dass sie nicht zu kompliziert im Aufbau ist und man sie leicht verstehen kann. Steuern wollen sie die Geschichte mit dem Moco Moco.

## Aufgaben

1. Sammelt eure Ideen zu dem Programm schriftlich. Orientiert euch dabei an den Fragen:

- Was soll erreicht werden?
- Wie können wir es umsetzen?
- Womit soll unser Programm gesteuert werden?

Weitere Anregungen findet ihr auf den Scratchcards (von der Lehrkraft) oder auf:

<http://scratch.mit.edu/> und <http://www.epic-stuff.de/produkte/makey-makey/>

2. Stellt eure Ideen eurer Lehrkraft vor und notiert euch Verbesserungsvorschläge oder Fragen.
3. Formuliert in euren eigenen Worten den Ablauf eures Programms oder erstellt eine Skizze dazu.
4. Erstellt auf eurem Rechner bzw. im Schulnetzwerk einen gemeinsamen Ordner/Speicherort für eure Arbeitsgruppe und setzt euren Programmablauf in Scratch um. Vergesst nicht zwischendurch zu speichern!
5. Wenn euer Programm funktioniert, erstellt bitte eine Präsentation, in der ihr euer Programm vorstellt. Folgende Fragen helfen euch weiter:
  - Worum geht es in eurem Programm?
  - Warum sieht es so aus?
  - Welche Steuerung habt ihr gewählt und warum?
  - Was passiert im Quellcode?

**Von:** Maxi Musterfrau

**An:** Max Mustermann

**Betreff:** Einladung Vollversammlung

Sehr geehrter Kollege,

hiermit laden wir Sie zur jährlichen Lehrervollversammlung im Festsaal ein. Bis jetzt haben fünf Gastredner\*innen aus aller Welt zugesagt, gerne nehmen wir noch Bewerbungen entgegen.

Die Veranstaltung wird Freitagabend beginnen und Samstagmittag mit einem Essen ausklingen.

Sollte es Ihnen möglich sein, an dieser Veranstaltung teilzunehmen, so antworten Sie auf diese E-Mail bitte mit „Ja“. Wir werden Ihnen daraufhin alle weiteren Informationen zusenden.

Sollten Sie verhindert sein, so antworten Sie bitte mit „Nein“.

Mit freundlichen Grüßen  
Ihr Festkomitee

## Arbeitsblatt B6.5

Erläuterung: Dieses Material kann für einen alternativen Einstieg/Unterrichtsverlauf genutzt werden:

Zeit	Phase	Sozialform/ Lehrer*in-Impuls	Inhalt/Unterrichtsgeschehen	Materialien
10 Min.	Einstieg	Lehrer*in-Vortrag	L erzählt, er hätte einen elektronischen Brief (E-Mail) bekommen, den er beantworten will. Seine Antwort schreibt L an die Tafel und fragt SuS, wie es möglich sei, diese in den Computer zu übertragen. Vorschläge an der Tafel sammeln, z. B.: Tastatur-Schreibprogramm, Maus-Benutzeroberfläche, Gamepad/Joystick-Spiele).	B6.5, Beamer, Computer, Tafel.
5 Min.	Gruppenfindung	Gruppendiskussion	Frage: Woher weiß der Computer eigentlich, was man mit den Eingabegeräten macht? Sammeln an der Tafel (Wissensstand der Klasse; möglicher Abstraktionsgrad).	Tafel
5 Min.	Vorbereitung	Gruppenarbeit	Text über Eingabegeräte, SuS sollen daraus ein Schaubild entwickeln und vorstellen/Fragen beantworten.	B6.2
5 Min.	Vorbereitung	Plenum	Frage: Kann man Computer auch mit einer Banane bedienen, wenn ja, wie? Abstimmen: Ja/Nein, Striche an der Tafel	Tafel, Banane.
15 Min.	Vorbereitung	Lehrer*in-Präsentation	Demonstration des Moco Moco, Lehrkraft kann Freiwilligen nach vorne holen. Antwort auf vorangegangene Frage (ja/nein) mit Obsttastatur eingeben.	Beamer, Computer, Bananen, angeschlossener Moco Moco.
45 Min.	Gruppenphase/ Sicherung	Gruppenarbeit	Ideal: 2er-Teams. SuS schließen Moco Mocos an (Anleitung), probieren Material aus und spielen. Sicherung: B6.1 Frage: Welche Materialien funktionieren, was haben sie gemeinsam?	Computer, Moco Mocos, verschiedenen Materialien (leitend und nicht leitend), B6.1.
15 Min.	Vorbereitung Präsentation	Plenum	Erfahrungsberichte, Besprechen der Fragen vom AB	Tafel

## Weiterführende Aufgabe

Die Lehrkraft zeigt den Schülerinnen und Schülern einen Moco Moco-Aufbau, der jedoch nicht funktioniert. Frage an die Schülerinnen und Schüler: Woran könnte das liegen? Thesen per Mindmap an der Tafel sammeln und Schülerinnen und Schüler nach vorne kommen lassen, um ihre Ideen zu überprüfen, nachdem sie ihre Vorgehensweise beschrieben haben (Anschlüsse nicht korrekt, falsches, nicht leitendes Material etc.).

Die E-Mail-Vorlage (Seite 1) kann individualisiert werden.

## Musterlösungen

### B6.1

Alle Materialien, die auch Strom leiten, funktionieren mit dem MocoMoco.

### B6.2

#### 1.1 Die Tastatur – Aufgabe 1

Die Tastaturbelegung ist im Deutschen nicht QWERTY, sondern QWERTZ. Das liegt daran, dass die Häufigkeitsverteilung der Buchstaben im Deutschen eine andere ist als im Englischen (Vergleich zur Tafel aus dem Modul B1.).

#### 1.2 Die Tastatur – Aufgabe 2

„[...] Er wollte verhindern, dass die Metallstifte der damals benutzten Schreibmaschinen sich beim Schreiben überkreuzen und verklemmen. Deswegen verteilte er die Buchstaben, die in der englischen Sprache am häufigsten ein Paar bilden, auf gegenüberliegende Seiten der Tastatur.“

#### 1.3 Die Maus – Aufgabe 3

1963	1968	1981	1984	1985	1991
wurde die erste Maus von Douglas C. Engelbart und William English gebaut.	wurde diese Maus vorgestellt, fand jedoch keine große Beachtung.	Wurde die erste kommerzielle Grafische Oberfläche von Xerox erarbeitet. Diese wurde allerdings nicht populär.	führte Apple die erste populäre grafische Oberfläche ein.	wurden die ersten Mäuse mit drei Tasten vorgestellt.	erschieden die ersten kabellosen Mäuse.

# IT2School

Gemeinsam IT entdecken



## Modul B7 – Meine App

App Inventor

Eine Entwicklung von



In Kooperation mit



Im Auftrag der



# Inhalt

1	App Inventor.....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Rolle der Unternehmensvertreter*innen.....	4
5	Inhalte des Moduls.....	5
5.1	Aufruf und Einstieg in den App Inventor .....	5
5.2	Die grafische Benutzeroberfläche.....	5
5.2.1	Entwicklungsumgebung Design Editor.....	6
5.2.2	Entwicklungsumgebung Blocks Editor .....	7
6	Unterrichtliche Umsetzung.....	7
6.1	Grober Unterrichtsplan.....	8
6.2	Stundenverlaufsskizzen .....	8
6.2.1	Einführung, Kennenlernen und Entwicklung einer App .....	8
6.2.2	Vertiefung: Eigene Projekte.....	8
7	Einbettung in verschiedene Fächer und Themen .....	8
8	Anschlussthemen.....	9
9	Literatur und Links .....	9
10	Arbeitsmaterialien .....	10
11	Glossar.....	10
12	FAQs und Feedback.....	12

# 1 App Inventor

Dieses Modul dient der Vertiefung im Bereich der Programmierung und Anwendungsentwicklung auf Android-Smartphones. Dabei bietet es viel Spielraum für die Kreativität der Schülerinnen und Schüler, aber auch der Lehrkraft selbst. Als Werkzeug für die Entwicklung von Applikationen (kurz: Apps) auf dem Smartphone dient der ursprünglich von Google entwickelte *App Inventor*, eine bausteinartige Entwicklungsumgebung, die sich ähnlich wie Scratch verhält.



Die Verbreitung und Nutzung von Smartphones nimmt unter Schülerinnen und Schülern (auch in der Sek. I) seit Jahren stark zu. Die Entwicklung eigener Apps ist für die Schülerinnen und Schüler daher ein naheliegendes Vorhaben und bedarf keiner gesonderten Motivation. Die Entwicklung selbst ist nicht nur Spielerei, sondern kann auch für den Unterricht und die Lehrkraft von Interesse sein. In verschiedenen Fächern lassen sich Themenfelder finden, für die das Smartphone als Werkzeug programmiert und genutzt werden kann.

<b>Lernfeld/Cluster:</b>	IT spielend entdecken	
<b>Zielgruppe/Klassenstufe:</b>		4. bis 5. Klasse
	X	6. bis 7. Klasse
	X	8. bis 10. Klasse
	X	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	4 bis 6 Einzelstunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Ein grundlegendes Programmierverständnis entwickeln</li> <li>• Eigenschaften von Programmen beschreiben</li> <li>• Erleben wie leicht kleine Programme selbst gestaltet werden können</li> <li>• Algorithmische Grundbausteine zur Programmierung verwenden</li> <li>• Programme entwerfen, implementieren und testen</li> <li>• Probleme selbstständig lösen</li> </ul>	
<b>Vorkenntnisse der Schüler*innen:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Erfahrung mit Scratch oder anderen Programmiersprachen</li> </ul>	
<b>Vorkenntnisse der Lehrkraft:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Erfahrung in der Programmierung</li> </ul>	
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Erfahrung in der Programmierung</li> </ul>	
<b>Sonstige Voraussetzungen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• Genügend Computer mit einer stabilen Internetverbindung (App Inventor kann nicht offline ausgeführt werden)</li> <li>• Ein anderer Browser als der Internet Explorer (da dieser vom App Inventor nicht unterstützt wird)</li> </ul>	



## 2 Warum gibt es das Modul?

Ähnlich wie in den Modulen B4 und B6 wird in diesem Modul den Schülerinnen und Schülern Raum für eigenständiges Entwickeln und Austesten gegeben. Der Fokus liegt dabei auf einer Vertiefung der Programmierfähigkeiten, zugleich geht es darum, den Umgang mit anderen Informatiksystemen als dem Computer kennenzulernen.

Die Schülerinnen und Schüler werden für die Entwicklung von Anwendungen begeistert und erfahren, welche Einsatzmöglichkeiten das Smartphones (noch) bietet. Dabei entwickeln sie ein Verständnis für die Grundlagen der App-Entwicklung und können gezielt Problemlösungen und eigene Projekte angehen. In der aktiven Auseinandersetzung und Entwicklung eigener Projekte erfahren die Schülerinnen und Schüler mehr über moderne Arbeitsprozesse und lernen, dass es nicht zuletzt an ihnen liegt, die digitale Gesellschaft mitzugestalten.

Auch wenn die Heranführung an die Programmierung im Vordergrund steht, kann das Modul im zweiten Schritt sehr gut dazu verwendet werden, IT-Werkzeuge für Unterrichtsfächer außerhalb der Informatik zu entwickeln. So können die Schülerinnen und Schüler zum Beispiel eine Vokabel-App für den Fremdsprachenunterricht entwickeln.

## 3 Ziele des Moduls

- Ein grundlegendes Programmierverständnis entwickeln
- Erleben, wie leicht Apps für Android-Smartphones gestaltet werden können
- Softwareerstellung als kreativen Prozess entdecken
- Eigenschaften von Programmen analysieren und beschreiben
- Algorithmische Grundbausteine zur Programmierung verwenden
- Entwerfen, Implementieren und Testen eigener Programme

## 4 Rolle der Unternehmensvertreter\*innen

Im Modul B7 – Meine App hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten, aktiv mitzuwirken. Hier einige Anregungen:

- Unterstützung der Lehrkraft beispielsweise bei der Einführung in die Programmier-Oberfläche des App Inventors
- Unterstützung der Schülerinnen und Schüler bei der Umsetzung ihrer eigenen kreativen Projekte
- Sie oder er kann zur Abschlusspräsentation der Projektergebnisse als Special-Guest eingeladen werden.

## 5 Inhalte des Moduls

Zur Entwicklung von Anwendungen (engl.: *applications*, kurz: Apps) wird in diesem Modul das Programm App Inventor verwendet. Es handelt sich dabei um eine Web-Anwendung, die nur über den Browser und mit bestehender Internetverbindung aufgerufen werden kann. App Inventor wurde von Google entwickelt und ermöglicht es, Anwendungen für das Android und iOS System zu erstellen.

Die Bedienung des Programms ist vergleichbar mit Scratch – so wird für die Programmierung ebenfalls eine grafische Benutzeroberfläche genutzt, auf der per Drag-and-Drop Bausteine zu einem Skript kombiniert werden. Um die selbstgeschriebenen Apps zu testen, ist kein Smartphone notwendig, da auf einen Emulator zurückgegriffen werden kann. Aber natürlich lassen sich die Apps auch auf dem Smartphone installieren. Eine wichtige Voraussetzung für die Nutzung des Programms ist neben der ständigen Internetverbindung ein Google-Nutzerkonto (über das Nutzer eines Android-Smartphones normalerweise verfügen oder schnell erstellt werden kann). Für den Unterricht kann die Lehrkraft auch selbstständig einige Nutzerkonten anlegen.

### 5.1 Aufruf und Einstieg in den App Inventor

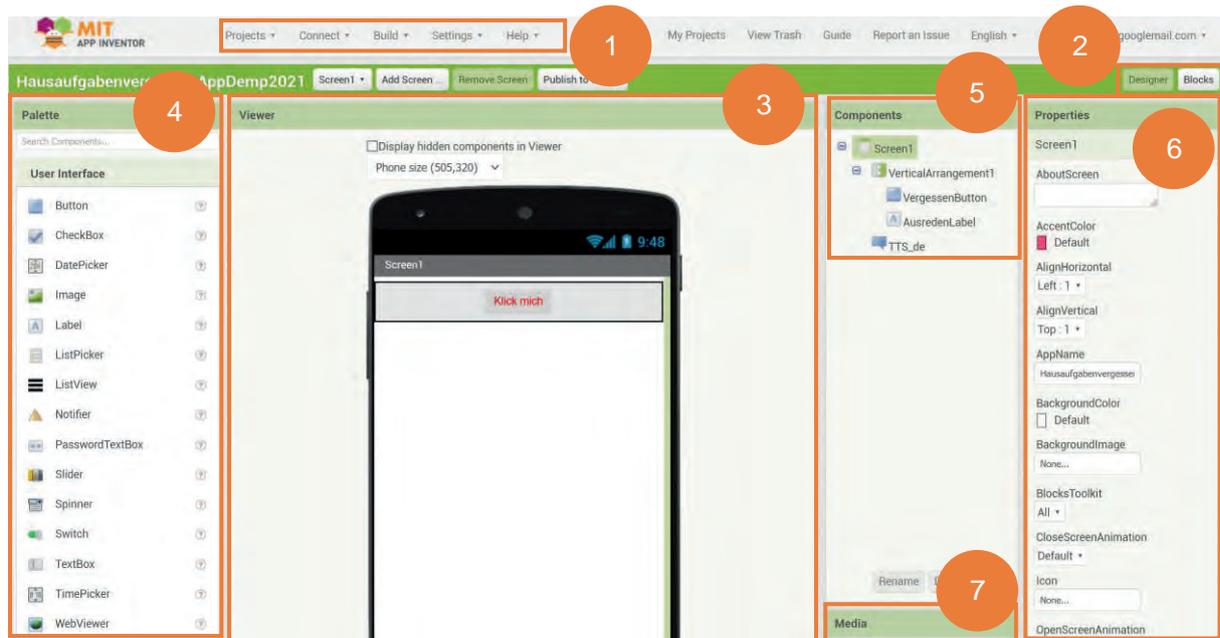
Die Webseite des App Inventors kann unter <https://appinventor.mit.edu> aufgerufen werden. Nach einem Klick auf **Create Apps!** (siehe oben links auf der Webseite), wird man zum Login mit einem Google-Account aufgefordert. Nach dem erfolgreichen Login ist eine Übersicht der Projekte zu sehen. Nach Anlegen eines neuen Projekts oder Anklicken eines bestehenden Projekts öffnet sich die Entwicklungsumgebung des App Inventors.

Bevor jedoch mit der eigentlichen App-Entwicklung begonnen werden kann, muss für das Testen der Apps noch ein Emulator-Programm oder eine App zur „Live-Vorschau“ (MIT AI2 Companion für Android oder MIT App Inventor für iOS) installiert werden. Nähere Informationen zu den Apps und dem Emulator finden sich hier: <https://appinventor.mit.edu/explore/ai2/setup.html>

### 5.2 Die grafische Benutzeroberfläche

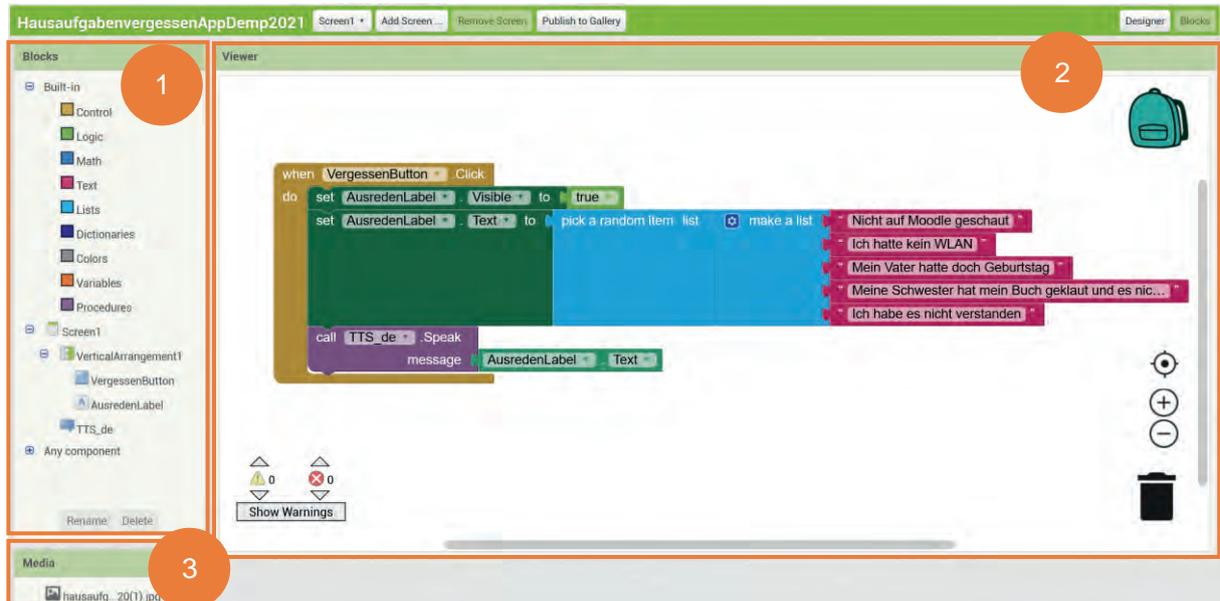
Der App Inventor verfügt über zwei Editoren. Im *Design Editor* werden die einzelnen App-Komponenten (Buttons, Texte, Bilder, Kamera, verschiedene Sensoren etc.) hinzugefügt, gelöscht, bearbeitet und angeordnet. Im *Blocks Editor* erfolgt die Programmierung der zuvor eingefügten Komponenten.

## 5.2.1 Entwicklungsumgebung Design Editor



1. In diesem Bereich sind allgemeine Optionen zum Projekt zu finden. Es kann gespeichert oder für die Installation auf dem Smartphone exportiert werden. Unter *Connect* sind die verschiedenen Möglichkeiten aufgelistet, um die App zu testen. Sie kann entweder mithilfe eines eigenen Programms auf dem Computer emuliert werden oder auf dem Smartphone über die App in einer Vorschau-Version aufgerufen werden.
2. Hier kann zwischen den beiden Editoren gewechselt werden.
3. Der *Viewer*-Bereich zeigt, wie die App grafisch aufgebaut ist. Sichtbare Komponenten wie Buttons, Texte oder Bilder sind hier dargestellt. Somit ist annähernd das zu sehen, was auch die Nutzer der fertigen App auf dem Smartphone angezeigt bekommen.
4. Innerhalb der in diesem Bereich aufgelisteten Kategorien befinden sich die einzelnen Komponenten, die zu der App hinzugefügt werden können, unter anderem:
  - *User Interface* enthält Komponenten, die zur Ein- und Ausgabe sowie zur Interaktion mit dem Nutzer dienen. Enthalten sind beispielsweise *Button*, *CheckBox*, *DatePicker* (Auswahlmöglichkeit für das Datum), *ListPicker* (Auswahllisten) etc.
  - *Sensors* enthält Komponenten, die Sensoren des Smartphones darstellen. Enthalten sind beispielsweise *AccelerometerSensor* (Geschwindigkeitssensor), *BarcodeScanner*, *LocationsSensor* (Positionssensor) etc.
5. Im Bereich *Components* sind alle sichtbaren und nicht sichtbaren Komponenten der App aufgelistet. Hier können auch Bezeichnungen geändert oder Komponenten gelöscht werden.
6. Sobald in Bereich *Viewer* oder *Components* eine Komponente ausgewählt wurde, können unter *Properties* Eigenschaften dieser Komponente verändert werden. So lässt sich beispielsweise die Höhe oder Breite sowie die Schriftfarbe eines Buttons oder Labels bestimmen.
7. Hier können eigene Medien wie Bilder und Musik hochgeladen werden, um sie dann in der App (als Hintergrundbild oder für die Soundausgabe) zu nutzen.

## 5.2.2 Entwicklungsumgebung Blocks Editor



1. Der Bereich *Blocks* beinhaltet Bausteine zur Programmierung der App. Sie sind aufrufbar, indem man mit dem Mauszeiger über eine Kategorie (z. B. *Control*, *Logic*) oder Komponenten (z. B. *Screen1*) fährt.
2. In den *Viewer*-Bereich müssen die einzelnen Bausteine per Drag-and-Drop gezogen und hier verbunden werden. Ein Rechtsklick ermöglicht das schnelle Duplizieren von Bausteinen (was ein erneutes Hineinziehen deutlich verkürzt).
3. In *Media* können eigene Medien hochgeladen werden, wie es bereits aus dem *Design Editor* bekannt.

## 6 Unterrichtliche Umsetzung

Für den Einstieg eignet sich eine Präsentation fertiger Projekte, wobei meist das Smartphone selbst schon eine Motivation für die Schülerinnen und Schüler darstellt. Für die Präsentation können Projekte anderer Schülergruppen oder auch unsere Beispiele genutzt werden. Im Anschluss können die Schülerinnen und Schüler schrittweise mit der Entwicklungsumgebung des App Inventor und den einzelnen Bausteinen vertraut gemacht werden. Im weiteren Verlauf können sie eigene Ideen für Projekte entwickeln und umsetzen.

Die Grundlagen für dieses Modul und die von den Schülerinnen und Schülern zu entwickelnde App stammen aus einem Modul<sup>1</sup> des Schüler\*innenlabors der RWTH Aachen.

<sup>1</sup> siehe <http://schuelerlabor.informatik.rwth-aachen.de/modul/einstieg-den-app-inventor>

## 6.1 Grober Unterrichtsplan

Unterrichtsszenarien	Kurze Zusammenfassung
Einführung, Kennenlernen und Entwicklung einer App	Anhand eines Quiz-Tutorials und/oder Lernkarten lernen die Schülerinnen und Schüler die Entwicklungsumgebung des App Inventors kennen und entwickeln Schritt für Schritt ihre erste App. Die Schülerinnen und Schüler erwerben so Grundkenntnisse in der Entwicklung von Applikationen mithilfe des App Inventors.
Vertiefung: Eigene Projekte	Eigene Ideen und Projekte werden von den Schülerinnen und Schülern entwickelt und anschließend mit dem App Inventor umgesetzt.

## 6.2 Stundenverlaufsskizzen

### 6.2.1 Einführung, Kennenlernen und Entwicklung einer App

Die Umsetzung dieses Unterrichtsszenarios benötigt keine feste Vorgabe. Daher soll hier lediglich auf die Handreichung für die Schülerinnen und Schüler (B7.1) sowie das Quiz-Tutorial (B7.2) und die Lernkarten (B7.3) zur Entwicklung einer ersten größeren App verwiesen werden. Die Schülerinnen und Schüler können mithilfe des Quiz-Tutorials eine erste, ganz einfache App selbst umsetzen. Mit Hilfe der Lernkarten gestalten sie eine App, mit der sich auf Knopfdruck ein Foto aufnehmen lässt, das anschließend durch Wischbewegungen mit dem Finger bemalt werden kann. Die Schülerinnen und Schüler benötigen dafür etwa zwei bis drei Einzelstunden. Vor dem Einsatz der Lernkarten sollte die Lehrkraft selbst einen kurzen Überblick zu der Entwicklungsumgebung geben sowie den Wechsel zwischen und Zusammenhang von Design und Blocks Editor erläutern.

### 6.2.2 Vertiefung: Eigene Projekte

Um die Kreativität der Schülerinnen und Schüler zu fördern und den App Inventor auch als Werkzeug für den eigenen Fachunterricht zu nutzen, können Sie die Schülerinnen und Schüler eigene Apps entwerfen lassen. Dabei können natürlich auch Vorgaben erteilt werden, etwa dass sich die zu entwickelnde App auf den Fachunterricht beziehen muss.

## 7 Einbettung in verschiedene Fächer und Themen

Da in diesem Modul die Entwicklung und Programmierung einer App für das Smartphone im Vordergrund steht, beziehen sich die zuzuordnenden Kompetenzen hauptsächlich auf die Informatik.

Jedoch bedeutet dies nicht, dass das Modul ausschließlich für den Informatik- oder Technikunterricht geeignet ist. Schon nach kurzer Zeit und nach Abschluss des Tutorials können die erworbenen Kenntnisse genutzt werden, um kleinere Anwendungen für Fachunterricht jenseits der Informatik und Technik zu realisieren. Beispielsweise lassen sich mit dem Wissen aus den Tutorial Werkzeugen für den Mathematikunterricht oder Vokabeltrainer für den Fremdsprachenunterricht erstellen. Zwar werden so kaum neue fachbezogene Kompetenzen vermittelt, aber die Planung und Entwicklung der App festigen bestehende Kompetenzen, weil die Schülerinnen und Schüler erworbenes Wissen reflektieren.

Die folgenden Kompetenzen finden sich entweder in den Bildungsstandards der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

## Informatik

Die Schülerinnen und Schüler ...

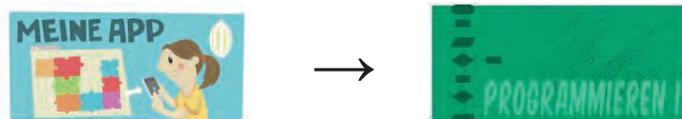
- erstellen Produkte unter Anwendung fortgeschrittener Techniken von Standardsoftware; falls es das Produkt erfordert, arbeiten sie sich in geringem Umfang in Spezialsoftware ein.
- erwerben beim Bearbeiten von Softwareprojekten in angemessenem Umfang Kenntnisse über Analyse- und Modellierungsverfahren sowie Projektmanagement.
- verstehen Programmabläufe und die Arbeitsweise von Schnittstellen.
- können erdachte Systeme in technische übertragen.
- kennen sich in Programmierumgebungen aus.
- gestalten eine Spielumgebung entsprechend der Spielidee.
- entwerfen und implementieren eine Spielvariante mit informatischen Werkzeugen.

## 8 Anschluss Themen

Als Anschluss Themen im Zusammenhang mit IT2School bieten sich folgende Module an:

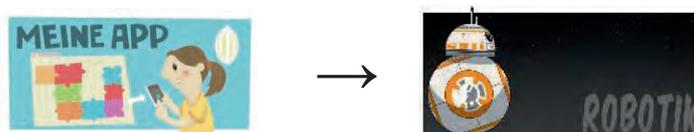
### Beispiel: Programmierung

Das Aufbaumodul A3 bietet eine gute Möglichkeit zur Vertiefung der Programmierkenntnisse. In diesem Modul lernen die Schülerinnen und Schüler die textuelle Programmiersprache Python kennen und vertiefen ihr Geschick in der Programmierung.



### Beispiel: It spielend entdecken

Möchten Sie ein nächstes kreatives Projekt mit ihren Schülerinnen und Schülern durchführen, empfehlen wir Ihnen das Modul A4 *Robotik*. Die Schülerinnen und Schüler bauen dabei aus Alltagsgegenständen einen Roboter und programmieren ihn mit dem App Inventor oder Arduino.



## 9 Literatur und Links

- Unterrichtsmaterialien und Kurse zum App Inventor. Online:  
<https://appinventor.mit.edu/explore/teach>

- Bopp, Julian (2015): Apps entwickeln wie DU willst – mit dem MIT App Inventor 2. Ideen für den Unterricht. Online:  
[https://www.juforum.de/fileadmin/downloads/oeffentlich/2014/App\\_Inventor\\_Tutorial\\_2.pdf](https://www.juforum.de/fileadmin/downloads/oeffentlich/2014/App_Inventor_Tutorial_2.pdf)
- RWTH Aachen: <https://schuelerlabor.informatik.rwth-aachen.de/en/modulmaterialien/erste-app>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B7.1	Einstieg in den App Inventor	Handreichung für den Einstieg, die in Kürze die Anmeldung beim App Inventor, den Aufbau sowie das Testen der Programme erläutert.
 B7.2	Quiz-Tutorial	Tutorial für eine einfache Quiz-App
 B7.3	Lernkarten	Lernkarten zur selbstgesteuerten Vertiefung.
 B7.4 L	BlueCoLight	Erläuterung zum BlueCoLight sowie einige Beispielprojekte zum umsetzen.
 B7.4 SuS	BlueCoLight	Erläuterungen zum BlueCoLight und zum App Inventor Projekt zum erstellen einer eigenen App für den BLueCoLight.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Android	Bezeichnet ein Betriebssystem für Smartphone- und Tablet-Systeme von Google. Es handelt sich dabei um eine freie und quelloffene Software, die auf der Grundlage von Linux entwickelt wurde.
App	Kurzbezeichnung für Applikation.
Bluetooth	Bluetooth wurde in den 1990er Jahren durch die Bluetooth Special Interest Group (SIG) entwickelt und ist seitdem ein Industrie-

	standard gemäß IEEE 802.15.1 für die Übertragung von Daten zwischen Geräten über kurze Distanz per Funktechnik (WPAN). Der Name „Bluetooth“ soll an den dänischen König Harald Blauzahn (englisch Harald Bluetooth) erinnern, der verfeindete Teile von Norwegen und Dänemark vereinte.
Drag-and-Drop	Oft auch Drag & Drop oder Drag'n'Drop, zu deutsch: „Ziehen und Ablegen“, stellt eine Methode zum Bewegen von Elementen auf grafischen Benutzeroberflächen dar. Ein Element kann dabei mithilfe eines Zeigegerätes (z. B. Maus) gezogen und an seinem Bestimmungsort losgelassen werden. Allgemeinen wird Drag-and-Drop genutzt, um Aktionen auszuführen oder Beziehungen zwischen Objekten herzustellen.
Editor	Engl. <i>to edit</i> . (Texte etc.) herausgeben, redigieren, bearbeiten, (Filme) schneiden. Ein Editor ist ein Computerprogramm, das zur Erstellung und Bearbeitung von digitalen Daten (Dateien) verwendet wird (deutsch auch: Bearbeitungsprogramm).
Emulator	Software-Emulatoren sind Programme, die Eigenschaften eines Computers oder eines Handys nachbilden und es so ermöglichen, Software für die Zielhardware auf einem Computer mit einer anderen Architektur zu verwenden und zu testen. So können z. B. Spiele für ältere Spielekonsolen auch auf einem PC oder einer neueren Spielekonsole gespielt werden oder Handyapps auf dem Computer getestet werden, ohne jedes Mal ein Handy benutzen zu müssen.
Google-Account	Auch Google-Konto; ein bei Google registriertes Benutzerkonto, mit dem verschiedene Google-Dienste genutzt werden können. In dem Konto werden z. B. personenbezogene Daten und Informationen gespeichert und verknüpft. Mit einem Google-Konto können Nutzer bei verschiedenen Anwendungen, auch bei Drittanbietern, anmelden, ohne jedes Mal neue Zugangsdaten eingeben zu müssen.

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Einstieg in den App Inventor

In diesem Tutorial erfährst du, wie du Apps für Smartphones mit Android und iOS Betriebssystem programmieren kannst. Du nutzt dafür das Programm **App Inventor**, das nur online im Browser (z. B. Firefox oder Chrome, nicht Internet Explorer) aufgerufen werden kann.

## 1 Anmeldung und Erstellung eines Projekts

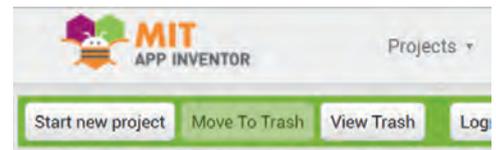
Um die Programmierung zu starten, musst du den App Inventor aufrufen. Dafür benötigst du ein Benutzerkonto (Account) bei Google, beachte hier bitte die Hinweise deiner Lehrkraft. Sie wird dir sagen, mit welchem Benutzerkonto du dich anmelden kannst.

1. Rufe im Browser folgende Internetseite auf: <https://appinventor.mit.edu>

2. Klicke auf „Create Apps!“ (oben links auf der Webseite). Danach musst du dich mit dem Google-Account anmelden.

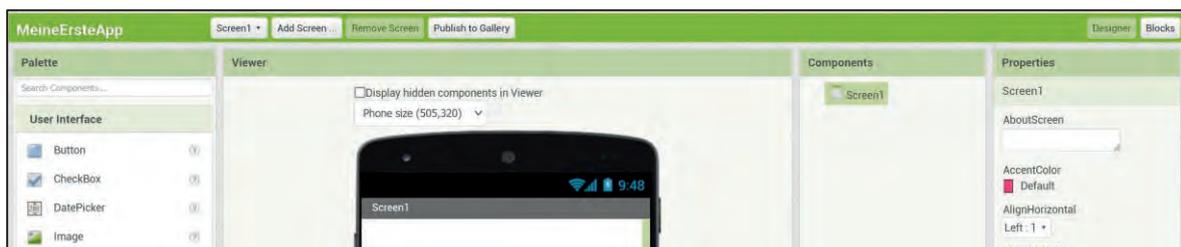


3. Sobald du eingeloggt bist, kannst du auf „Projects“, „My Projects“ und dann auf „Start new project“ klicken, um ein neues Projekt anzulegen. Gib nun einen Namen für das Projekt an; zum Beispiel „MeineErsteApp“. Danach solltest du die Entwicklungsumgebung vom App Inventor sehen.



## 2 Entwicklungsumgebung – Design Editor

Die Entwicklungsumgebung des App Inventor umfasst zwei Editoren, die jeweils in mehrere Bereiche aufgeteilt sind. Den ersten Editor siehst du bereits. Hierbei handelt es sich um den **Design Editor**. Im Bild unten siehst du die Aufteilung in *Palette*, *Viewer*, *Components* und *Properties*. Einen Gesamtüberblick mit Erläuterungen zum Design Editor findest du auf den letzten Seiten dieses Tutorials.



In **Palette** findest du alle Komponenten, die dir für deine App zur Verfügung stehen. Unter *User Interface* (Benutzerschnittstelle) beispielsweise *Buttons* (Knöpfe), unter *Media* die *Camera* und unter *Drawing and Animation* (Zeichnen und Bewegung) das *Canvas* (Leinwand). Die einzelnen Komponenten kannst du in den **Viewer** ziehen. Im **Viewer** (Betrachter) siehst du, wie deine App später auf dem Handy aussehen wird. Unter **Components** werden die Komponenten angezeigt, die du schon in deine App eingefügt hast. Hier kannst du Komponenten auch umbenennen und löschen. Unter **Properties** werden die Eigenschaften der Komponente angezeigt, die du gerade unter **Components** oder im **Viewer** angeklickt hast, um zum Beispiel Texte von Buttons ändern oder Hintergrundfarben für ein *Canvas* auswählen.

### 3 Entwicklungsumgebung – Blocks Editor

Im Design Editor kannst du grundlegende Komponenten für deine Apps einbauen. Damit diese jetzt aber auch tun, was sie sollen, musst du es ihnen beibringen.

Die einzelnen Komponenten der App müssen also mit Funktionen ausgestattet werden. Dies geschieht im **Blocks Editor**, der für jede Komponente verschiedene Funktionsbausteine zur Verfügung stellt, die wie ein Puzzle zusammengebaut werden. Zwischen dem **Design** und dem **Blocks Editor** schaltest du mit den Buttons *Designer* und *Blocks* im oberen rechten Bereich der Webseite um. Sobald du zum Blocks Editor gewechselt bist, siehst du, wie im Bild unten, eine Aufteilung in *Blocks* und *Viewer*.



Unter *Blocks* findest du Kategorien wie *Control* (Kontrolle), *Logic* (Logik) und *Math* (Mathematik) mit vordefinierten Bausteinen. Die Farbkodierung sollte dir helfen, die richtigen Bausteine zu finden. Weiter unten findest du in diesem Bereich auch die Komponenten, die du vorher im Design Editor hinzugefügt hast. In den *Viewer* werden die Bausteine gezogen und miteinander verbunden. Hier wird der App gesagt, was wann zu tun ist.

### 4 Testen einer App

App Inventor bietet mehrere Möglichkeiten zum Testen der erstellten Apps. So kann die App über WIFI oder USB-Kabel auf das Smartphone geladen werden. Sofern kein Smartphone vorhanden ist, kann auch ein Programm (ein Emulator) zum Testen verwendet werden.

#### Verbindung über WLAN

1. Installiere zunächst die **MIT App Inventor-App (iOS)** bzw. die **MIT AI2 Companion App (Android)** auf deinem Gerät

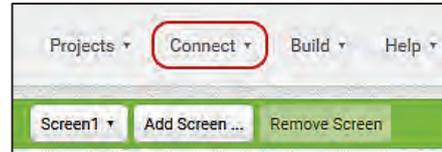


iOS



Android

2. Melde den PC und das Smartphone selben WLAN an.
3. Starte die App auf deinem Smartphone.
4. Klicke im App Inventor auf *Connect*.
5. Wähle *AI Companion* aus, scanne den QR-Code mit dem App Inventor App ein ein, und schon kannst du deine eigene App testen.



## Verbinden über USB-Kabel (nur Android)

1. Installiere zunächst die **MIT AI2 Companion-App** aus Google Play auf deinem Android-Gerät. Zu der App gelangst du unter anderem über folgenden Link:

<https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>

2. Installiere die **App Inventor-Software** auf dem Computer (**auf das richtige Betriebssystem achten; unter Windows 7 und aufwärts als Administrator**). Zu dem Programm gelangst du über folgenden Link (siehe Schritt 1):

<http://explore.appinventor.mit.edu/ai2/setup-device-usb>

3. Öffne das Programm **aiStarters** auf dem Computer.
4. Verbinde das Android-Gerät und den Computer über ein USB-Kabel.
5. Aktiviere auf dem Android-Gerät *USB-Debugging*, sofern dies nicht aktiv ist (unter Einstellungen > Entwickleroptionen zu finden).
6. Klicke im App Inventor auf *Connect*.
7. Wähle USB aus und warte kurz.
8. Nun kannst du deine App testen.



## Verbindung zum Emulator

1. Installiere die **App Inventor-Software** für das entsprechende Betriebssystem auf dem Computer (**auf das richtige Betriebssystem achten, unter Windows 7 und aufwärts als Administrator**). Zu dem Programm gelangst du über folgenden Link (siehe Schritt 1):

<http://explore.appinventor.mit.edu/ai2/setup-device-usb>

2. Öffnen das Programm **aiStarters** auf dem Computer.
3. Klicke im App Inventor auf *Connect* und wähle *Emulator* aus.
4. Warte bis die Companion App sich im Emulator öffnet. (Dies kann etwas länger dauern! Vorher bitte nichts im Emulator machen ...)
5. Wenn eine Update-Meldung im App Inventor (Browser) erscheint, dann musst du den Emulator vorher noch updaten. Folger dazu folgender Schritt für Schritt Anweisung:
  - a. In der *Replace application* Meldung auf *OK* klicken
  - b. Klicke dann auf *Install*
  - c. Nach der Installation muss du den Emulator schliesen und wieder bei Schritt 2 beginnen
6. Warten bis der Emulator die Daten der App erhalten hat und diese anzeigt.
7. Nun kannst du deine App testen.



**Hinweis:** Die Verwendung des Emulators ist zwar der Vollständigkeits halber erwähnt, jedoch wird die Verwendung der **Companion-App** mittels WLAN empfohlen.

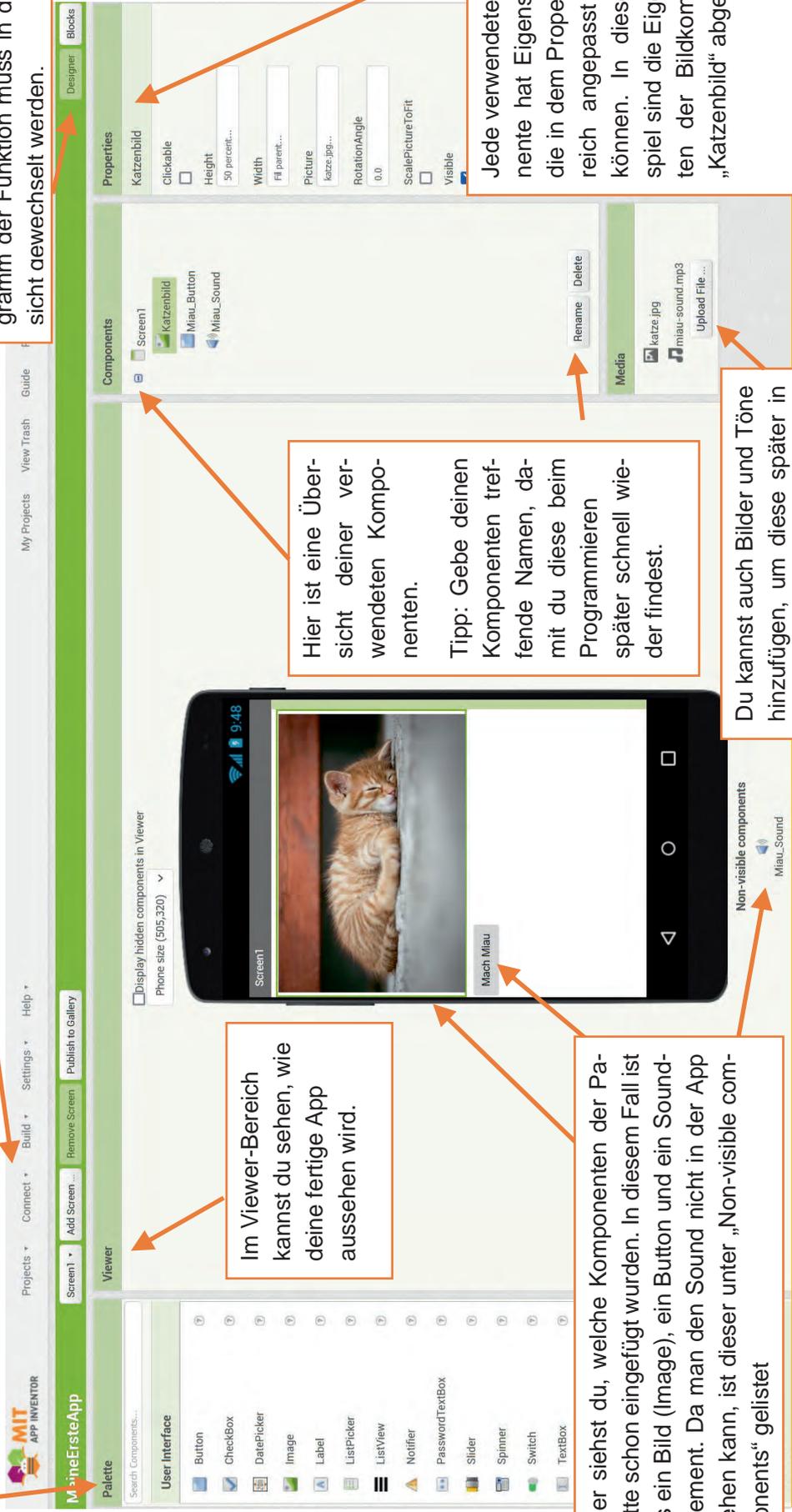
# Arbeitsmaterial B7.1

## Übersicht Design Editor

In der Palette befinden sich alle Komponenten, die du zum Bauen der App benötigst.

Unter Connect findest du die Optionen zum Testen deiner App. Unter Build kannst du die App herunterladen (nur kompatibel zu Android Geräten)

Aktuell befindest du dich in der „Designer“ Perspektive. Hier wird das Aussehen der App fertiggestellt, aber nicht die Funktion. Zum Programm der Funktion muss in die „Blocks“-Ansicht gewechselt werden.



Im Viewer-Bereich kannst du sehen, wie deine fertige App aussehen wird.

Hier ist eine Übersicht deiner verwendeten Komponenten.  
Tipp: Gebe deinen Komponenten treffende Namen, damit du diese beim Programmieren später schnell wieder findest.

Hier siehst du, welche Komponenten der Palette schon eingefügt wurden. In diesem Fall ist es ein Bild (Image), ein Button und ein Sound-Element. Da man den Sound nicht in der App sehen kann, ist dieser unter „Non-visible components“ gelistet

Du kannst auch Bilder und Töne hinzufügen, um diese später in der App einzubinden. Klicke dafür auf „Upload File ...“

Jede verwendete Komponente hat Eigenschaften, die in dem Properties Bereich angepasst werden können. In diesem Beispiel sind die Eigenschaften der Bildkomponente „Katzenbild“ abgebildet.

## Arbeitsmaterial B7.1

### Übersicht Blocks Editor

In diesem Bereich findest du Bausteine, die du zum Erstellen deiner App benötigst. Diese sind nach Farben und Funktion sortiert.

Aktuell befindest du dich in der „Blocks“ Perspektive. Hier wird die Funktion der App fertiggestellt. Falls du etwas am Aussehen der App oder den Komponenten ändern möchtest, dann musst du wieder in die „Design“ Perspektive zurück wechseln.

When Clicked

- do
- call Miau\_Sound .Play

katzenbild.jpg

miau-sound.mp3

Upload File ...

Per Drag and Drop kannst du die Bausteine in die Mülltonne schieben, um diese von der Arbeitsfläche zu löschen.

Der Rucksack ermöglicht es dir Bausteine zwischen deinen Projekten zu kopieren. Das ist praktisch, falls du Funktionen aus einem anderen Projekt schnell übernehmen möchtest.

Wenn du den Miau\_Button klickst, dann wird die Audiodatei der Komponente Miau\_Sound abgespielt.

Hier ist eine Übersicht der verwendeten Komponenten aus der Designer-Perspektive. Mit einem Klick auf die Komponente, werden dir die dazugehörigen Funktionen angezeigt.

Das ist die Arbeitsfläche. Hier entsteht dein Programm. Mit einem Klick auf die rechte Maustaste auf der Arbeitsfläche (oder den Programmierbausteinen) werden nützliche Optionen angezeigt.

# App Inventor – Quiz Tutorial

In diesem Tutorial wirst du ein kleines Quiz mit dem App Inventor erstellen. Dabei handelt es sich aber erst einmal nur um eine grobe Vorlage, die du im Anschluss noch erweitern kannst.

Unsere Quiz-App wird direkt mit der ersten Frage starten und bei richtiger Antwort mit der nächsten Frage weitermachen. Nach richtiger Beantwortung der zweiten Frage kann dann schon der Punktestand erscheinen.

## Hinweis

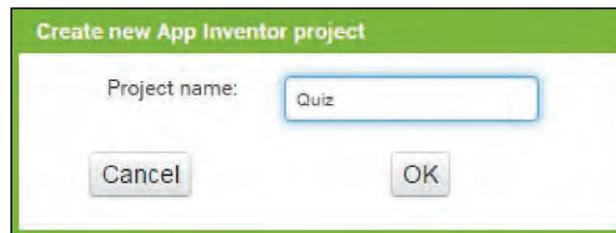
Es wird davon ausgegangen, dass du bereits eine kurze Einführung in den App Inventor hattest und weißt ...

- wo und wie du die Entwicklungsumgebung aufrufst,
- was die Design- und Block-Ansicht ist und wie sie aufgebaut sind.

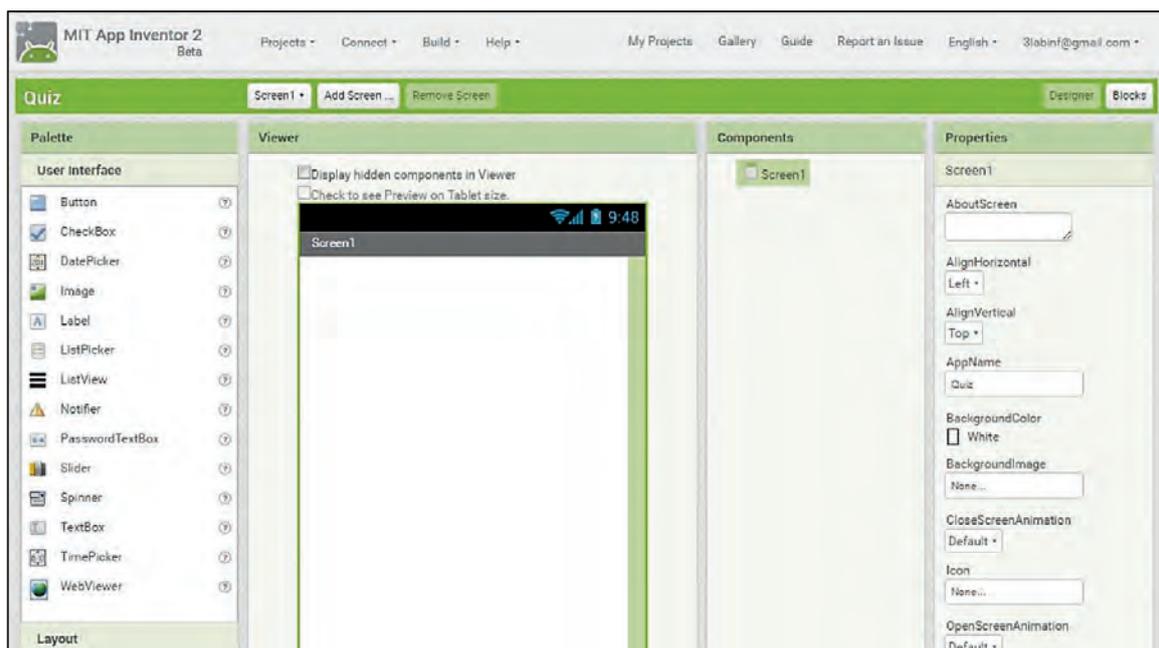
Wenn du bisher keine Einführung hattest, dann lies bitte die Einführung B7.1 genau durch bevor du hier weiter machst.

## Schritt 1

Rufe die Webseite von App Inventor auf und starte die Entwicklungsumgebung. Rufe dann die Übersicht deiner Projekte auf und erstelle ein neues Projekt mit dem Namen *Quiz*.

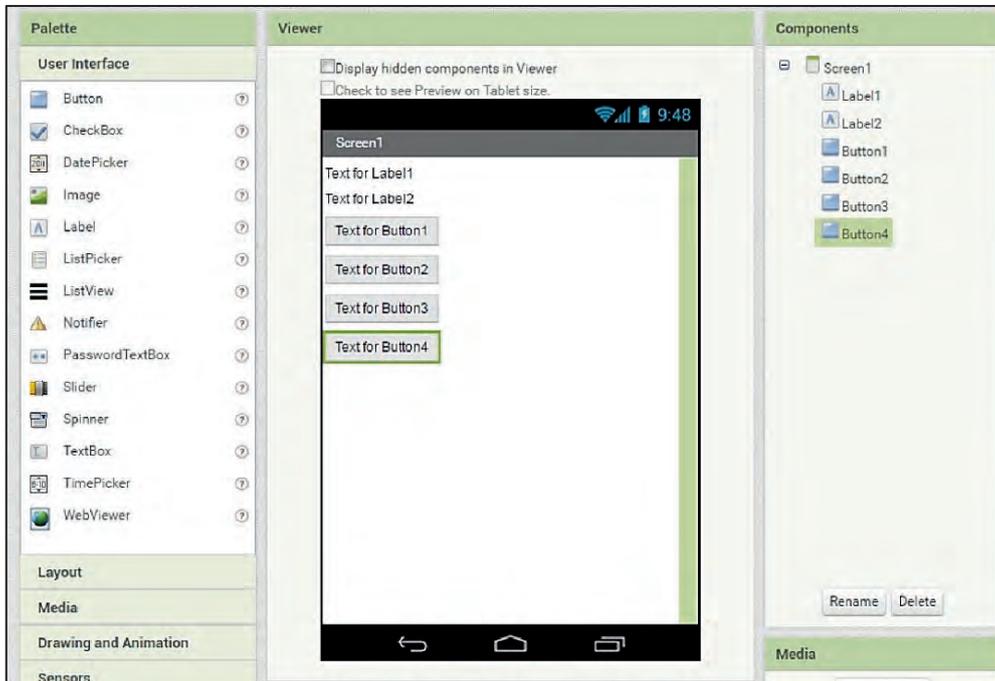


Anschließend erhältst du die gewohnte Ansicht eines neuen Projekts.



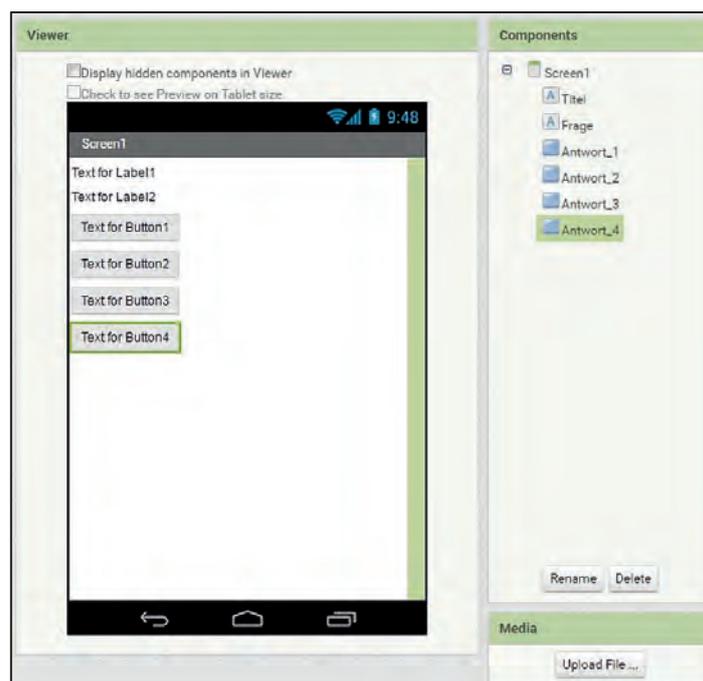
## Schritt 2

Für die erste Frage des Quiz benötigen wir zwei *Labels* und vier *Buttons*, die du von der *Palette* in den *Viewer* ziehen musst. Achte darauf, dass die Komponenten wie auf dem folgenden Bild angeordnet sind, d.h. dass die zwei *Labels* ganz oben sind und daran die vier *Buttons*.



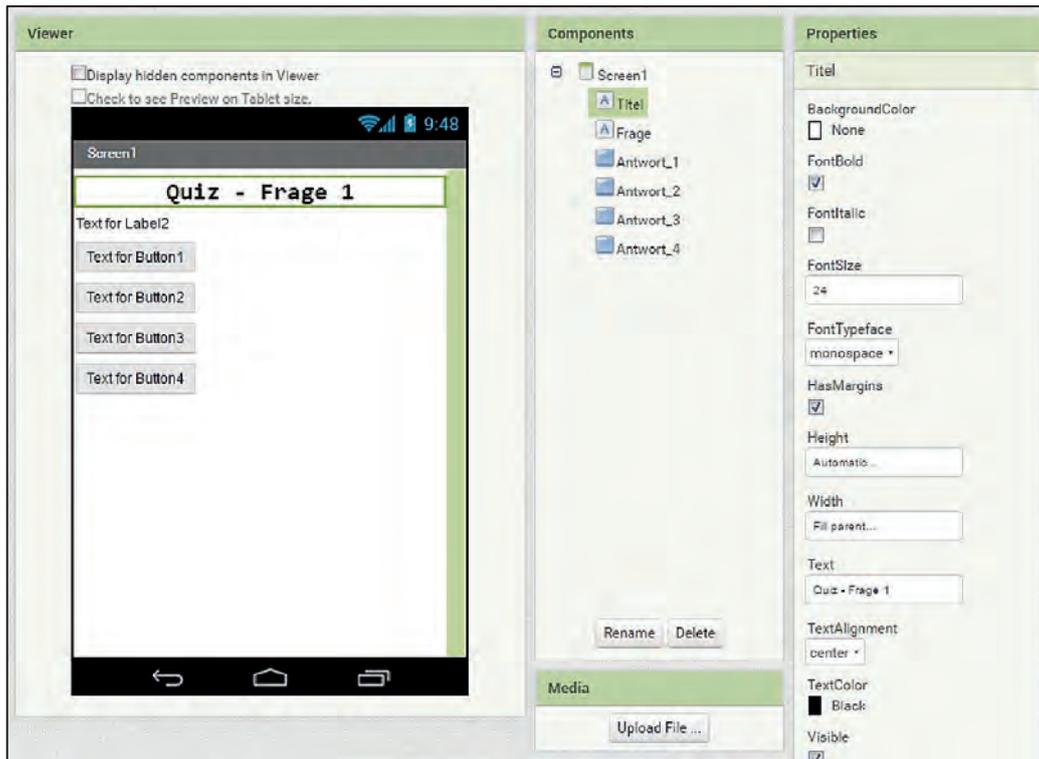
## Schritt 3

Damit wir später die *Labels* und *Buttons* besser unterscheiden können, kannst du die Komponenten umbenennen. Das erste *Label* sollte der Titel sein, das zweite *Label* enthält die Frage und die vier *Buttons* werden die Antwortmöglichkeiten. Benenne die Komponenten wie beschrieben, du kannst dich an folgendem Bild orientieren.



## Schritt 4

Anschließend kannst du den Titel ändern. Orientiere dich am besten an dem folgenden Bild.



Wichtige Eigenschaften (Properties) sind hier: FontBold (Schrift dick), FontSize (Schriftgröße), FontTypeface (Schriftart), Width (Breite des Labels), Text und TextAlignment (Textausrichtung).

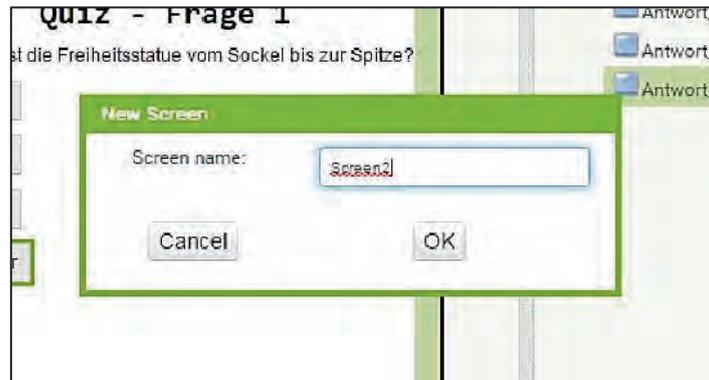
## Schritt 5

Nachdem du den Titel angepasst hast, kannst du nun auch die anderen Komponenten bearbeiten. Orientiere dich auch hier am besten an dem folgenden Bild.



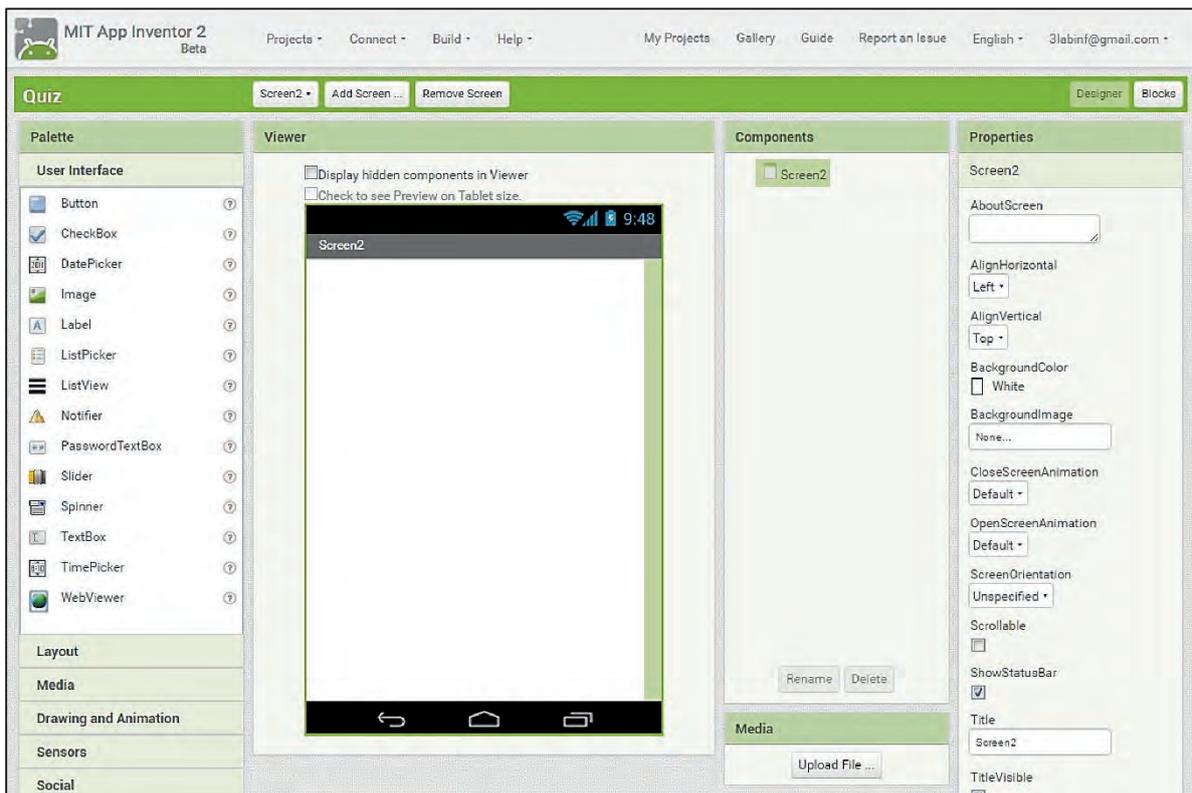
## Schritt 6

Damit wäre die Grundlage für die erste Frage geschaffen. Um nun eine weitere Frage hinzuzufügen, brauchen wir einen neuen *Screen*. Klicke dazu auf *Add Screen* und lass den Namen am besten auf *Screen2* und drücke auf *OK*.

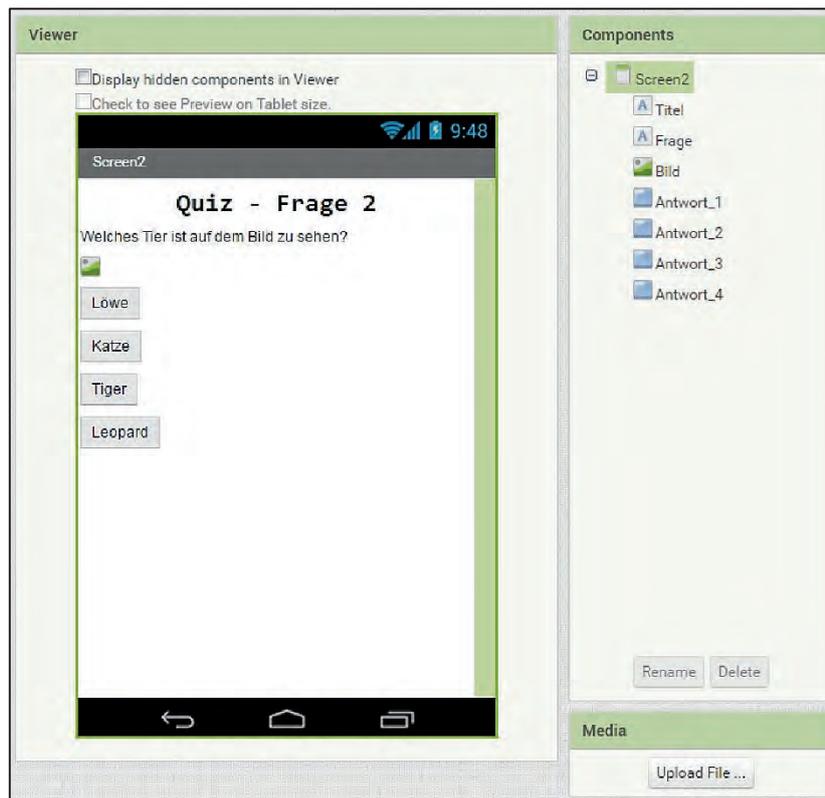


## Schritt 7

Nachdem du auf *OK* geklickt hast, wird nun der neue *Screen* angezeigt.

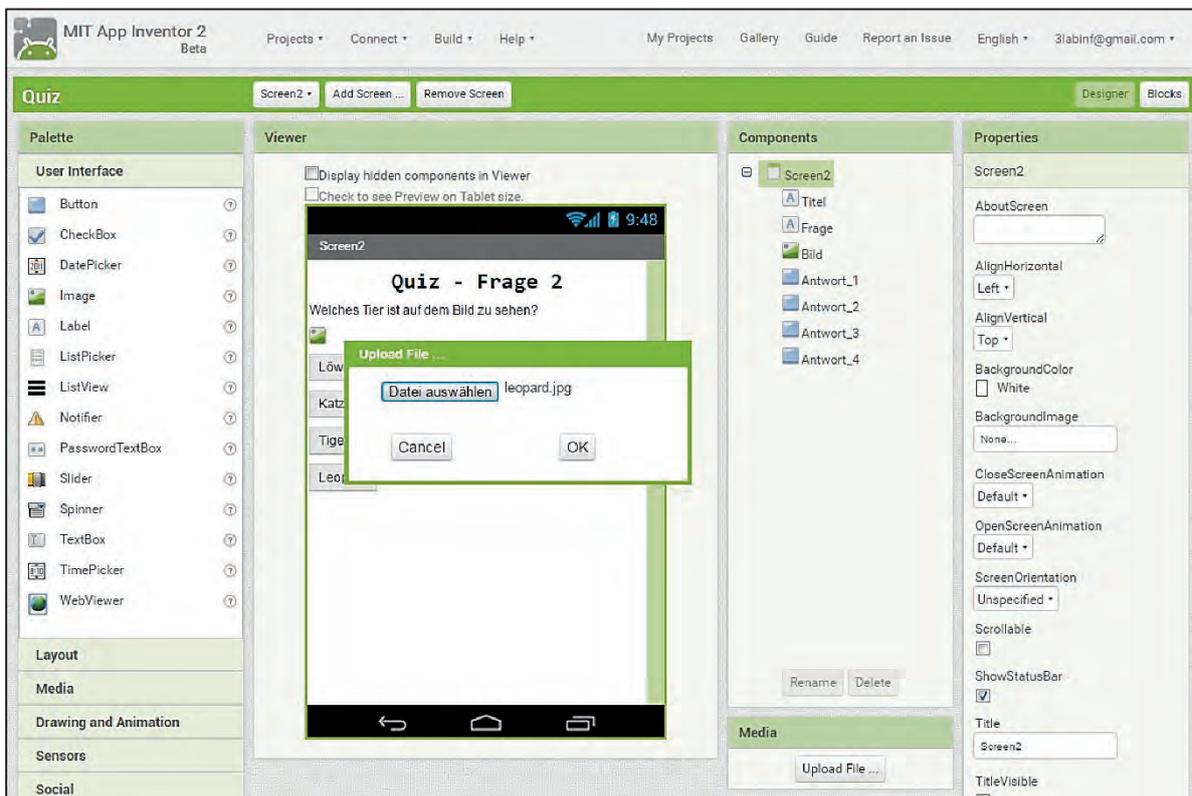


Du siehst, dass im *Viewer* und unter *Components* keines der vorherigen *Labels* und *Buttons* angezeigt wird. Daher musst du erneut zwei *Labels* und vier *Buttons* von der *Palette* in den *Viewer* des zweiten *Screens* ziehen. Ziehe anschließend aber zusätzlich auch ein *Image* (Bild) zwischen das zweite *Label* und den ersten *Button*. Dies wollen wir für die zweite Frage nutzen. Benenne die Komponenten dann am besten wieder so um, wie du es im ersten *Screen* gemacht hast und ändere auch deren Eigenschaften. Insgesamt sollte das dann wie auf dem folgenden Bild aussehen.



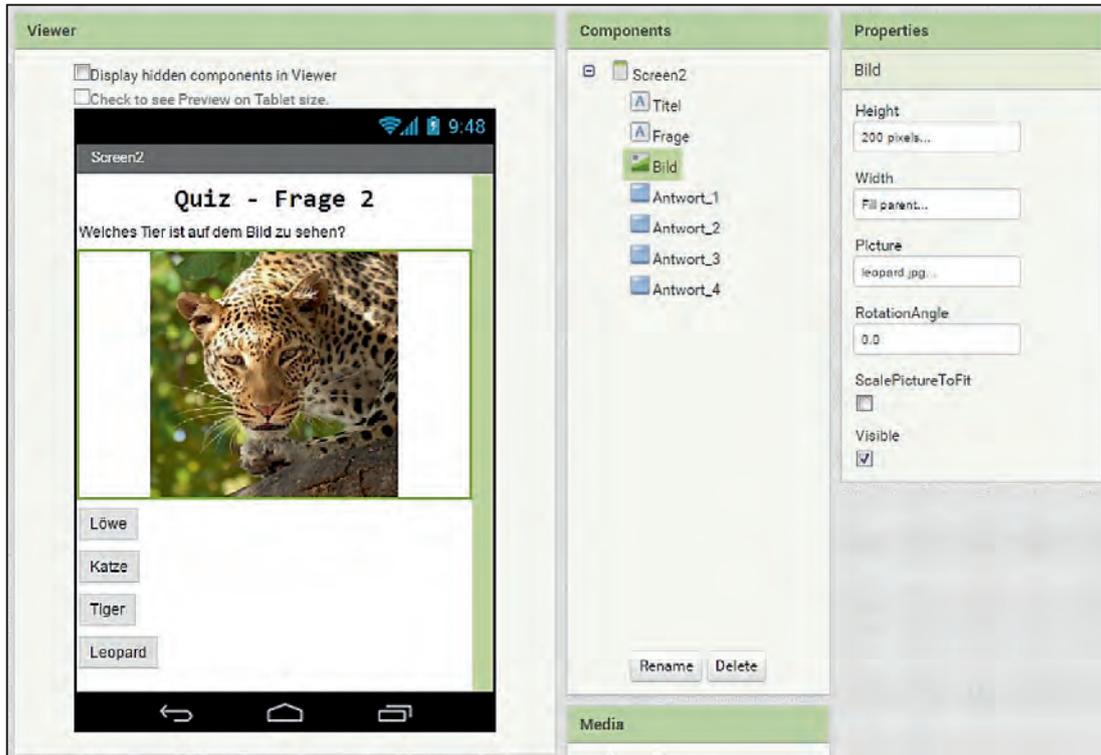
## Schritt 8

Damit ein Bild angezeigt wird und die Frage somit auch beantwortet werden kann, musst du ein Bild im App Inventor hochladen. Such dazu erst einmal ein Bild eines Leopards im Internet und klicke anschließend auf *Upload File ...* um dieses Bild hochzuladen.



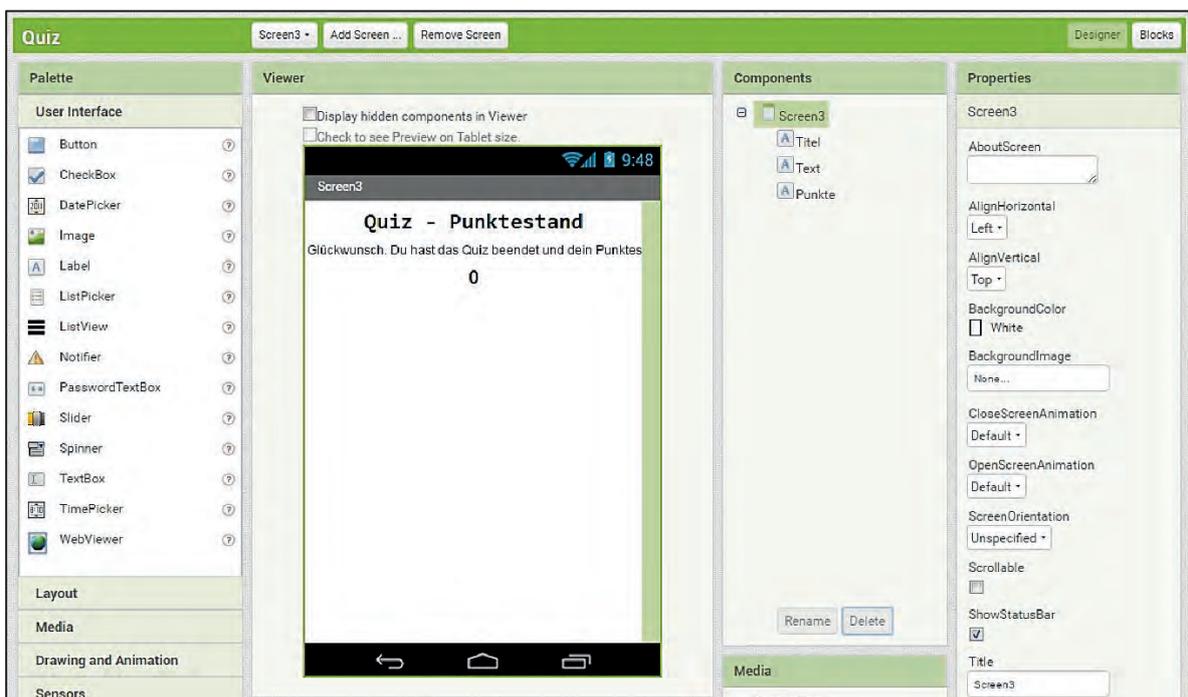
## Schritt 9

Wähle dann die Image-Komponente aus und ändere die Eigenschaft *Picture*. Anschließend sollte das Ergebnis ungefähr so aussehen:



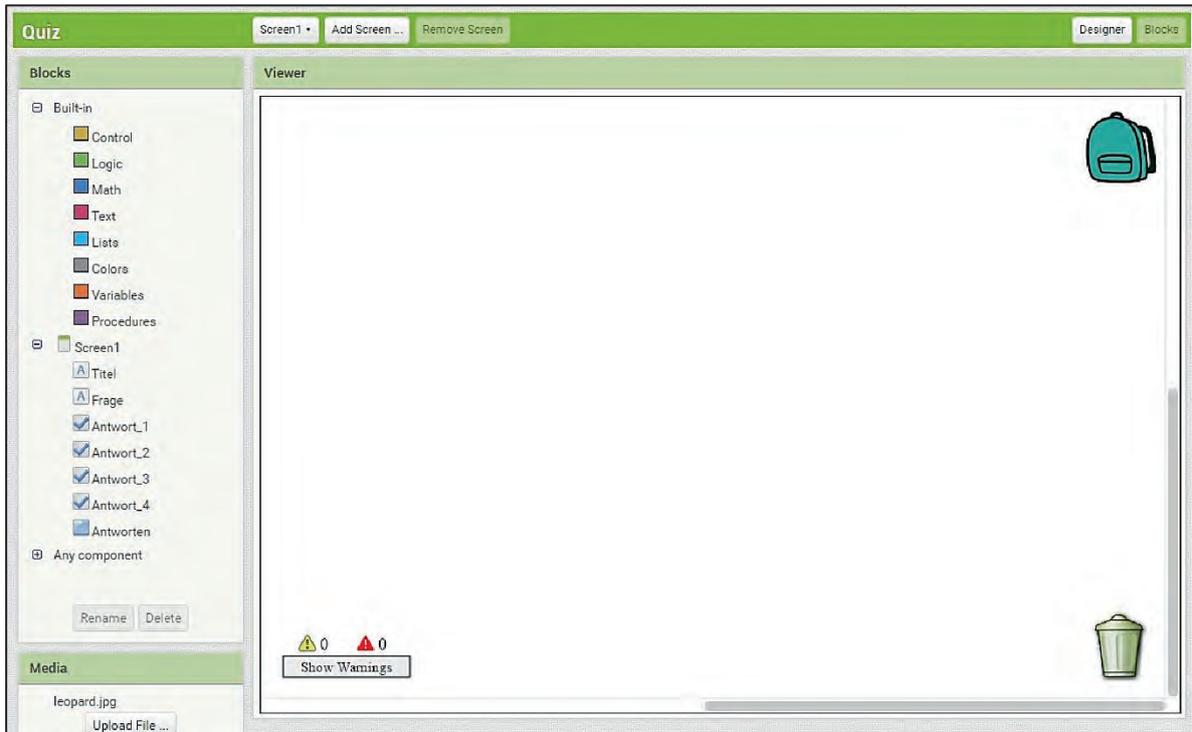
## Schritt 10

Nun brauchen wir einen dritten und letzten *Screen* für den Punktestand des Quiz. Klicke dazu wieder auf *Add Screen* und füge einen neuen *Screen* mit Namen *Screen3* hinzu. Füge, sobald der neue *Screen* angezeigt wird, drei *Labels* hinzu und ändere deren Eigenschaften so, dass du folgendes Ergebnis erhältst:



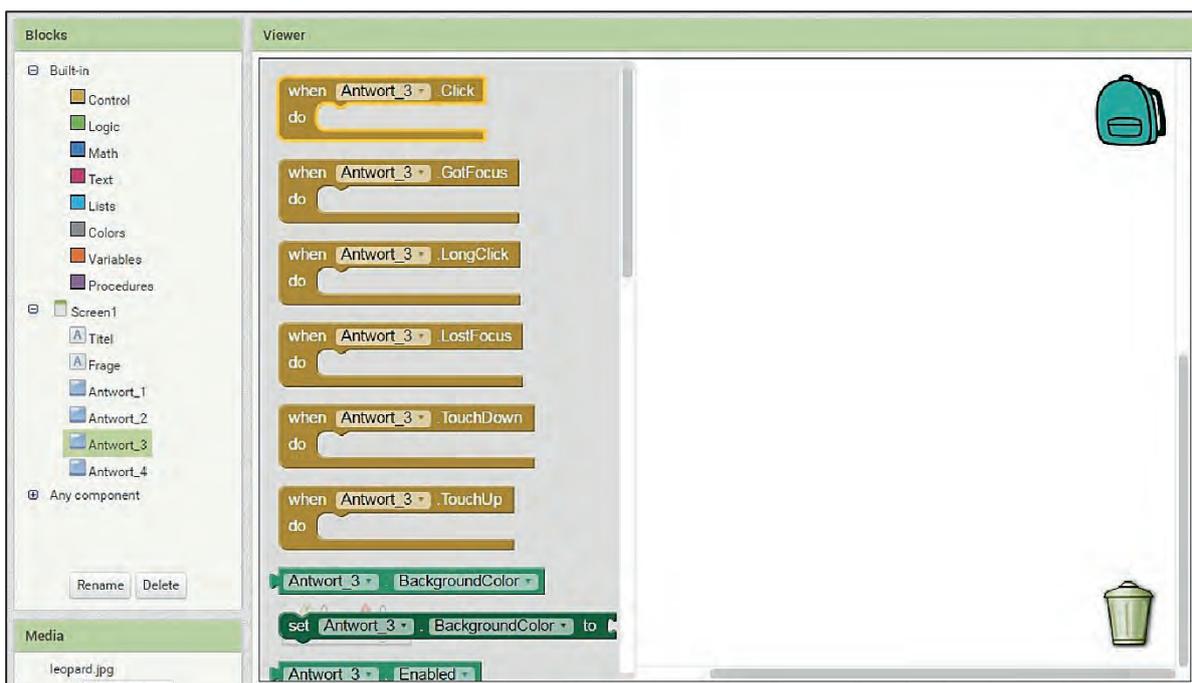
## Schritt 12

Damit ist die grafische Vorlage des Quiz fertig und du kannst dich an die Programmierung machen. Wähle dazu *Screen1* aus dem Dropdown-Menü aus und wechsle dann in die *Blocks*-Ansicht.



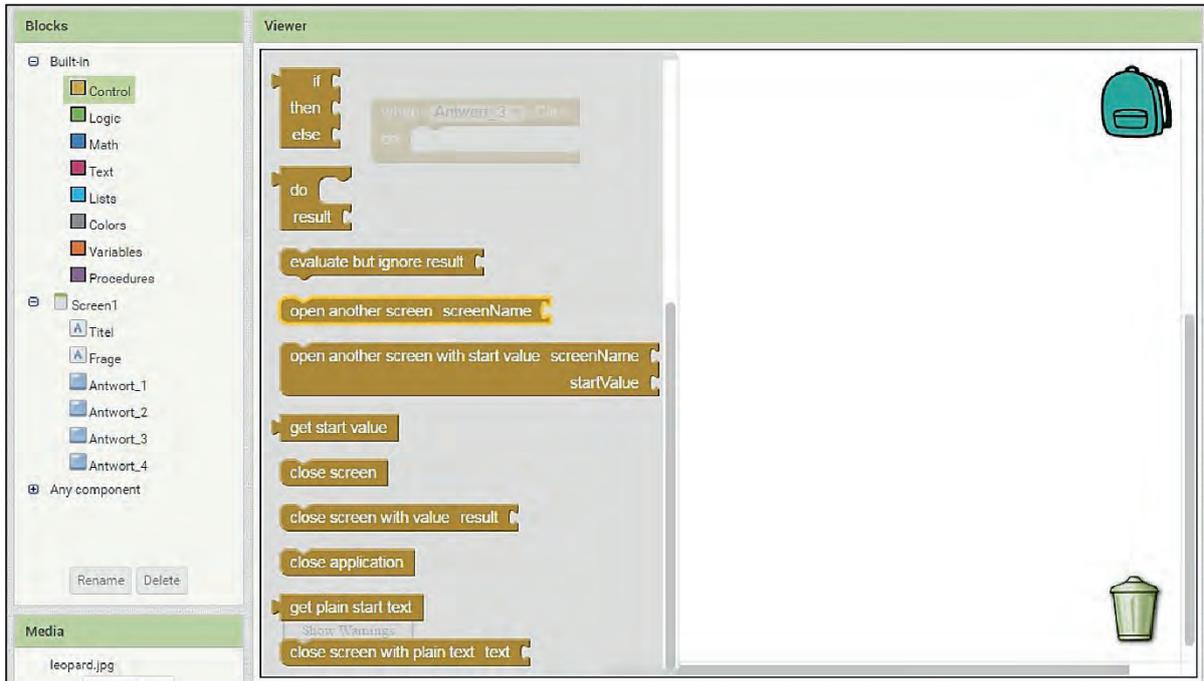
## Schritt 13

Zunächst soll man beim Drücken der richtigen Antwort zur nächsten Frage kommen. Dazu brauchst du den Baustein, welcher für das Drücken von *Antwort\_3* (das ist hier richtige Antwort) zuständig ist. Klicke unter *Blocks* auf *Antwort\_3* und ziehe den Baustein *when Antwort\_3.Click* in den *Viewer*.

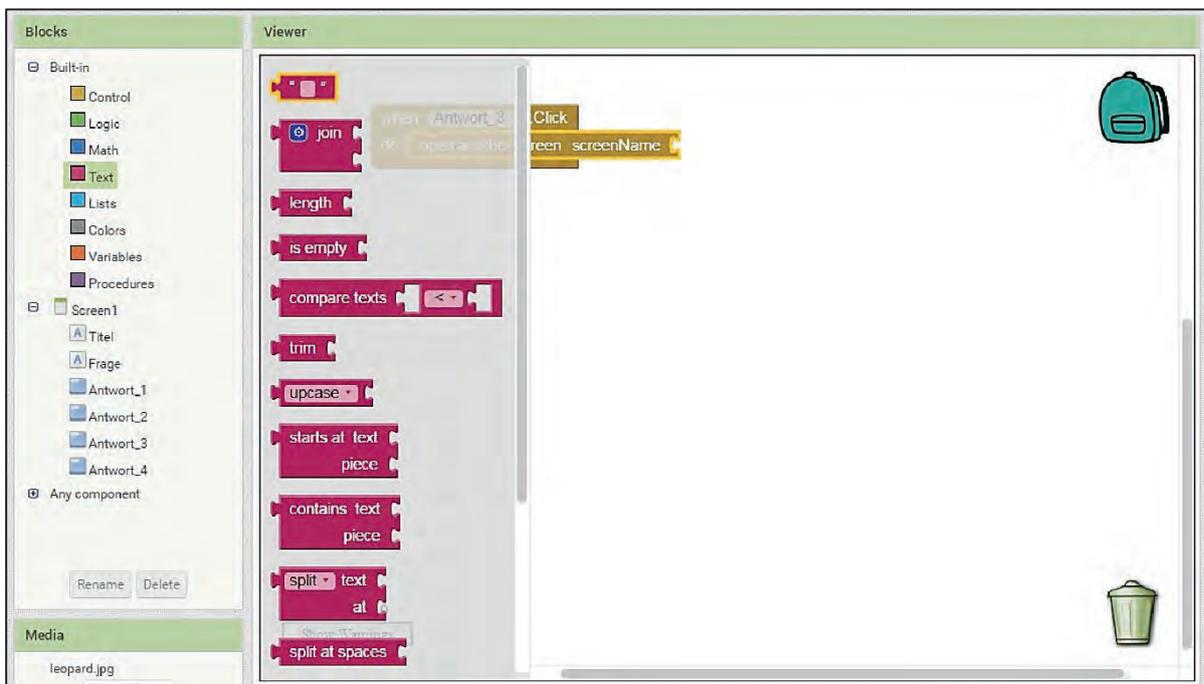


## Schritt 14

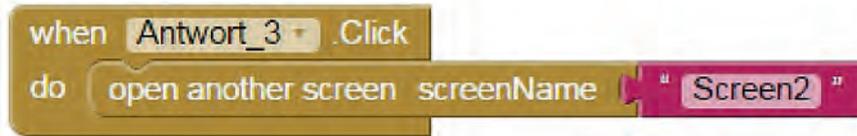
Damit nun auch auf den zweiten Screen gewechselt wird, benötigst du noch den Baustein *open another screen*, den du unter Control findest.



Damit die App weiß, welcher Screen geöffnet werden soll, brauchst du noch den Baustein “ ” für eine *String* (Zeichenkette), welchen du unter *Text* findest.



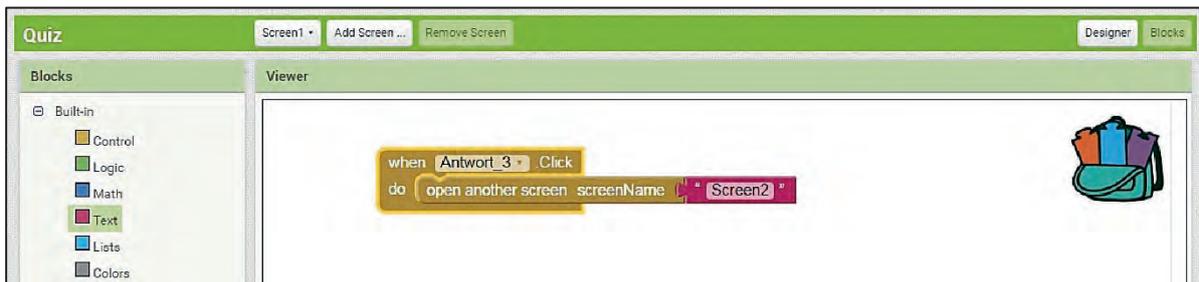
Setze nun die Bausteine zusammen und ändere den Inhalt des *String*-Bausteines in *Screen2*. Das Ergebnis sollte so aussehen:



## Schritt 15

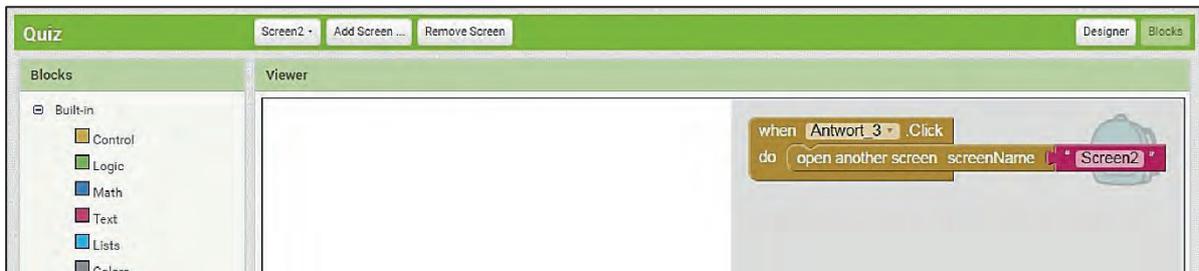
Nun kommt der Wechsel auf den dritten *Screen* bei richtiger Beantwortung der zweiten Frage. Damit die Arbeit aber einfacher wird, solltest du das Skript (die fertig verketteten Bausteine aus dem letzten Schritt) in den *Rucksack* in der oberen rechten Ecke des *Viewers* ziehen. Dies hat den Vorteil, dass du das *Skript* auch in anderen *Screens* aus dem *Rucksack* herausziehen kannst und dir damit Arbeit sparst.

Nachdem du das *Skript* in den *Rucksack* gezogen hast, solltest du folgendes Ergebnis haben.



## Schritt 16

Wechsle nun mittels des Dropdown-Menüs in den *Screen2*. Du siehst, dass dieser *Screen* bisher keine Bausteine im *Viewer* besitzt. Klickst du aber auf den *Rucksack*, dann siehst du das *Skript*, welches du vorhin hineingezogen hast. Ziehe das Skript aus dem *Rucksack* in den *Viewer*.



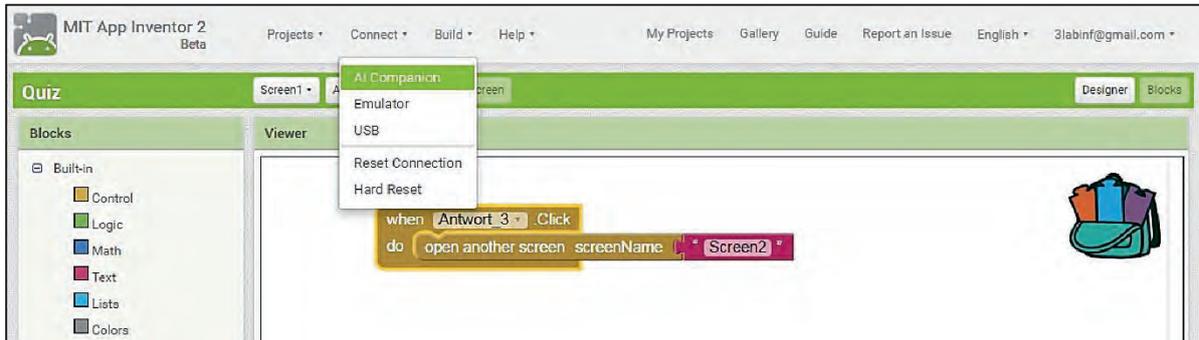
## Schritt 17

Da bei der zweiten Frage die vierte Antwortmöglichkeit die Richtige ist, musst du das *Skript* etwas abändern. Ändere *Antwort\_3* zu *Antwort\_4* und *Screen2* zu *Screen3*. Es sollte danach so aussehen:



## Schritt 18

Es wird Zeit für einen ersten Test der App. Wechsel dazu in den *Screen1* und klicke oben auf *Connect* und wähle *AI Companion* aus.



Es erscheint nun eine Mitteilung mit einem QR-Code. Scanne den QR-Code mit deinem Smartphone und teste deine App.

## Schritt 19

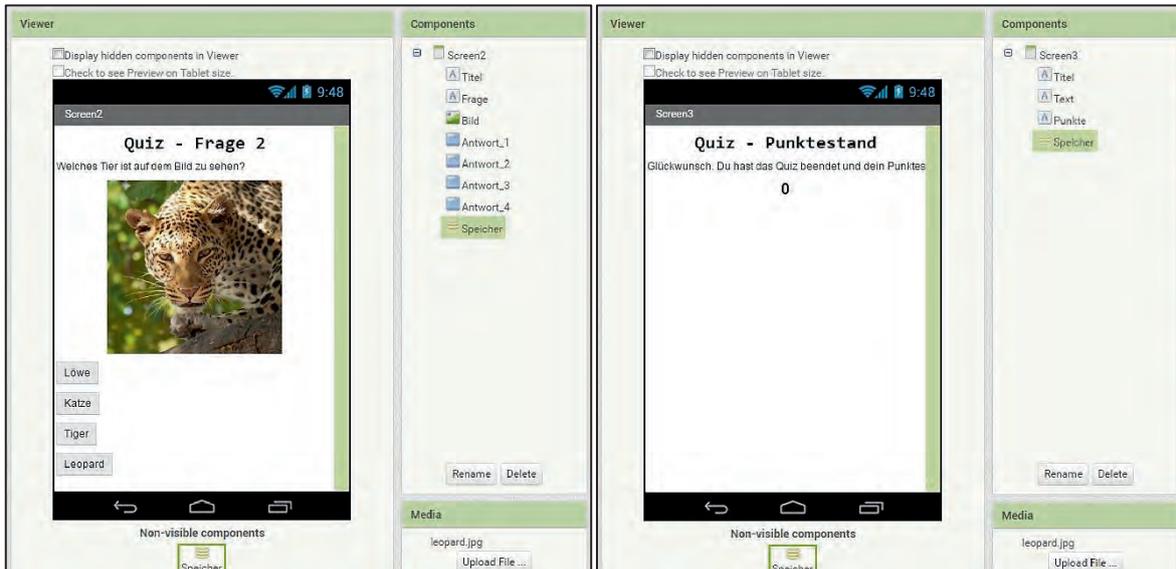
Deine App sollte nun eigentlich schon recht gut funktionieren. Es fehlt nun nur ein Punkttestand und ein Hinweis, dass eine falsche Antwort gewählt wurde. Kümmere dich erst einmal um die Punkte.

Dazu brauchst du eine neue Komponente, weshalb du in den *Designer*-Ansicht wechseln musst. Wähle dann *Screen1* über das Dropdown-Menü aus, sofern dieses nicht bereits offen ist. Anschließend musst du *TinyDB*, welche du in der *Palette* und *Storage* findest, in den *Viewer* ziehen. Benenne diese Komponente dann in *Speicher* um. Das sollte insgesamt dann so aussehen:



## Schritt 20

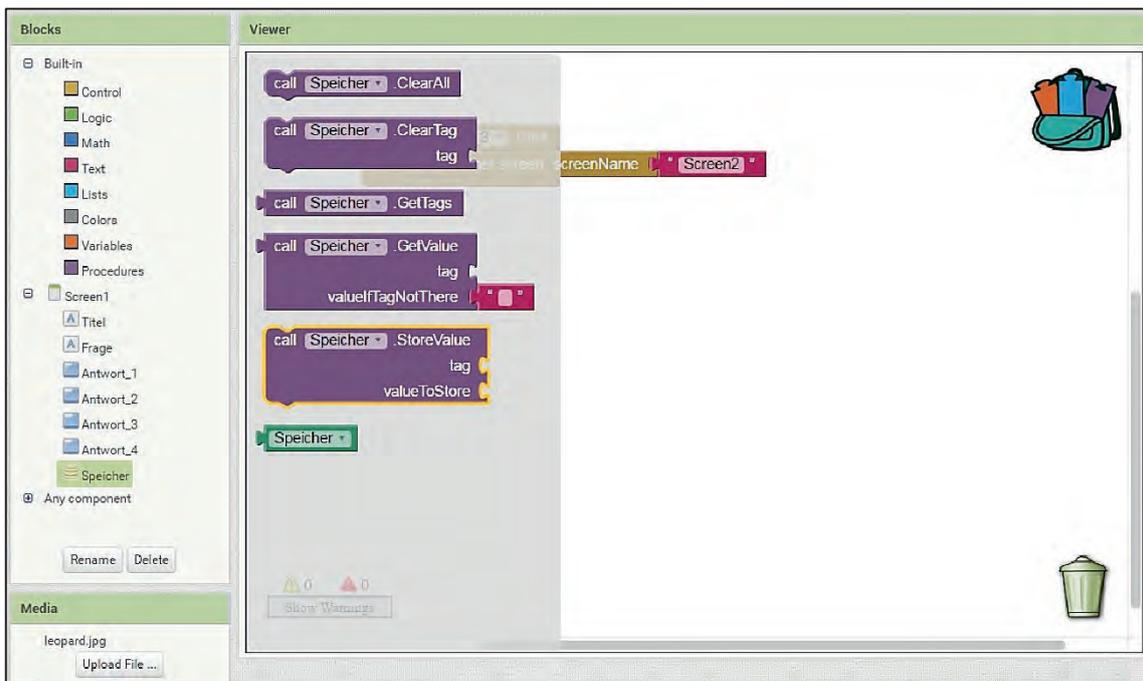
Wenn du im Dropdown-Menü nun den *Screen2* auswählst, wirst du sehen, dass dort die Komponente *TinyDB* nicht vorhanden ist. Wiederhole also das Hinzufügen der Komponente und das Umbenennen in *Speicher* auch für den *Screen2* und *Screen3*.



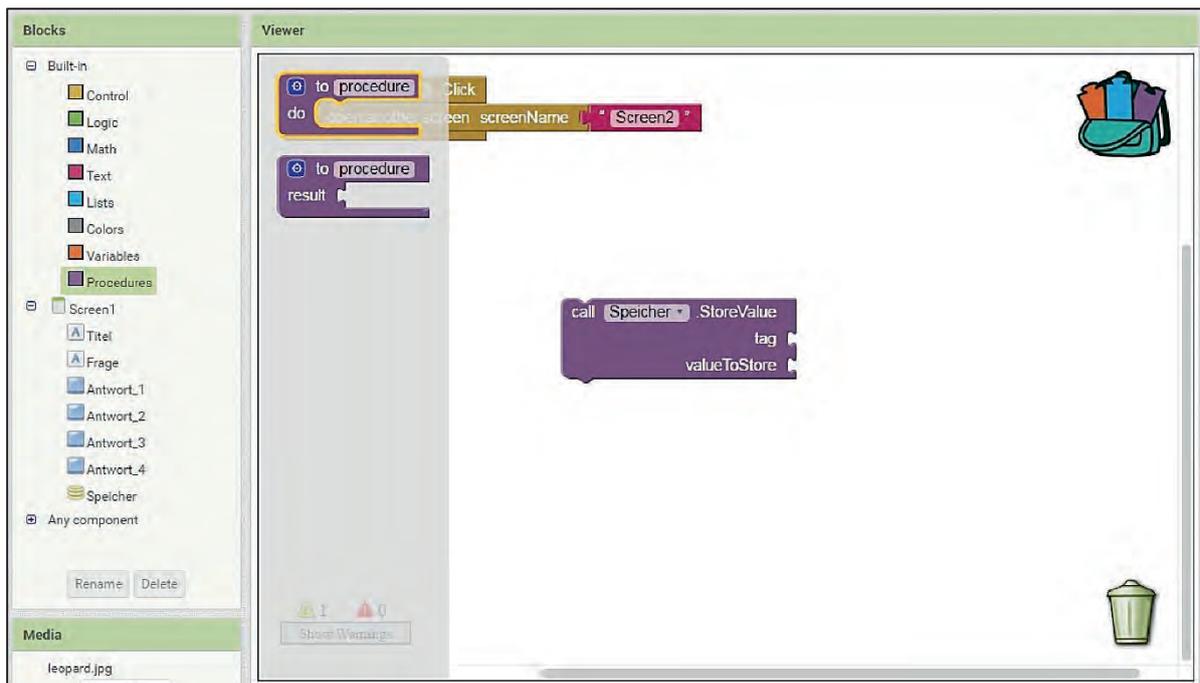
## Schritt 21

Nun wechsele wieder in die *Blocks*-Ansicht von *Screen1* und es geht weiter mit der Programmierung.

Damit du dir wieder ein wenig Arbeit sparst, baust du nun eine Prozedur (Methode), die die Änderungen am Punktestand speichert, wofür du drei Bausteine brauchst. Der erste Baustein heißt *call Speicher.StoreValue* und befindet sich unter *Block* in der Komponente *Speicher*.



Der zweite notwendige Baustein heißt *to procedure do* und befindet sich unter *Procedures*.

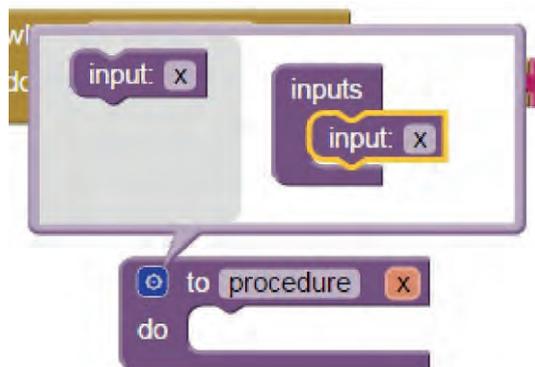


Der dritte und letzte notwendige Baustein ist *call Speicher.GetValue* und befindet sich auch bei der Komponente *Speicher*.

## Schritt 22

Wenn du nun die drei oben erwähnten Bausteine in den Viewer gezogen hast, kannst du anfangen deine Prozedur und damit einen eigenen Baustein zusammenzusetzen.

Wichtig ist hierfür der zweite Baustein *to procedure do*. Du hast bereits gesehen, dass bestimmte Bausteine bestimmte Werte angehängt werden können. Dies muss unser neuer Baustein auch können. Damit das geht, musst du auf das Zahnradsymbol klicken und in dem Fenster, das sich öffnet, den *input* Baustein einmal mit dem *inputs* Baustein verknüpfen (siehe Bild).



Anschließend kannst du den Text *procedure* und *x* des Bausteins in *Ändere\_Punkttestand* und *Änderung* ändern, so wie auf dem folgenden Bild gezeigt.



Damit hast du deinen eigenen Baustein erstellt, der in der Kategorie *Procedures* nun auffindbar ist. Doch damit der Baustein auch etwas macht, musst du ihn noch mit den anderen Bausteinen aus dem vorherigen Schritt verknüpfen. Mache es wie auf dem folgenden Bild.

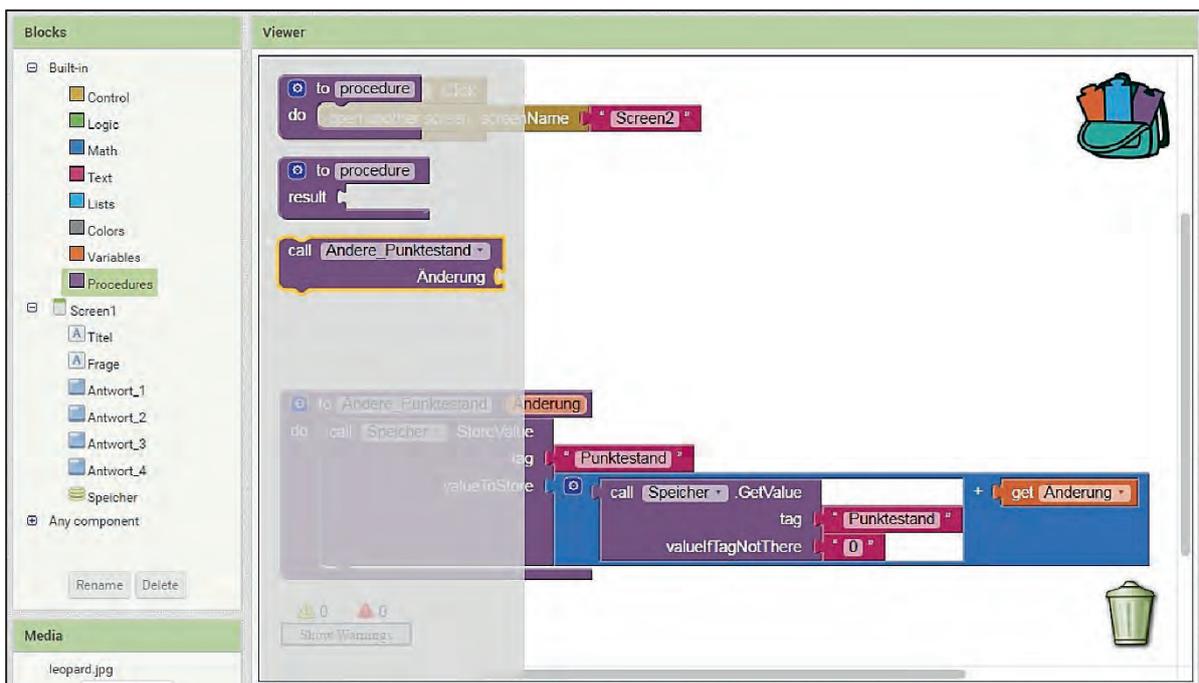


Dir fällt sicher auf, dass du noch zwei *String* Bausteine brauchst, die du unter Text findest. Außerdem brauchst du auch noch den *get Änderung* Baustein, den du bekommst, indem du auf *to Ändere\_Punkttestand Änderung* Baustein auf *Änderung* klickst (siehe Bild).



## Schritt 23

Wenn du damit fertig bist, fehlt noch die Punkteänderung, wenn man auf die richtige Antwort klickt. Füge dazu den Baustein *call Ändere\_Punkttestand* aus der Kategorie *Procedures* in den Viewer.



Binde diesen Baustein und den für eine Zahl in den when Antwort\_3.Click Baustein ein, so wie auf dem folgenden Bild gezeigt.



## Schritt 24

Sobald nun *Antwort\_3* auf dem *Screen1* angeklickt wird, wird der Punkttestand um 10 erhöht. Damit bei einer falschen Antwort der Punkttestand verringert wird und der Spieler erkennt, dass er eine falsche Antwort gegeben hat, braucht es noch etwas Arbeit von dir.

Füge nun Bausteine für die anderen Antworten auf *Screen1* hinzu und Sorge dafür, dass wenn diese Angeklickt werden, sich der Punkttestand um 5 verringert, der *Button* nicht mehr aktiv ist (*Enabled* ist auf *false* gesetzt) und die Hintergrundfarbe des *Buttons* rot ist. Du kannst dich an dem folgenden Bild orientieren. Dort ist gezeigt, welche Bausteine du für *Antwort\_1* verwenden musst.



Wenn du fertig bist, kannst du die Skripte für die richtige Antwort und für eine falsche Antwort (egal welche) und das Skript zu *Ändere\_Punkttestand* Änderung in den Rucksack ziehen, um dir für *Screen2* wieder etwas Arbeit zu sparen. Du bist nun mit *Screen1* fast fertig, es fehlt nur noch eine Kleinigkeit.

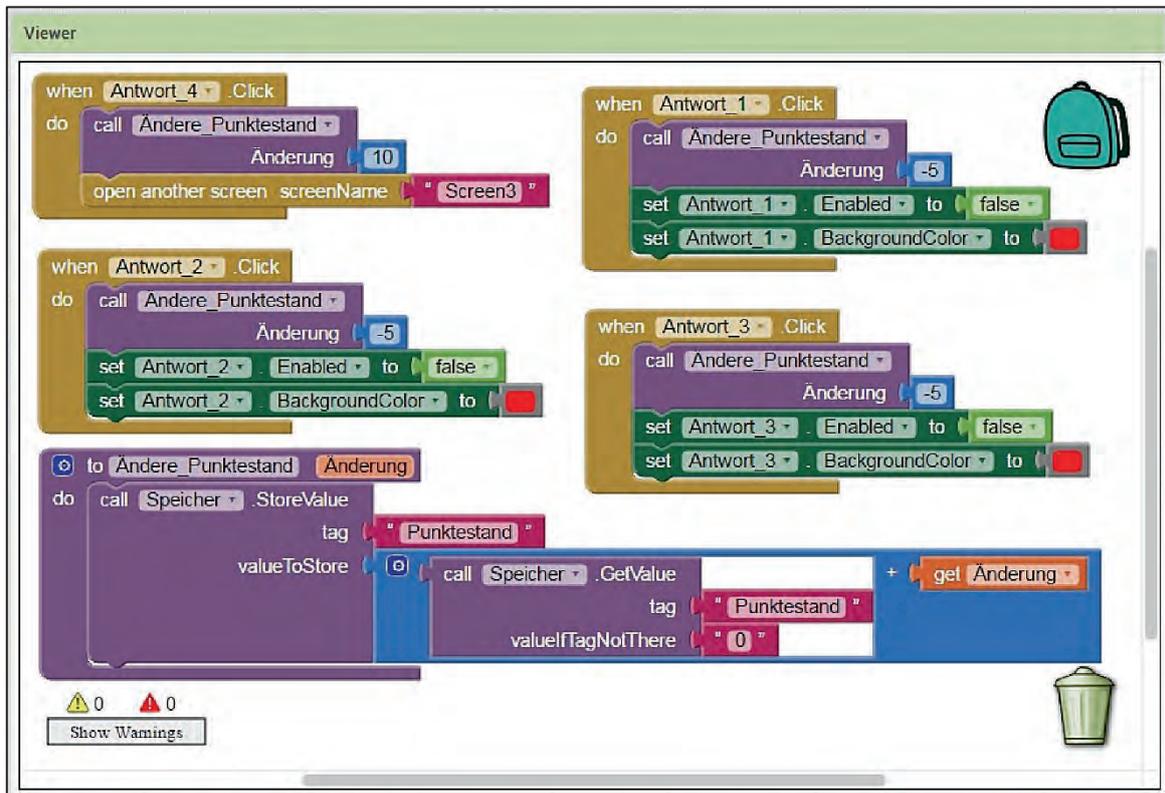
## Schritt 25

Damit der Punkttestand auch immer wieder bei 0 anfängt, sobald man das Quiz beginnt, brauchen wir nochmal drei Bausteine. Der erste Baustein ist *when Screen1.Initialize* und befindet sich unter *Screen1*. Der zweite Baustein ist *call Speicher.ClearTag* und befindet sich unter *Speicher*. Der dritte Baustein ist wieder mal der bereits bekannte *String* Baustein. Verknüpfe diese drei Bausteine wie auf dem folgenden Bild gezeigt.



## Schritt 26

Du bist nun mit *Screen1* fertig und kannst in der *Block-Ansicht* in den *Screen2* wechseln. Wenn du wie in Schritt 24 gesagt, die Bausteine in den Rucksack gezogen hast, kannst du diese Bausteine nun aus dem Rucksack in den Viewer ziehen. Anschließend musst du die entsprechenden Bausteine für den *Screen2* anpassen. Wichtig ist, dass hier *Antwort\_4* für die richtige Antwortmöglichkeit steht. Wenn du fertig bist, dann sollte der Inhalt des *Viewers* ungefähr so wie auf dem folgenden Bild aussehen.



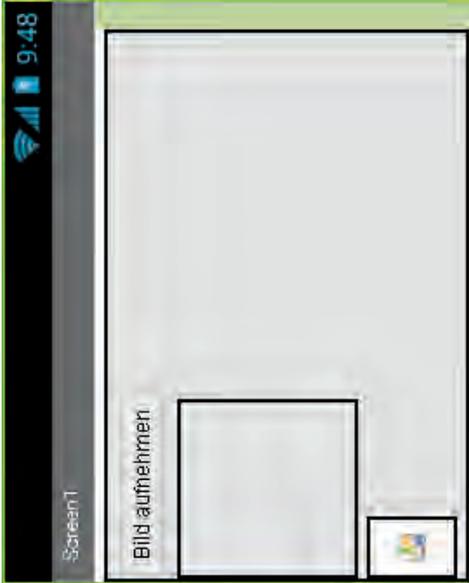
## Schritt 27

Damit bist du auch schon mit *Screen2* fertig und es bleibt nur noch das Anzeigen des Punktestandes auf *Screen3*. Dafür musst du nur das Skript auf dem folgenden Bild nachbauen. Du findest die Bausteine bei den jeweiligen Komponenten *Screen3*, *Punkte* und *Speicher*.



## Ende

Glückwunsch, du bist nun fertig mit deinem Quiz. Wie wäre es mit einem Test? Solltest du bereits die App auf dem Smartphone mittels *AI Companion* getestet haben und nun merken, dass sie sich nicht mit der im App Inventor aktualisiert hat, dann empfiehlt es sich die Verbindung neu aufzubauen. Wähle dazu auf *Reset Connection* im Reiter *Connect* und anschließend wieder auf *AI Companion*.

0	Ein Foto soll es machen	
So kann es aussehen		<p>Das soll passieren</p> <p>Zuerst soll per Knopfdruck mit der Kamera ein Foto geschossen werden. Dieses geschossene Bild soll dann als Hintergrund der Leinwand gesetzt werden.</p> <p>Außerdem soll mit den <i>Screen Arrangements</i> gleich in der Vertikalen ein bisschen für Ordnung sorgen.</p>
Neue Komponenten	<ul style="list-style-type: none"> <li>1 x Button „Bild aufnehmen“</li> <li>1 x Leinwand (Canvas)</li> <li>1 x Kamera (Camera)</li> <li>1 x VerticalArrangement</li> </ul>	<p>Tipps</p> <p>Die benötigten Bausteine kannst du bei den neuen Komponenten (Button, Leinwand, Kamera) finden.</p>

## 0 Ein Foto soll es machen

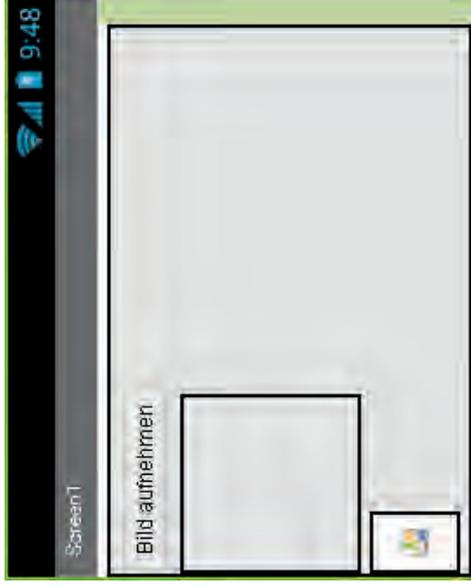
## Lösung



Um auf den get-image-Baustein zugreifen zu können, musst du auf image klicken.

## 1.1 Punkte sind der Anfang

So kann es aussehen



Das soll passieren

Bis jetzt lässt sich mit der App ein Foto schießen und auf der Leinwand (Canvas) anzeigen.

Nun soll aber ein Punkt auf der Leinwand gezeichnet werden, sobald über den Touchscreen die Leinwand berührt wird.

Neue Komponenten

Tipps

Keine

Die benötigten Bausteine befinden sich im *Block Editor* unter *Leinwand*.

## 1.1 Punkte sind der Anfang

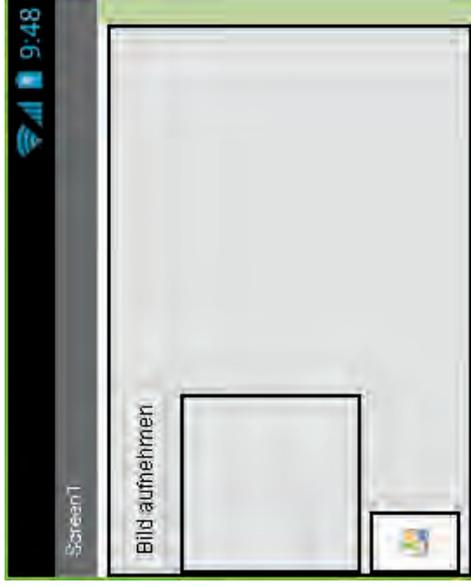
### Lösung



Um auf den `get x`- bzw. `get y`-Baustein zugreifen zu können, musst du auf das `x` bzw. `y` klicken.

## 1.2 Viele Punkte machen einen Strich

So kann es aussehen



Das soll passieren

Nun können auf die Leinwand und damit auf das geschossene Foto Punkte gemalt werden.

Es fehlen nun noch Striche. Dies ist die neue Aufgabe für dich.

Der erste Druck auf den Touchscreen soll den Start des Striches bilden, durch das Ziehen auf dem Touchscreen soll der Strich gezeichnet werden. Lässt man den Touchscreen los, wird der Strich beendet.

Neue Komponenten

Tipps

Keine

Die benötigten Bausteine befinden sich im *Block Editor* unter *Leinwand*.

## 1.2 Viele Punkte machen einen Strich

### Lösung



Verwendet man für  $x1$  und  $y1$  die Bausteine  $startX$  und  $startY$ , dann ergibt sich ein toller Effekt.  
(Wobei dies die gestellte Aufgabe nicht lösen würde ... vielleicht ein Extra für deine App?)

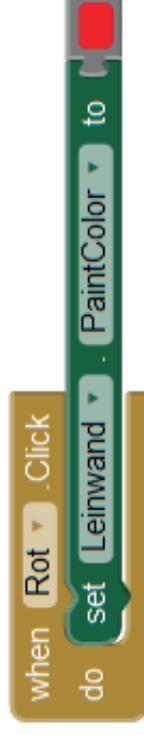
2.1	Es werde bunt!	
So kann es aussehen	Das soll passieren	<p>Nun soll Farbe ins Spiel oder vielmehr in die App kommen.</p> <p>Die Aufgabe lautet, dass sich, wenn du auf einen Button für eine Farbe klickst, die Farbe auf der Leinwand ändert.</p> <p>Für den Anfang wären Schwarz, Blau, Rot und Gelb eine gute Idee, du kannst aber auch andere Farben verwenden. Damit diese schön nebeneinander angeordnet werden können, benötigst du die <i>HorizontalArrangement-Komponente</i>.</p>
Neue Komponenten	Tipps	<p>Sobald du im <i>Block Editor</i> die Programmierung für einen Button fertig hast, kannst du dessen Programmierung ganz einfach kopieren und so vervielfältigen. Klicke dazu auf die Programmierung des Buttons und wähle <i>Duplicate</i> aus. Dann musst du nur noch die Bezeichnung des Buttons und die Farbe anpassen.</p>



## 2.1 Es werde bunt!

### Lösung

Suche im *Block Editor* einen Farb-Button und ziehe den Click-Baustein auf die Arbeitsfläche. Im Auswahlfeld der Leinwand findest du den unteren grünen Baustein, der in die Lücke des Click-Bausteins gehört. Unter *Colors* findest du Bausteine für die Farben. Ziehe die passende Farbe in die hintere Lücke.



## 2.2 Jeder macht mal Fehler ...

So kann es aussehen



Das soll passieren

Es wäre sehr praktisch, wenn man das übermalte Foto wieder säubern könnte.  
Füge dazu einen Button hinzu.

Neue Komponenten

Tipps

1 x Button *Bild säubern*

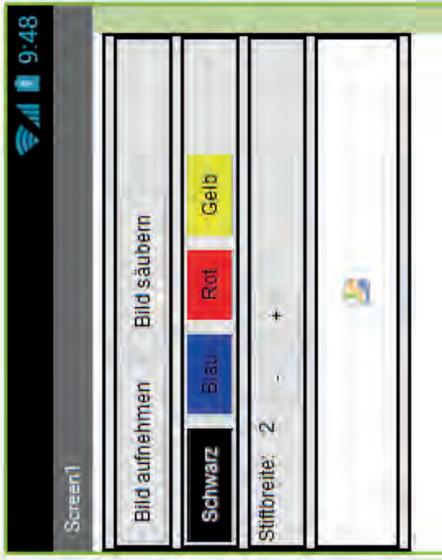
Keine

## 2.2 Jeder macht mal Fehler ...

Lösung



## 2.3 Mal größer und mal kleiner

So kann es aussehen	Das soll passieren
	<p>Dir wird aufgefallen sein, dass die Striche bisher sehr dünn sind.</p> <p>Sorge nun dafür, dass man die Stiftbreite verändern kann. Wenn du auf „+“ klickst, soll die Stiftbreite erhöht werden, durch Klicken auf „-“ verringert.</p> <p>Wenn sich die Stiftbreite verändert, soll der neue Wert auch angezeigt werden.</p> <p>Der Startwert der Stiftbreite soll 2 sein.</p>
Neue Komponenten	Tipps
<p>2 x Label (<i>Stiftbreite</i> und <i>Wert der Stiftbreite</i>)</p> <p>2 x Buttons (+ und -)</p>	<p>Du brauchst eine Variable für die Stiftbreite.</p>

## 2.3 Mal größer und mal kleiner

### Lösung

Damit der Stift (dein Finger) in verschiedenen Stiftbreiten malen kann, musst du ihm zunächst eine Stiftbreite zuweisen. Dazu musst du eine **Variable** anlegen. Gehe dazu im Bereich *Blocks* auf die Kategorie *Variables* und ziehe nebenstehenden Baustein in den Viewer.



Wenn du auf *name* klickst, kannst du den Namen der Variable ändern. Rechts an dem Baustein kannst du den Startwert festlegen. Ändere nun den Namen der Variable auf *Stiftbreite* und hänge den nebenstehenden Baustein (du findest ihn in der *Math*-Kategorie) an die rechte Lücke.



Ändere anschließend den Startwert auf 2.

Um beim Klicken auf den *Plus*-Button die Stiftbreite zu vergrößern, ziehe nebenstehenden Baustein in den Arbeitsbereich.



Anschließend musst du dafür sorgen, dass die Variable *Stiftbreite* um 1 erhöht wird und sich die Stiftbreiten bei der Leinwand und dem Label anpassen. Dafür brauchst du die folgenden Bausteine.

Suche sie zusammen und füge sie in dieser Reihenfolge in den Baustein *when Plus.Click* do ein. Die Programmierung für den *Minus*-Button verläuft entsprechend.



## 2.4 Die Stiftbreite und ihre Grenzen

So kann es aussehen



Das soll passieren

Bisher kann die Stiftbreite beliebig groß werden und sogar kleiner als 1. Das sollte aber nicht so sein.

Um das zu verhindern, benötigst du Bedingungen. Sie legen den Geltungsbe-  
reich für bestimmten Aktionen fest.

Ändere deine App so ab, dass die Stiftbreite nicht kleiner als 1 und nicht größer  
als 6 werden kann.

Neue Komponenten

Tipps

Keine

Du brauchst *if-then*-Bedingungsbausteine.

## 2.4 Die Stiftbreite und ihre Grenzen

### Lösung

Für den *Minus*-Button soll zum Beispiel gelten: Wenn (*if*) die Stiftbreite größer als 1 ist, dann (*then*) wird der Wert der Stiftbreite verringert und die Stiftbreite der Leinwand sowie der Text des Labels aktualisiert. Wenn die Stiftbreite 1 oder kleiner ist, soll dies aber nicht passieren. Um das zu erreichen, braucht man im Block Editor diese Bausteine:

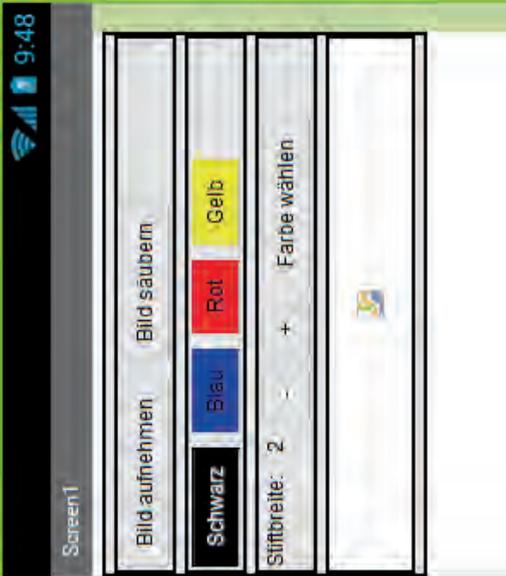


Für den *Plus*-Button benötigst du auch den *if-then*-Baustein, nur das die Bedingung eine andere ist. Damit die Stiftbreite nicht größer als 6 wird, müsste man die Bedingung wie folgt gestalten (natürlich kann man auch einen anderen Wert als 6 festlegen, je nachdem welche Stiftbreite man erlauben möchte):



Füge die Bedingungen für den *Plus*- und *Minus*-Button ein.

## 3.1 Noch mehr Farben! (Teil 1)

<p>So kann es aussehen</p>	<p>Das soll passieren</p> <p>Die Auswahl an Farben soll größer werden. Sorge dafür, dass eine Liste mit Farbnamen erscheint, aus der man die gewünschte Farbe auswählen kann. Dazu brauchst du Folgendes:</p> <ol style="list-style-type: none"> <li>1. Eine Liste mit verschiedenen Namen von Farben.</li> <li>2. In der App muss ein <i>ListPicker</i> angezeigt werden, auf den man klicken kann, um eine Farbe auszuwählen. Außerdem musst du ihm sagen, aus was (den Farben) ausgewählt werden soll. Dazu benötigst du folgenden Baustein:</li> </ol>
	
<p>Neue Komponenten</p> <p>1 x <i>ListPicker</i>: <i>Farbe wählen</i></p> <p><b>Hinweis:</b> Die Buttons zur Farbauswahl kannst du löschen oder zur Schnellauswahl behalten.</p>	<p>Tipps</p> <ul style="list-style-type: none"> <li>• Listen kann man mit dem Baustein <i>make-a-list</i> erstellen.</li> <li>• Wenn du mehr Felder in der Liste brauchst, klicke auf das Zahnrad im blauen Kasten des Bausteins.</li> </ul>

### 3.1 Noch mehr Farben! (Teil 1)

#### Lösung

Schritt 1 sollte folgende Blöcke liefern (es können natürlich auch andere Farben sein):



Schritt 2 sollte folgende Blöcke liefern:



## 3.2 Noch mehr Farben! (Teil 2)

So kann es aussehen



Das soll passieren

Jetzt musst du nun noch dafür sorgen, dass sich auch die Stiftfarbe ändert, wenn man eine Farbe auswählt.

1. Du benötigst zuerst eine weitere Liste. Diesmal soll sie die ausgewählten Farb-Bausteine enthalten.
2. Damit die Farbe der Leinwand richtig gesetzt wird, brauchst du folgenden Baustein:



Aus der oben genannten Liste wird das Element mit dem Index (= Nummer) ausgewählt, den du unten angibst.

Neue Komponenten

Tipps

Keine

- Du musst die Farben in der Liste in der gleichen Reihenfolge anordnen wie die Farbnamen.
- Die Eigenschaft *SelectionIndex* vom *ListPicker* gibt an, welches Element vom Benutzer ausgewählt wurde.

## 3.2 Noch mehr Farben! (Teil 2)

### Lösung

Schritt 1 sollte folgende Blöcke liefern (es können natürlich auch andere Farben sein, sie müssen aber die gleiche Reihenfolge aufweisen wie die Farbnamen):



Schritt 2 sollte folgende Blöcke liefern:



### 3.3 Teile deine App mit der Welt

<p>So kann es aussehen</p>	 <p>Das soll passieren</p> <p>Damit auch deine Freundinnen und Freunde deine Meisterwerke bewundern können, benötigst deine App noch einen Button zum Teilen.</p> <p>Füge einen solchen Button zu deiner App hinzu. Wenn man auf den <i>Teilen</i> Button klickt, soll die Komponente <i>Sharing1</i> aufgerufen werden und eine Datei teilen. Zuerst muss dafür das <i>Canvas</i> (= Leinwand) aber noch unter einem beliebigen Dateinamen (<i>name.jpg</i>) abgespeichert werden.</p>
<p>Neue Komponenten</p> <p>1 x Button <i>Teilen</i></p> <p>1 x <i>Sharing</i></p> 	<p>Tipps</p> <p>Auf deinem Smartphone wird automatisch eine Liste mit verschiedenen Apps aufgerufen, mit denen du das Bild verschicken kannst, beispielsweise WhatsApp und Gmail. Aber auch das Speichern in der Dropbox ist möglich, wenn du die entsprechende App installiert hast. Darum brauchst du dich nicht mehr kümmern.</p>

### 3.3 Teile deine App mit der Welt

Lösung



# BlueCoLight

Das BlueCoLight ist ein kleiner mobiler Controller mit Bluetoothschnittstelle zum Steuern eines LED-Strips und eines kleinen Motors. Die Farben sowie der Motor können dabei manuell oder durch eine App einzeln angesteuert werden. Die Verbindung von App Inventor und BlueCoLight bietet ein großes Potenzial bei der Planung und Umsetzung eigener Projekte.

Für einen ersten Test existiert bereits eine Demo-App. Mittels des QR-Codes auf der rechten Seite kann eine Videoanleitung zur Einrichtung der App auf dem Smartphone abgerufen werden. Mit Hilfe dieser App kann das BlueCoLight gesteuert werden.

Damit die Schülerinnen und Schüler jedoch auch selbst die Möglichkeit haben eine App für ihren BlueCoLight zu schreiben, gibt es ein Beispielprojekt für den App Inventor auf das die Schülerinnen und Schüler zurückgreifen können. Das Beispielprojekt lässt sich in der *Gallery* des App Inventor unter dem Stichwort *BlueCoLight\_Sample* finden.



Die folgenden Bilder zeigen einige Beispiele für mögliche Projekte, die sich mittels App Inventor und BlueCoLight umsetzen lassen:



Selbstgemachter Weihnachtsbaum aus Holz



Bild-Vitrine mit 3D-Effekt



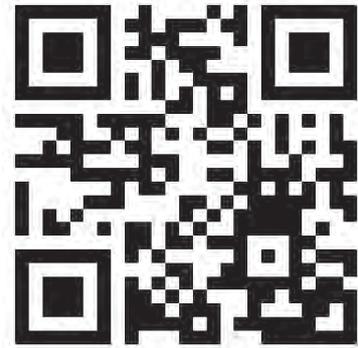
Infinity mirror (Unendlicher Spiegel; Auch sehr gut als Tischfläche zu gestalten)

Quelle: [https://en.wikipedia.org/wiki/Infinity\\_mirror](https://en.wikipedia.org/wiki/Infinity_mirror)

# BlueCoLight

Das BlueCoLight ist ein kleiner mobiler Controller mit Bluetoothschnittstelle zum Steuern eines LED-Strips und eines kleinen Motors. Die Farben sowie der Motor können dabei manuell oder durch eine App einzeln angesteuert werden. Die Verbindung von App Inventor und BlueCoLight bietet ein großes Potenzial bei der Planung und Umsetzung eigener Projekte.

Für einen ersten Test existiert bereits eine Demo-App. Mittels des QR-Codes auf der rechten Seite gelangst du zu einer Video-Anleitung, in welcher beschrieben wird, wie die BlueCoLight App auf deinem Smartphone ausprobiert werden kann. Mit Hilfe dieser App kann das BlueCoLight gesteuert werden.



Die folgenden Bilder zeigen einige Beispiele für mögliche Projekte, die sich mittels App Inventor und BlueCoLight umsetzen lassen:



Selbstgemachter Weihnachtsbaum aus Holz

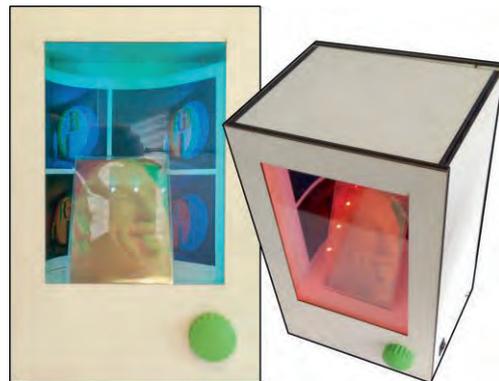


Bild-Vitrine mit 3D-Effekt

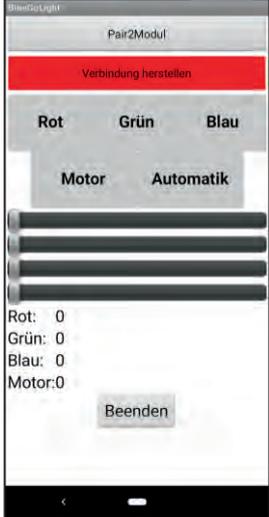


Infinity mirror (Unendlicher Spiegel; Auch sehr gut als Tischfläche zu gestalten)

Quelle: [https://en.wikipedia.org/wiki/Infinity\\_mirror](https://en.wikipedia.org/wiki/Infinity_mirror)

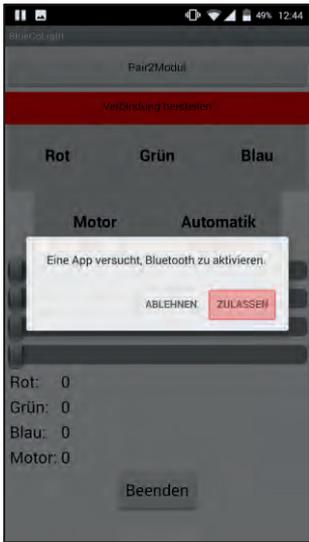
Auf den folgenden Seiten wird erklärt, wie du die App auf deinem Smartphone installieren und testen kannst. Außerdem wird dir auch erklärt, wie du die App selber mit Hilfe des App Inventors umgestalten und erweitern kannst.

## BlueCoLight App auf das Smartphone übertragen und testen

<p><b>Schritt 1:</b> QR-Code scannen</p> 	<p><b>Schritt 2:</b> Der Videoanleitung folgen</p> 	<p><b>Schritt 3:</b> BlueCoLight App öffnen</p> 
--------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------

**Hinweis:** Sofern die App permanent installiert ist, kann diese natürlich auch über das Icon für die BlueCoLight App gestartet werden. Gegebenenfalls musst du diese selbst auf den Homescreen deines Smartphones hinzufügen.



<p><b>Optionaler Schritt 4:</b> Bluetooth aktivieren</p> 	<p><b>Schritt 5:</b> Mit BlueCoLight verbinden</p> 	<p><b>Schritt 6:</b> BlueCoLight auswählen</p> 
--------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------

**Hinweis:** Anfangs kann die Bezeichnung des BlueCoLights noch etwas kryptisch und besteht nur aus Zahlen und Doppelpunkten sein. Wähle hier am besten den Eintrag aus, welcher zuletzt erschienen ist. Bei mehreren Geräten müsst ihr das etwas koordinieren.

### Schritt 7: PIN (1234) eingeben

### Schritt 8: Zurück zum Start der App und mit BlueCoLight verbinden

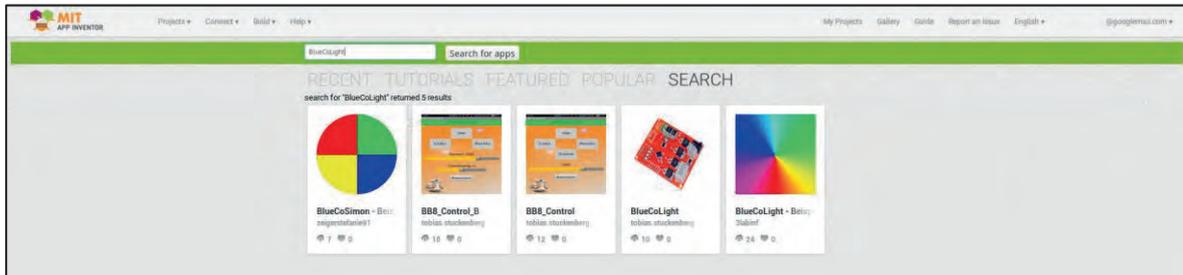
### Schritt 9: Gerät auswählen

### Schritt 10: BlueCoLight mit der App testen

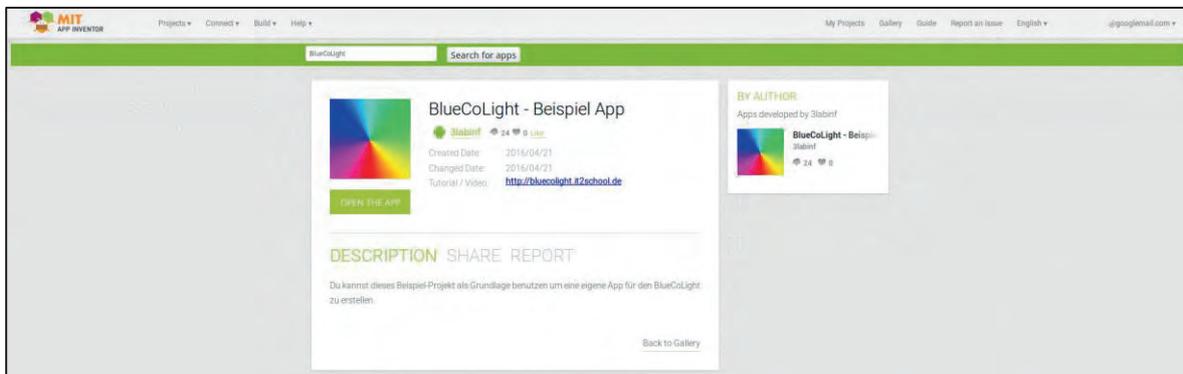
## Eigene App mittels App Inventor erstellen

Um die obige Beispiel-App zum Testen des BlueCoLights umzugestalten oder um eigenen Funktionen zu erweitern, existiert in der Gallery des App Inventors der Quellcode der Beispiel-App.

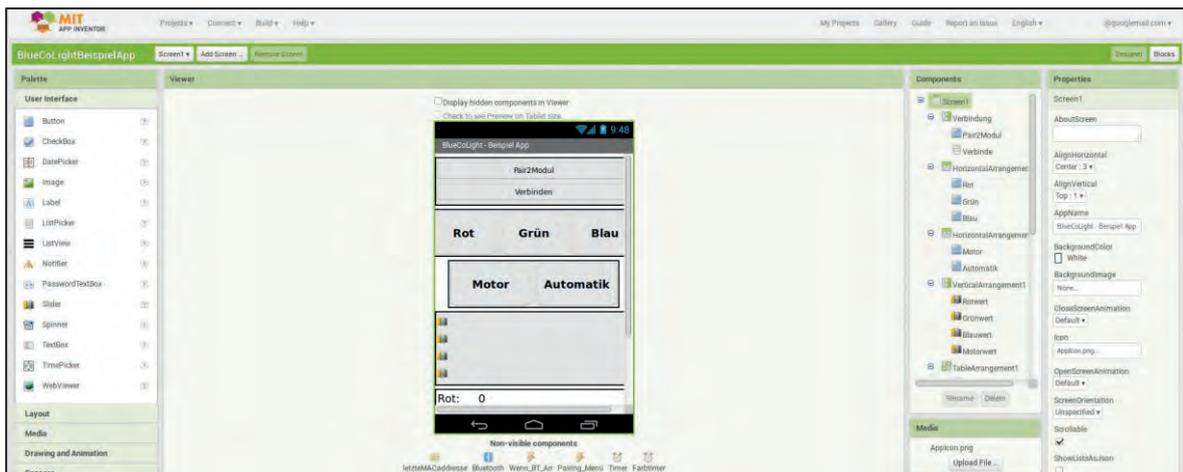
Melde dich dazu einfach im App Inventor an und suche in der Gallery nach BlueCoLight. Wähle dann die App „BlueCoLight – Beispiel App“ von 3labinf aus.



Öffne diese App mit einem Klick auf „Open the App“ und überlege dir einen passenden Projektnamen.

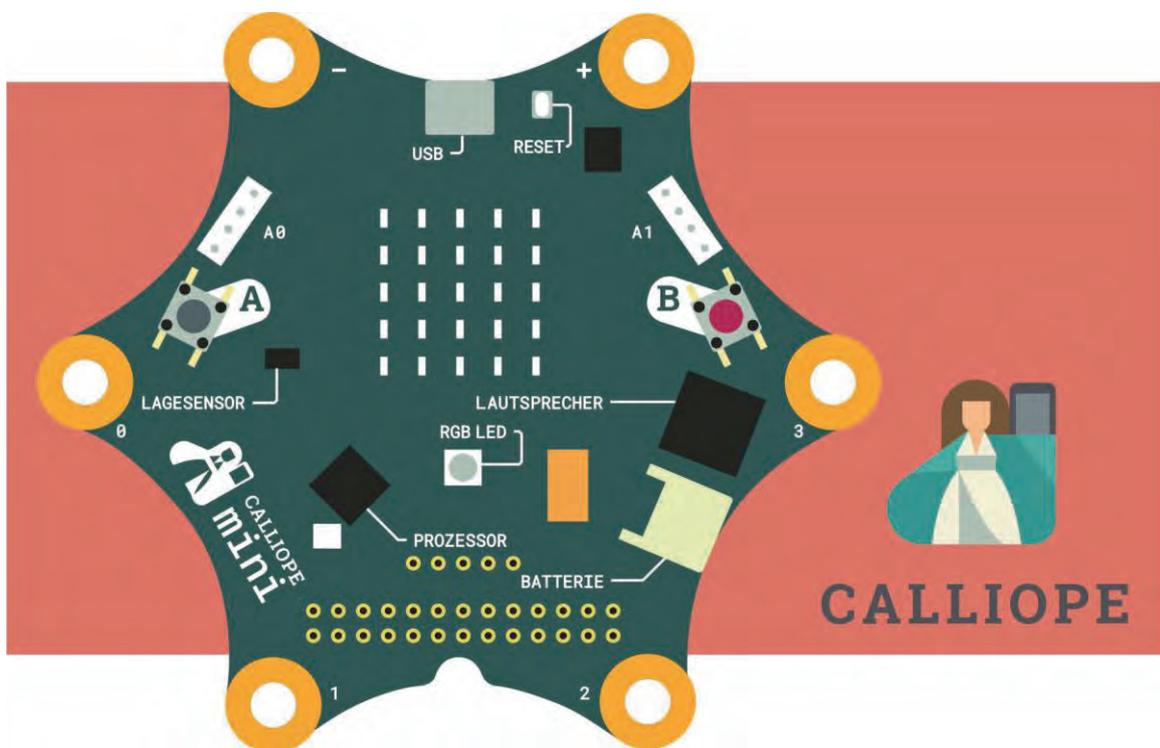


Nun kannst du die bestehende BlueCoLight-App nach deinen Wünschen anpassen.



# IT2School

Gemeinsam IT entdecken



## Modul B8 – Calliope mini

Der Calliope mini Mikrocontroller

Eine Entwicklung von



In Kooperation mit



Im Auftrag der

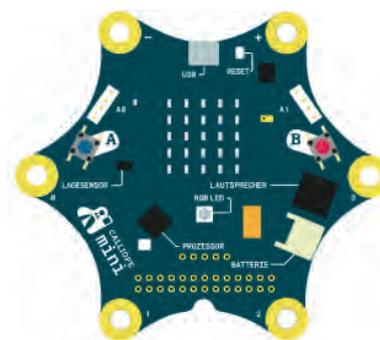


# Inhalt

1	Der Calliope mini.....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Rolle der Unternehmensvertreter*innen.....	4
5	Inhalte des Moduls.....	5
5.1	Aufbau und Anschlüsse des Calliope mini .....	5
5.2	Die Programmierumgebungen .....	7
5.2.1	Webbasierte Programmierumgebungen .....	7
5.2.2	Mobile Programmierumgebungen .....	9
5.3	Der MakeCode-Editor im Detail.....	10
5.4	Open Roberta Lab im Detail.....	11
5.5	Die Programmübertragung .....	12
5.6	Hinweise zu unterschiedlichen Calliope Versionen.....	14
6	Unterrichtliche Umsetzung.....	15
6.1	Grober Unterrichtsplan .....	16
6.2	Stundenverlaufsskizzen.....	17
6.2.1	Variante 1 Grundschule .....	17
6.2.2	Variante 2 Sek I .....	18
7	Einbettung in verschiedene Fächer und Themen .....	20
8	Anschlussthemen.....	21
9	Literatur und Links .....	22
10	Arbeitsmaterialien .....	22
11	Glossar.....	23
12	FAQs und Feedback.....	23

# 1 Der Calliope mini

In diesem Modul erhalten die Schülerinnen und Schüler einen spannenden und spielerischen Zugang zu Computertechnik. Hierfür wurde das Calliope mini Board, gefördert vom Bundesministerium für Wirtschaft und Energie, entwickelt.



Schon Schülerinnen und Schüler der Primarstufe bekommen damit einen Einblick in die Welt der Informatik. Nach ersten kleinen Programmieraufträgen haben die Schülerinnen und Schüler die Möglichkeit für eine alltagsnahe Problemstellung kreative Lösungsansätze mit dem Calliope mini zu entwickeln.

<b>Lernfeld/Cluster:</b>	IT selber machen und teilen
<b>Zielgruppe/Klassenstufe:</b>	X 4. bis 5. Klasse
	X 6. bis 7. Klasse
	X 8. bis 10. Klasse
	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	3 – 8 Stunden
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• Informatik als etwas Kreatives und Gestaltbares erleben</li> <li>• Ein grundlegendes Programmierverständnis entwickeln</li> <li>• Grundlagen der Informationsverarbeitung verstehen (EVA-Prinzip)</li> <li>• Algorithmische Grundbausteine zur Programmierung verwenden</li> <li>• Informatisches Denken auf Problemstellungen aus ihrer Lebenswelt anwenden</li> <li>• Entwerfen, Implementieren und Testen erster eigener Programme</li> </ul>
<b>Vorkenntnisse der Schüler*innen:</b>	<ul style="list-style-type: none"> <li>• Umgang mit der Maus</li> <li>• Speichern von Dateien auf Wechseldatenträgern</li> <li>• Öffnen und Schließen des Browsers</li> <li>• Es ist von Vorteil, wenn das Modul B1 – Blinzeln sowie B5- Programmieren schon einmal durchgeführt wurde</li> </ul>
<b>Vorkenntnisse der Lehrkraft:</b>	Aufgaben des Moduls einmal selbst durchlaufen
<b>Vorkenntnisse der Unternehmensvertreter*innen:</b>	Keine
<b>Sonstige Voraussetzungen:</b>	<ul style="list-style-type: none"> <li>• Internetverbindung</li> <li>• PCs oder Laptops (für jeden oder Kleingruppen)</li> </ul>



## 2 Warum gibt es das Modul?

Im Jahr 2015 wurde in Großbritannien der Kleinstcomputer BBC Micro:Bit vorgestellt und kostenlos an über eine Millionen Schülerinnen und Schüler verteilt. Ziel des Projekts ist es, einen niedrighschwelligigen Zugang zur Informatik zu ermöglichen. Diese Initiative gab in Deutschland den Anstoß, ein ähnliches Projekt umzusetzen – das Ergebnis ist der Calliope mini.

Statt den BBC Micro:Bit zu übernehmen, haben die Entwicklerinnen und Entwickler einige Änderungen vorgenommen, um die Platine noch kindgerechter und mit mehr Möglichkeiten auszustatten. Beispielsweise sind mehr Sensoren integriert und auch Motoren können angeschlossen werden. Daher lassen sich mit dieser Platine zahlreiche Projekte in Schule und Unterricht umsetzen.

Die Calliope-Initiative möchte auch in Deutschland möglichst viele Kinder für Informatik begeistern. Dieses Modul leistet einen Beitrag dazu und stellt Lehr- und Lernmaterialien für Lehrkräfte zusammen, um dieses Projekt erfolgreich umzusetzen und nachhaltig in den Unterricht zu verankern.

Ziel des Moduls ist Lehrkräften sowie Schülerinnen und Schülern – genau wie in England der Micro:Bit - einen niedrighschwelligigen Einstieg zu ermöglichen und Anregungen für interessante Projekte zu geben.

Die Schülerinnen und Schüler lernen dabei nicht nur informatische Grundkompetenzen wie das Programmieren, sondern sie erwerben auch analytische Fähigkeiten, Problemlösekompetenzen sowie durch Gruppenarbeit auch soziale Kompetenzen, wie Teamfähigkeit, Kooperation und Kommunikation.

## 3 Ziele des Moduls

- Informatik als etwas Kreatives und Gestaltbares erleben
- Ein grundlegendes Programmierverständnis entwickeln
- Grundlagen der Informationsverarbeitung verstehen (EVA-Prinzip)
- Algorithmische Grundbausteine zur Programmierung verwenden
- Informatisches Denken auf Problemstellungen aus ihrer Lebenswelt anwenden
- Entwerfen, Implementieren und Testen eigener Programme

## 4 Rolle der Unternehmensvertreter\*innen

Im *Modul B8 – Calliope* hat der\*die Unternehmensvertreter\*in mehrere Möglichkeiten aktiv mitzuwirken. Hier einige Anregungen:

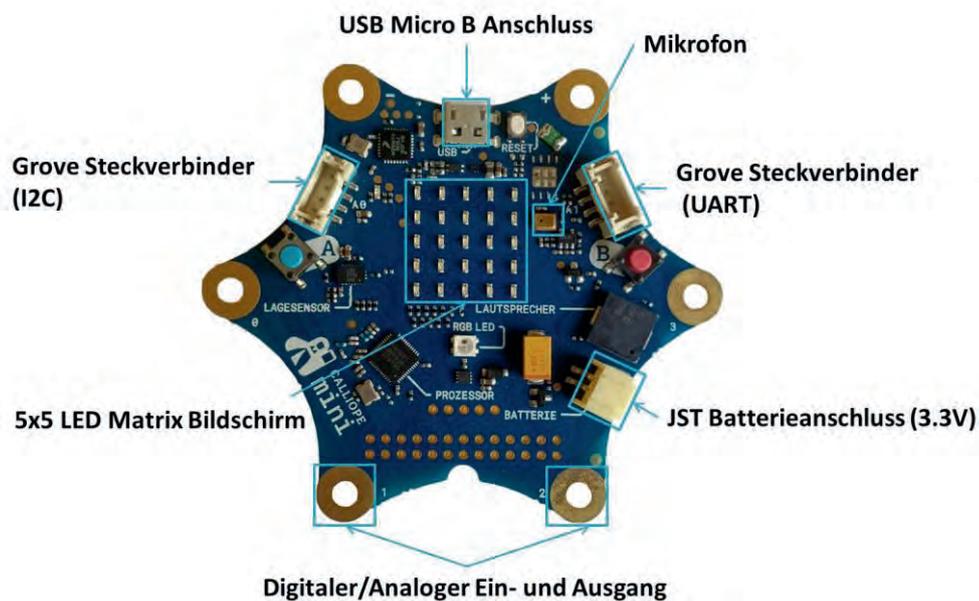
- Co-Teacher - Unterstützung der Lehrkraft beispielsweise bei der Einführung in die Programmier-Oberfläche für den Calliope
- Unterstützung der Schülerinnen und Schüler bei der Umsetzung ihrer eigenen kreativen Projekte

- Sie oder er kann zur Abschlusspräsentation der Projektergebnisse als Special-Guest eingeladen werden.
- Der\*die Unternehmensvertreter\*in besucht Fortbildungen und gibt das Wissen an die Partnerschule weiter

## 5 Inhalte des Moduls

### 5.1 Aufbau und Anschlüsse des Calliope mini

Die Calliope-Platine beinhaltet schon jede Menge Möglichkeiten, um direkt zu starten. Folgende Anschlüsse und Bauteile sind auf dem Board zu finden:



#### Prozessor:

Der nRF51822 Mikrocontroller ist die Schaltzentrale der Platine. In diesem Chip befinden sich unter anderem der Prozessor, der Programm- und Arbeitsspeicher und die Schnittstellen zu zahlreichen Sensoren und Aktoren des Calliope minis. Zudem ermöglicht der Mikrocontroller in Kombination mit der Leiterplattenantenne, die sich unter dem Calliope mini Schriftzug befindet, die Bluetooth-Kommunikation zu anderen Geräten.

#### Spannungsversorgung

Der Calliope mini kann entweder direkt am Computer über ein Micro USB-Kabel (5V) oder mit einem Batterieblock (3V) betrieben werden. Mit Hilfe der oberen beiden, mit (+) und (-) gekennzeichneten Krokodilklemmenanschlüsse, kann man den Calliope mini als Spannungsquelle (3V, max. 100mA Strom) für Projekte nutzen.

<b>Micro-USB</b>	Mit Hilfe des Micro-USB-Anschlusses kann der Calliope mini an den Computer angeschlossen werden, um die eigenen Programme aufzuspielen oder Daten über die serielle Schnittstelle zu übertragen.
<b>LEDs:</b>	Zentral auf dem Calliope mini ist ein 5x5-LED-Raster aufgebracht. Diese leuchten rot und können dazu verwendet werden, eine Laufschrift, Animationen oder kleine Abbildungen anzuzeigen. Diese LEDs werden gleichzeitig zur Messung der Helligkeit verwendet. Direkt darunter befindet sich eine einzelne RGB-LED, die in über 16 Millionen verschiedenen Farben leuchten kann.
<b>Helligkeits-Sensor</b>	Die LEDs der LED-Matrix werden nicht nur zur Anzeige, sondern auch zur Helligkeitsmessung genutzt. Die Lichtintensität umfasst je nach Entwicklungsumgebung den Umfang von 0 (dunkel) bis 255 (sehr hell) [MakeCode] bzw. 0% bis 100% [NEPO].
<b>Taster:</b>	Auf dem Calliope befinden sich zwei Drucktaster. Diese sind blau und rot und mit (A) und (B) beschriftet. Mit einfachen Programmen kann man selbst bestimmen, was passieren soll, wenn man jeweils einen oder beide Taster drückt.
<b>Lautsprecher</b>	Mit Hilfe des Lautsprechers kann man nicht nur einzelne Töne abspielen, sondern ganze Lieder komponieren.
<b>Mikrofon</b>	Dieser Sensor wandelt akustische Signale in elektrische Signale um, die an den Mikrocontroller weitergeleitet werden. Der Mikrocontroller wandelt diese Spannung dann in einen diskreten Wert zwischen 0% und 100% um (je höher, desto lauter).
<b>Motortreiber</b>	Der Motortreiber befindet sich unterhalb der RGB-LED. Dieser ermöglicht es zwei Gleichstrommotoren mit einer Spannung bis max. 11V in einer Richtung anzutreiben.
<b>Lagesensor</b>	Um die Lage des Calliope minis im Raum zu bestimmen, stehen drei Sensoren, die in einem Chip vereint werden, zur Verfügung: ein Beschleunigungssensor, ein Gyroskop und ein Magnetometer.
<b>GPIOs</b>	Im unteren Bereich des Calliope mini befinden sich 26 freie Pins.
<b>Ecken (Input-Output-Anschlüsse)</b>	Die Ecken sind mit (0), (1), (2) und (3) beschriftet. Alle vier Ecken sind berührungsempfindlich. Darüber hinaus können die Ecken (1) und (2) Spannungen messen.
<b>Grove-Steckverbinder</b>	Die Grove-Steckverbinder befinden sich auf der linken und rechten Seite der Platine. An diese können Erweiterungen angeschlossen werden, wie zum Beispiel Farbsensoren oder Infrarot Entfernungssensoren. Eine Einführung in das Grove System finden Sie hier: <a href="http://wiki.seeed.cc/Grove_System">wiki.seeed.cc/Grove_System</a>

## Reset-Taster

Dieser weiße Taster befindet sich unmittelbar neben dem USB-Anschluss. Durch die Betätigung dieses Tasters, wird das aktuelle Programm auf dem Mikrocontroller neu gestartet.

## 5.2 Die Programmierumgebungen

Um den Calliope mini zu programmieren, stehen derzeit diverse Programmierumgebungen zur Verfügung. So gibt es drei browserbasierte Editoren: Mini Edit, Open Roberta Lab und Makecode. Um diese Entwicklungsumgebungen zu nutzen, müssen keine extra Programme auf den Computern installiert werden. Man benötigt aber initial einen Internetzugang.

Zudem gibt es einige App zum Programmieren des Calliope mini. Dazu zählen die Calliope App für Android und iOS und Swift Playgrounds (nur iPad). Der Calliope mini wird via Bluetooth mit den mobilen Endgeräten verbunden und mit Programmen bespielt. Da Swift Playgrounds nur mit iPads kompatibel ist, wird dieser Entwicklungsumgebung nicht mehr näher beschrieben. Es finden sich aber zahlreiche Handreichungen und Material im Internet.

Außerdem gibt es Editoren, die auf dem Computer installiert lokal ausgeführt werden können und daher keinen Internetzugang benötigen. Dazu zählen zum Beispiel microBlocks<sup>1</sup>, abbozza! Calliope<sup>2</sup>, TigerJython4Kids<sup>3</sup> und Segger Embedded Studio<sup>4</sup>. Da sich das Unterrichtsmaterial in diesem Modul nicht auf diese Umgebungen bezieht, werden diese im Folgenden nicht näher beschrieben. In den Fußnoten finden sich aber die entsprechenden Links zu den Entwicklungsumgebungen mit weiterführenden Informationen.

### 5.2.1 Webbasierte Programmierumgebungen

#### Calliope mini Editor (<https://miniedit.calliope.cc>)

Der Editor, der speziell für den Calliope entwickelt wurde, ist eine einfach zu bedienende Oberfläche, die onlinebasiert ist.

Durch den sehr reduzierten Aufbau ist dieser Editor besonders für Grundschülerinnen und Schüler geeignet. Mit wenigen Klicks können sie „wenn-dann-Befehle“ programmieren und auf ihren Calliope mini flashen. Es besteht auch die Möglichkeit, sich den Quellcode anzeigen zu lassen. Für anspruchsvollere Projekte ist der Calliope mini Editor nicht geeignet, da komplexere Programme nicht möglich sind und nicht ein kleiner Teil der Sensorik und Aktuatorik auf der Platine genutzt werden. Daher wird dieser Editor nicht im Arbeitsmaterial berücksichtigt. Außerdem hat der Server hin und wieder Probleme mit der Erreichbarkeit. Aus diesem Grund sollte man sich lieber einer der beiden folgenden Alternativen beschäftigen.

---

<sup>1</sup> <https://microblocks.fun/>

<sup>2</sup> <https://inf-didaktik.rz.uos.de/abbozza/calliope/>

<sup>3</sup> [https://www.tigerjython4kids.ch/index.php?inhalt\\_links=robotik/navigation.inc.php&inhalt\\_mitte=robotik/calliope/loslegen.inc.php](https://www.tigerjython4kids.ch/index.php?inhalt_links=robotik/navigation.inc.php&inhalt_mitte=robotik/calliope/loslegen.inc.php)

<sup>4</sup> <https://www.segger.com/evaluate-our-software/nordic-semiconductor/calliope-mini/>

### **Open Roberta Lab (<https://lab.open-roberta.org/>)**

Auch die Programmierumgebung Open Roberta Lab, entwickelt vom Fraunhofer Institut, wird im Browser geöffnet. Das Programmieren erfolgt im Baukasten-Prinzip und erinnert an die Umgebung von Scratch. Durch diese grafische Programmieroberfläche wird Kindern und Jugendlichen ein einfacher und intuitiver Einstieg ermöglicht. Auch die Lego Mindstorms Systeme, wie EV3 und NXT, lassen sich mit NEPO, der Programmiersprache des Open Roberta Lab programmieren. Daher ist diese Plattform in einigen Schulen schon im Einsatz.

Aufgrund der weiten Verbreitung und der Möglichkeit die Entwicklungsumgebung später noch für andere Robotiksysteme nutzen zu können, gibt es für diesen Editor Arbeitsmaterial. Das Programm kann direkt auf den Calliope übertragen oder aber auch als Simulation auf dem Computer getestet werden.

Bei Bedarf kann dieser Editor auch lokal auf dem eigenen Rechner bzw. für alle Geräte im lokalen Netzwerk zugänglich gemacht werden. Technische Details sind hier verlinkt: <https://github.com/OpenRoberta/openroberta-lab>

### **MakeCode – Editor (<https://makecode.calliope.cc/>)**

Der MakeCode-Editor eignet sich sowohl für Anfänger\*innen-Projekte als auch umfassendere fortgeschrittene Projekte. Der Editor ist auch onlinebasiert und ermöglicht das Programmieren mit Programmier-Blöcken als auch mit JavaScript und Python. Es kann beliebig zwischen den Programmiersprachen gewechselt werden.

Da es sich bei dieser Entwicklungsumgebung um eine HTML5 Web-Anwendung handelt, kann diese auch ohne Internetverbindung verwendet werden, sofern sich diese im Cache des Webbrowsers befindet und mindestens eine Datei bereits kompiliert wurde.

Auch dieser Editor beinhaltet einen Simulator und einen Debug-Modus. Im Debug-Modus können die Programmabläufe langsam und Schritt für Schritt durchgegangen und visualisiert werden. Zudem lassen sich die Werte der verwendeten Variablen ausgeben.

Die Arbeitsblätter zu diesem Modul beziehen sich auch auf diesen Editor, da wie bereits erwähnt, kleine Anfangsprojekte als auch umfassende Projekte für Fortgeschrittene möglich sind und die Kinder und Jugendlichen nicht die Programmierumgebung wechseln müssen, wenn sie an ihre Grenzen stoßen (siehe Mini Edit). Darüber hinaus wurde der MakeCode Editor bereits in England mit dem micro:bit erprobt, sodass zahlreiche Unterrichtseinheiten, Beispiele und Projekte, welche zum Calliope mini kompatibel sind, problemlos adaptiert werden können.

Bei Bedarf kann dieser Editor auch lokal auf dem eigenen Rechner bzw. für alle Geräte im lokalen Netzwerk zugänglich gemacht werden. Technische Details sind hier verlinkt: <https://github.com/microsoft/pxt-calliope>

## 5.2.2 Mobile Programmierumgebungen

### Calliope mini App



<https://play.google.com/store/apps/details?id=cc.calliope.mini&hl=de&gl=US>



<https://apps.apple.com/de/app/calliope-mini/id1309545545>



Neben der Programmierung am PC mit Hilfe eines Webbrowsers, besteht auch die Möglichkeit den Calliope mini mit Hilfe eines mobilen Endgerätes zu programmieren. Die dafür notwendige Calliope App ist sowohl im Play-Store, als auch im App-Store kostenfrei verfügbar (siehe QR-Codes und Links oben).

Die Schülerinnen und Schüler können zum Programmieren aus den drei zuvor vorgestellten Entwicklungsumgebungen wählen. Das Übertragen der fertigen Programme erfolgt nicht über das USB-Kabel, sondern über Bluetooth. Im Vorfeld müssen Calliope mini und Endgerät miteinander gekoppelt werden. Dieser Vorgang wird detailliert unter dem folgenden Link beschrieben:

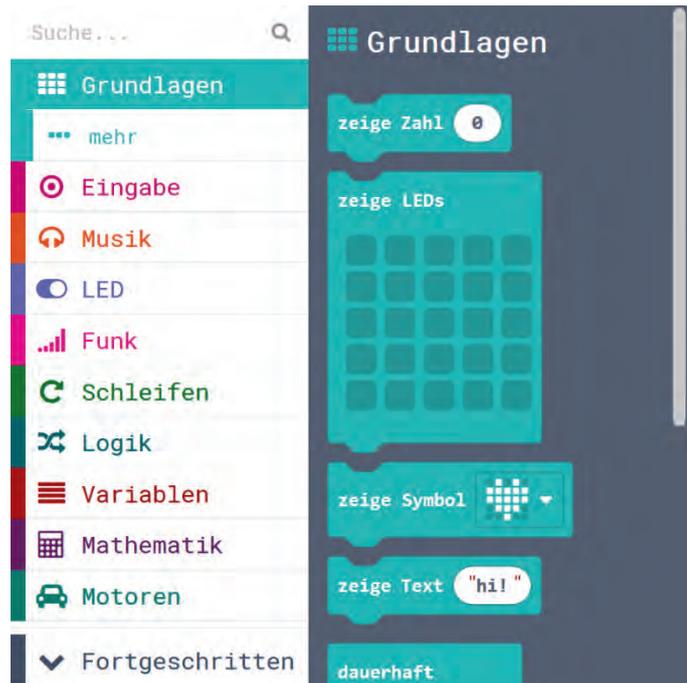
<https://calliope.cc/programmieren/mobil>



### 5.3 Der MakeCode-Editor im Detail

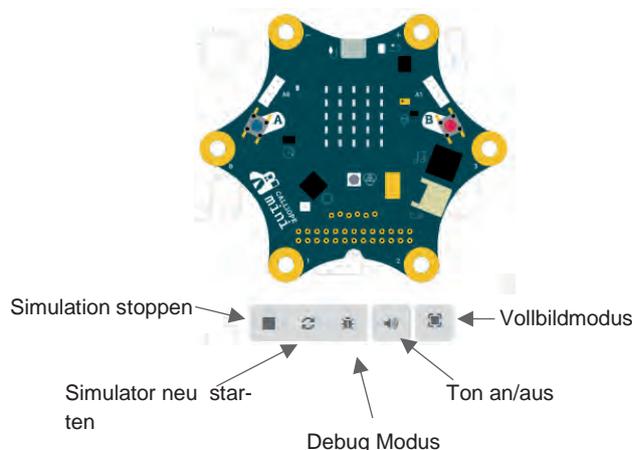
Der MakeCode-Editor bietet zahlreiche Programmierblöcke, die ähnlich wie bei der Programmierumgebung Scratch (siehe Modul B5) farblich und inhaltlich zusammengefasst sind. Dadurch ist eine schnelle Orientierung möglich. Zudem gibt es eine praktische Suchfunktion. Mit einem Klick auf ein Feld (z.B. Grundlagen) öffnen sich die einzelnen Bausteine, die unter der Kategorie zusammengefasst sind.

Die einzelnen Bausteine können per Drag&Dop in das weiße Feld rechts neben den Programmierbausteinen gezogen und wie Puzzleteile zusammengesetzt werden.

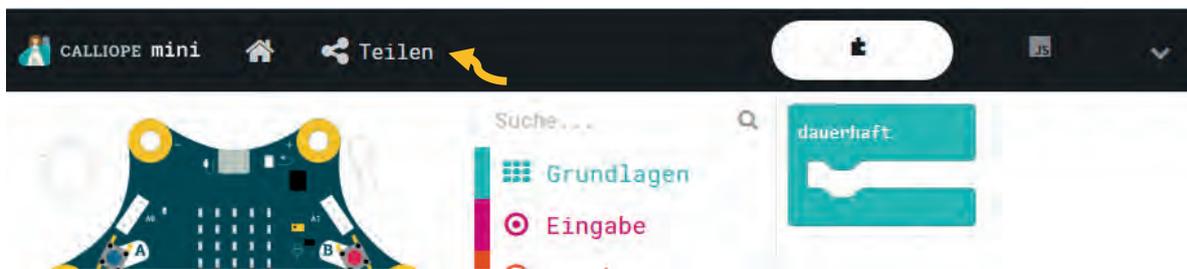


Computer benötigen Computerprogramme mit eindeutigen Anweisungen. Wenn der MakeCode Editor eine Anweisung (bzw. einen Programmierblock) nicht eindeutig zuordnen kann (siehe Beispiel oben), wird der entsprechende Block ausgegraut. Sobald eindeutig ist, wie und wo der Block verwendet werden soll, wird dieser wieder in satten Farben dargestellt.

Bevor man die eigenen Programme auf den Calliope mini überträgt, kann man sie mit Hilfe eines Simulators testen.



Im Debug-Modus kann das Programm Schritt-für-Schritt oder in Zeitlupe ausgeführt werden. Dies ist hilfreich bei der Fehlersuche oder zum Testen von Randbedingungen und Spezialfällen.



Projekte können einfach mit einem Weblink bzw. einen entsprechenden QR-Code zu diesem Link geteilt werden. Nachdem das Programm fertig ist, klickt man einfach auf Teilen und erzeugt den Link.

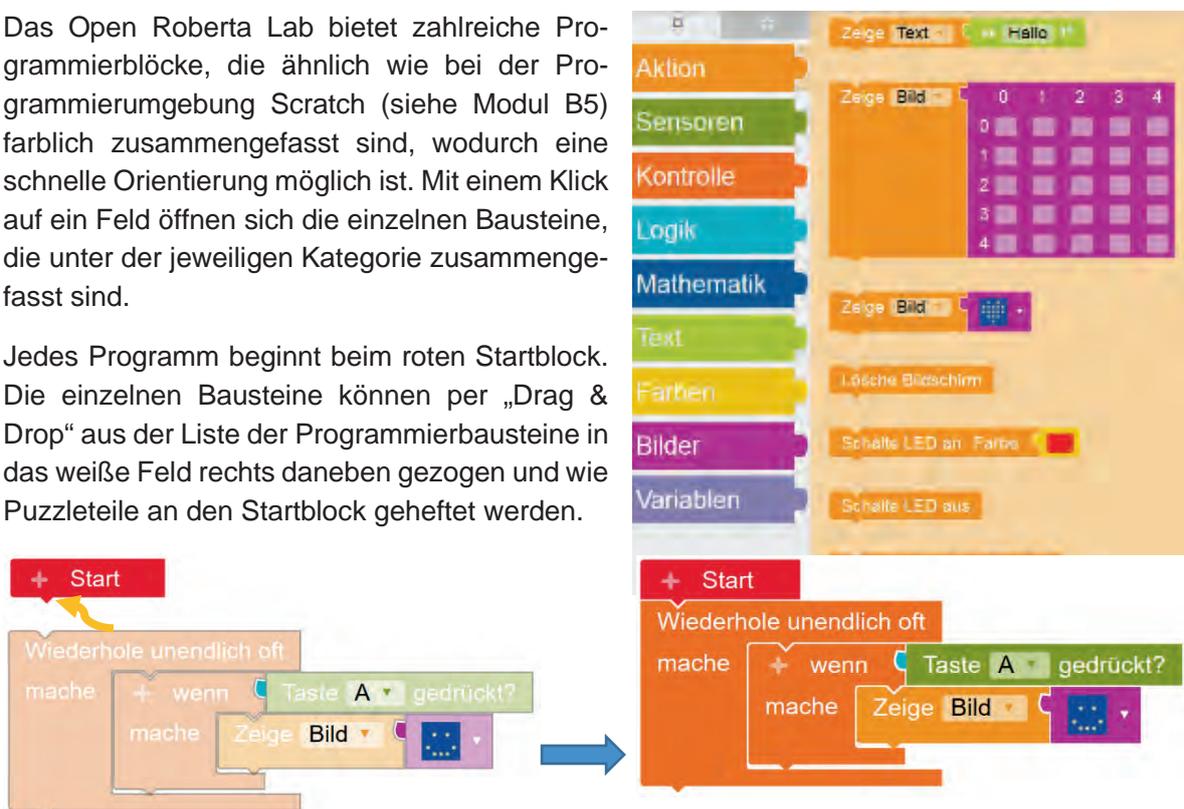
Eine umfangreiche Beschreibung des Editors ist hier verlinkt:

<https://makecode.calliope.cc/blocks>

#### 5.4 Open Roberta Lab im Detail

Das Open Roberta Lab bietet zahlreiche Programmierblöcke, die ähnlich wie bei der Programmierumgebung Scratch (siehe Modul B5) farblich zusammengefasst sind, wodurch eine schnelle Orientierung möglich ist. Mit einem Klick auf ein Feld öffnen sich die einzelnen Bausteine, die unter der jeweiligen Kategorie zusammengefasst sind.

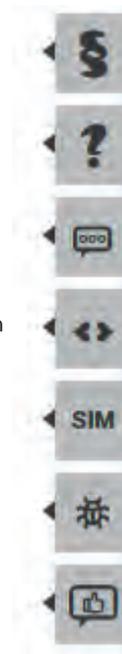
Jedes Programm beginnt beim roten Startblock. Die einzelnen Bausteine können per „Drag & Drop“ aus der Liste der Programmierbausteine in das weiße Feld rechts daneben gezogen und wie Puzzleteile an den Startblock geheftet werden.



Computer benötigen Computerprogramme mit eindeutigen Anweisungen. Wenn der NEPO Editor eine Anweisung (bzw. einen Programmierblock) nicht eindeutig zuordnen kann (siehe Beispiel oben), wird der entsprechende Block ausgegraut. Sobald eindeutig ist, wie und wo der Block verwendet werden soll, wird dieser wieder in satten Farben dargestellt.

Bevor man das eigene Programm auf den Calliope mini überträgt, kann man dieses mithilfe eines Simulators testen. Die Simulation wird durch einen Klick auf das Icon mit der Beschriftung

„Sim“ im rechten Teil des Bildschirms geöffnet (siehe Bild rechts.) Um die Simulation zu starten, klickt man im Simulationsfenster unten rechts auf den „Play“-Knopf.



Hinweis: Vereinzelt wurde mit Problemen bei der Simulation in Verbindung mit dem Safari Browser berichtet. Bei Problemen sollte ein anderer Webbrowser verwendet werden.

Man hat zwei Möglichkeiten Projekte zu teilen. Man klickt oben links im Bildschirm auf das Menü-Icon, wählt „Exportiere Programm“ aus und speichert das Programm als XML-Datei ab. Diese Datei kann dann wieder importiert werden. Alternativ kann auch ein Link erstellt werden, der geteilt werden kann.

Quellcode anzeigen

Simulation starten

Debug-Modus

Eine umfangreiche Beschreibung des Editors ist hier verlinkt:

<https://jira.iais.fraunhofer.de/wiki/display/ORInfo/Programmieren+Calliope+mini>

## 5.5 Die Programmübertragung

Hat man das Programm fertig gestellt, muss dieses auf den Calliope mini übertragen werden. Dieser Vorgang ist für beide Editoren analog

1. **MakeCode:** Zunächst gibt man dem Programm einen beschreibenden Namen, damit man die Datei später noch ihrer Funktion zuordnen kann. Im Anschluss daran kann der grüne Button „Herunterladen“ gedrückt werden.

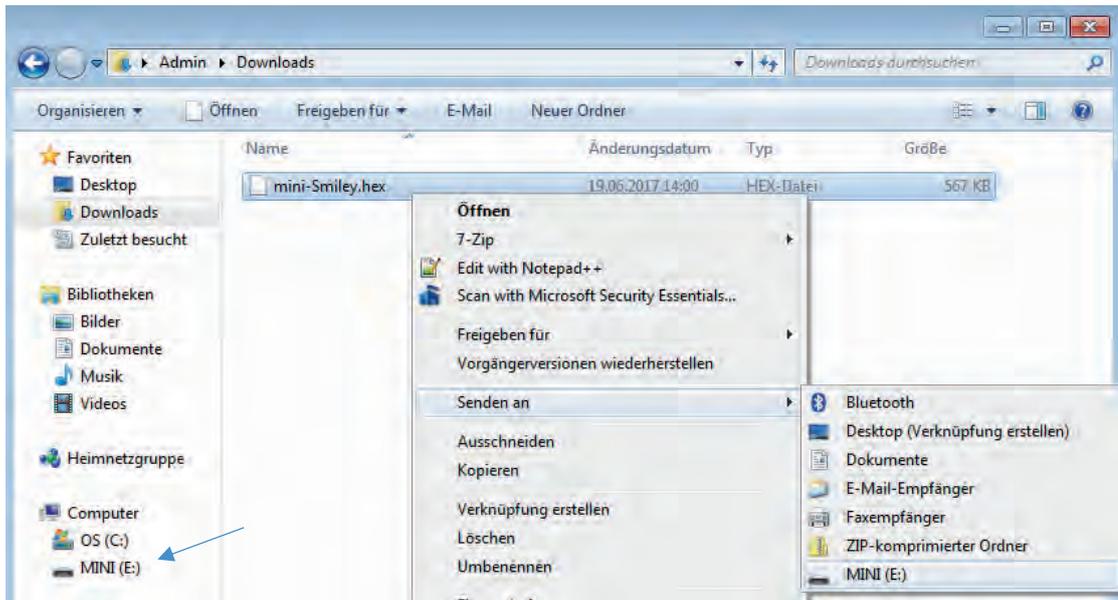


**NEPO:** Sobald das Programm fertig ist, klickt man auf den „Play“-Knopf unten rechts im Bildschirm. Es scheint ein Fenster mit weiteren Anleitungen für den Download.

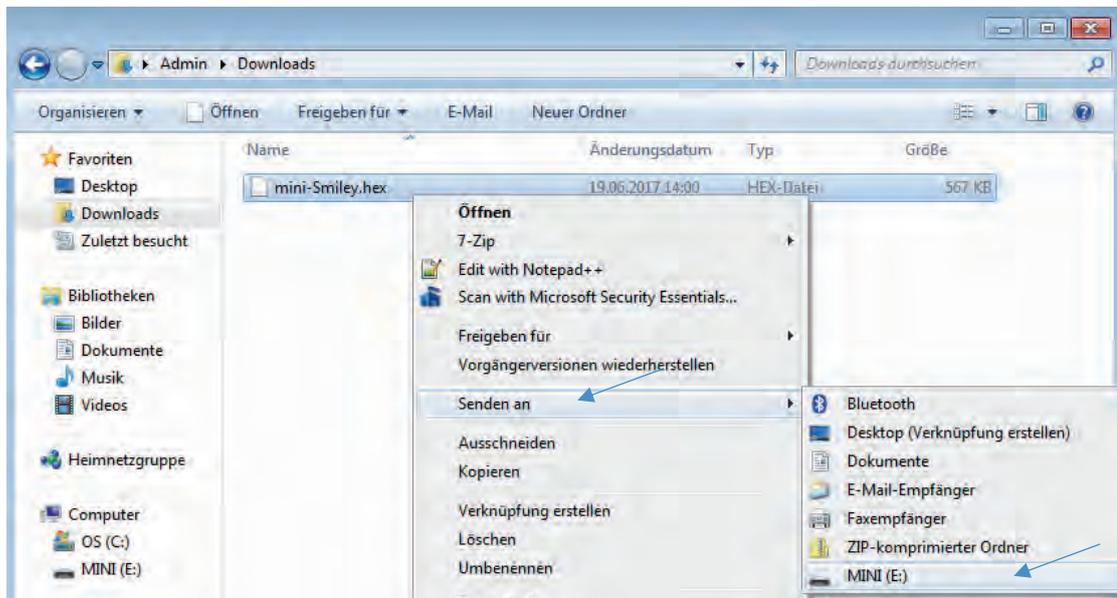


2. Das Programm mit der Endung .hex wird im Download-Verzeichnis gespeichert.  
Tipp: Man sollte darauf achten, dass das Verzeichnis beim Start des Projekts mit den Schülerinnen und Schülern vollständig leer ist, damit die Kinder und Jugendlichen nicht lange nach ihren Dateien suchen müssen.  
Alternativ können Sie auch den Download-Pfad des Webbrowsers ändern, sodass das Programm direkt auf dem mini gespeichert wird.

3. Nach dem Speichern wird der Calliope mini mit Hilfe eines Mico-USB-Kabels mit dem Computer verbunden. Der Calliope sollte dann wie ein gewöhnliches Laufwerk mit dem Namen „MINI“ erscheinen.



4. Durch Betätigung der rechten Maustaste auf der Datei öffnet sich das Kontextmenü. In diesem wählen Sie „Senden an“ aus, um in das nächste Kontextmenü zu gelangen. Durch das Klicken der linken Maustaste auf das Laufwerk MINI (E:) wird die Datei dann auf den mini kopiert. Alternativ kann die Datei auch per Drag&Drop direkt auf den mini gezogen werden. Während des Kopiervorgangs, blinkt die gelbe Status-LED mehrfach auf. Sobald die Übertragung abgeschlossen ist, verbindet sich der Calliope mini erneut mit dem Rechner und die gelbe Status-LED leuchtet wieder stetig.



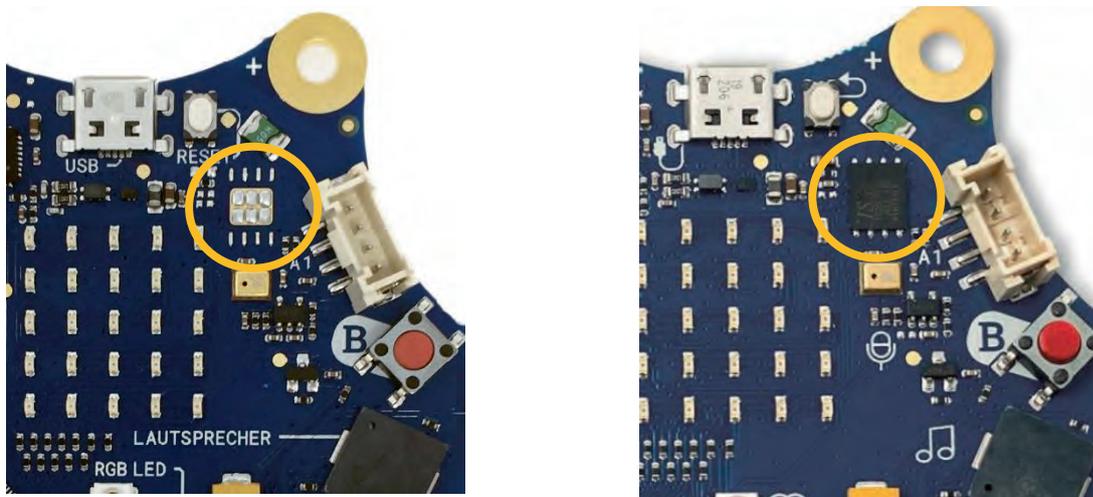
5. Je nach Version (siehe Hinweise in 5.6) wird sofort oder durch Drücken des RESET-Knopfes das Programm auf dem Calliope mini ausgeführt. Das Programm ist nun auf

dem Calliope gespeichert. Bei Bedarf kann man nun auch die Verbindung zum Computer trennen und den mini mit Hilfe der Batteriebox mit Spannung versorgen.

## 5.6 Hinweise zu unterschiedlichen Calliope Versionen

Mittlerweile sind verschiedene Calliope Versionen veröffentlicht worden, die sich nicht nur optisch unterscheiden. Auf der Rückseite, neben dem USB-Anschluss, befindet sich die Versionsnummer. So gibt es die Version 0.3 (Vorgestellt beim IT-Gipfel und verfügbar für Pilotierungen), Version 1.0 (erstes, offizielles Release), Version 1.3 (Updates im Schaltplan: Low Energy Mode deaktiviert, Mikrofonschaltung verstärkt, Stromversorgung trennt nun vollständig, wenn Batterie und USB angeschlossen sind) und die aktuelle Version 2.0.

Datenblätter mit den technischen Details sind hier verlinkt: <https://calliope-mini.github.io/v20/>



Oben im Bild sieht man einen Ausschnitt der Versionen 1.3 und 2.0. Auf der neuen Version befindet sich zusätzlich ein 128MB großer Speicherbaustein, der es ermöglicht bis zur 25 Programme auf dem Calliope zu speichern. Wenn man einen Calliope 2.0 am Computer anschließt, werden zwei Laufwerke erkannt. Auf dem Laufwerk MINI kann man die Programme, wie zuvor auch, herüber kopieren.

Im Laufwerk FLASH befinden sich die gespeicherten Programme. Damit ein Programm aufgerufen werden kann, **muss** der Dateiname in Großbuchstaben geschrieben werden und darf nicht länger als acht Buchstaben lang sein.

Um die gespeicherten Programme zu starten, muss der RESET-Taster (neben USB-Stecker) etwa fünf Sekunden lang gedrückt werden. Das Display leuchtet dann kurz komplett auf und dann ist nur noch ein einzige rote LED zu sehen. Mit den Tastern A und B kann nun durch die Programme navigiert werden. Die erste LED oben rechts steht für das erste Programm, die LED rechts daneben für das zweite Programm und so weiter. Drückt man nun Taster A und Taster B gleichzeitig, wird das entsprechende Programm ausgewählt und auf den Mikrocontroller übertragen. Nach einem kurzen Augenblick ist der Vorgang abgeschlossen. Weitere Hinweise zum neuen Speicherbaustein sind hier verlinkt: <https://calliope.cc/start/tipps>



## 6 Unterrichtliche Umsetzung

Dieses Modul liefert eine einfache Grundschulvariante zum Einstieg, sowie ein Tutorial und ein tiefergehendes Projekt für die Sek I (5.-7. Klasse).

In der 3.- und 4. Klasse empfiehlt sich mit einer Programmereinheit ohne Computer zu beginnen. Insbesondere für Kinder, die das erste Mal programmieren, kann es hilfreich sein zu verstehen, was Programmieren überhaupt bedeutet. Zu diesem Zweck kann das Programmieren mit Befehlen in der Turnhalle, Aula oder auch Klassenraum demonstriert werden. Mit Hilfe des B 8.1 kann die Lehrkraft und auch andere Schülerinnen und Schüler im Klassenraum programmiert werden. Dabei werden auch schon die ersten Programmbausteine für den Calliope mini verwendet.

Wurde zuvor das Modul *B5 - Programmieren* und *B6 - Mein Anschluss* durchgeführt, kann dieser Einstieg auch weggelassen werden. Die Schülerinnen und Schüler erhalten in diesen beiden Modulen einen Einblick in die Programmierung mit Bausteinen.

Im nächsten Schritt lernen die Kinder den Calliope mini und dessen Programmier-Editor kennen. Gemeinsam wird der Calliope erkundet und eine kleine Aufgabe zur Einführung gemeinsam umgesetzt, sowie erste kleine Programmieraufträge gegeben.

Danach können die Schülerinnen und Schüler eine erste eigene Programmierung entwickeln. Dafür erhalten sie eine lebensweltorientierte Problemstellung zu der sie kreative Lösungsansätze entwickeln können.

Auch für die Sekundarstufe bietet es sich an, zuvor die Module *B5 - Programmieren*, *B6 - Mein Anschluss* oder auch *B7 - Meine App* durchzuführen. Die Schülerinnen und Schüler erhalten darin einen Einblick in die Programmierung mit Bausteinen.

Zu Beginn der Einheit mit dem Calliope mini können die Schülerinnen und Schüler ein Tutorial durcharbeiten, bei dem sie angeleitet werden, den Mikrocontroller selbständig zu erkunden und einzelnen Bausteine auszuprobieren. Im zweiten Tutorial können die Schülerinnen und Schüler ein Spiel nach Anleitung programmieren. Hierdurch lernen sie auch Bausteine für Fortgeschrittene kennen.

Zum Abschluss erhalten die Schülerinnen und Schüler verschiedene Programmieraufträge mit einer lebensweltorientierten Problemstellung. Die unterschiedlichen Programmieraufträge können auch für die Binnendifferenzierung genutzt werden.

## 6.1 Grober Unterrichtsplan

### Variante 1 – Grundschule

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Was heißt „programmieren“? Robo-Rallye zum Einstieg (dieser Einstieg ist für Kinder ohne Programmiererfahrung)
Einstieg	Vorstellung des Calliope mini und Kennenlernen des Editors
Vertiefung	Erste Programmieraufträge zum Kennenlernen der Programmieroberfläche
Vertiefung	Eigenständige Entwicklung und Umsetzung einer Idee
Präsentation	Präsentation der Ergebnisse

### Variante 2 – Sek I

Unterrichtsszenarien	Kurze Zusammenfassung
Einstieg	Vorstellung des Calliope mini, erstes Ausprobieren
Einstieg	Kennenlernen der Programmierumgebung MakeCode/Open Roberta Lab mit Hilfe eines Tutorials
Vertiefung	Umsetzung eines Programmierauftrages
Präsentation	Präsentation der Ergebnisse
Vertiefung	Umsetzung einer eigenen Idee

## 6.2 Stundenverlaufsskizzen

### 6.2.1 Variante 1 Grundschule

Variante 1 ist für die Grundschule geeignet ist. Dafür werden etwa 4-6 Stunden benötigt, je nach Einstieg.

#### Abkürzungen/Legende

AB = Arbeitsblatt/Arbeitsblätter; L = Lehrkraft; MuM = Mitschülerinnen und Mitschüler; SuS = Schülerinnen und Schüler; UV = Unternehmensvertreter\*in

#### Einführung

Zeit	Phase	Sozialform/ Impuls	Inhalt/Unterrichtsgeschehen	Material
45 Min.	Einstieg	Einstieg, Gruppen- oder Tandemarbeit	Analoger Einstieg für SuS ohne Programmierkenntnisse: Was heißt programmieren? (haben die SuS Vorerfahrungen mit Modul B5 – Programmieren kann dieser Einstieg auch weggelassen werden) Falls die Kinder Probleme mit dem Pfeifen haben, kann auch eine Trillerpfeife oder Triangel verwendet werden.	B8.1 Blatt mit A und B drauf (z.B. Post-It) Ggf. Trillerpfeife oder Triangel
15 Min.	Hinführung	Plenum, Unterrichts- gespräch	Der Calliope mini wird vorgestellt; die SuS erkunden den Calliope mini: <ul style="list-style-type: none"> <li>• Was ist darauf zu erkennen?</li> <li>• Kennt ihr die Begriffe auf der Platine?</li> <li>• Wo kann man die Energieversorgung anschließen?</li> </ul>	B8.2 Calliope minis (je nach Gruppenanzahl)
30 Min.	Erarbeitung	Plenum bzw. Gruppen- arbeit	Vorstellung der Programmieroberfläche ( <a href="https://makecode.calliope.cc/">https://makecode.calliope.cc/</a> oder <a href="https://lab.open-roberta.org/">https://lab.open-roberta.org/</a> ) <ul style="list-style-type: none"> <li>• Wie wird das Programm aufgerufen und wie ist es aufgebaut?</li> <li>• Wie werden Programme geschrieben?</li> <li>• Welche Programmbausteine sind bekannt?</li> <li>• Wie wird das Programm auf den Calliope übertragen?</li> </ul>	B8.2 Beamer, PC oder Lap- tops für jede Gruppe)

			Die Vorstellung sollte in der Grundschule idealerweise im Plenum stattfinden, L zeigt die einzelnen Komponenten über den Beamer, AB 8.2 dient zur Hilfestellung	
45 Min.	Sicherung	Gruppenarbeit	SuS erhalten B8.3, arbeiten es durch und lösen die darin enthaltenen Aufgaben, sie lernen dadurch die verschiedenen Programmierbausteine kennen, um die nachfolgende Aufgabe zu lösen	B8.3
60 min.	Transfer	Gruppenarbeit	SuS erhalten B8.4 „Hanna muss ins Krankenhaus“; sie entwickeln eigene Lösungsideen und setzen diese um, folgende Problemstellungen und damit mögliche Lösungsideen können aus dem Text abgeleitet werden: <ul style="list-style-type: none"> <li>• Etwas gegen Langeweile</li> <li>• Eine Alarmanlage für das Schaf</li> <li>• Zum Antworten, wenn Hanna das Sprechen schwerfällt</li> <li>• Zum Fiebermessen</li> </ul>	B8.4
20-30 min.	Präsentation	Plenum	Die SuS präsentieren ihre Ergebnisse in der Klasse, dabei wird auch die Programmierung mit Hilfe eines Beamers gezeigt	PC, Beamer

### 6.2.2 Variante 2 Sek I

Zeit	Phase	Sozialform/Impuls	Inhalt/Unterrichtsgeschehen	Material
10 Min.	Einstieg	Plenum, Unterrichtsgespräch	Vorstellung des Calliope mini, L führt kurz in die Entwicklungsumgebung ein ( <a href="https://makecode.calliope.cc">https://makecode.calliope.cc</a> oder <a href="https://lab.open-roberta.org/">https://lab.open-roberta.org/</a> ) Folgende Fragestellungen sollten dabei beantwortet werden: <ul style="list-style-type: none"> <li>• Was ist der Calliope mini?</li> <li>• Wie wird die Programmierumgebung aufgerufen?</li> <li>• Wie können Programme geschrieben werden?</li> </ul>	

20 min.	Einstieg		<ul style="list-style-type: none"> <li>Wie können die Programme auf den Calliope mini übertragen werden?</li> </ul>	
120 Min.	Erarbeitung	Tandemarbeit Gruppenaufgabe	<p>Selbständiges Ausprobieren der Programmierbausteine</p> <p>Die SuS erhalten B8.6 mit unterschiedlichen Programmieraufträgen</p> <ul style="list-style-type: none"> <li>Alarmanlage (2 Varianten: leicht und schwer) (hierfür können auch weitere Materialien bereitgestellt werden z.B. Krokodilklemmen)</li> <li>Schrittzähler</li> <li>Eigene Idee (je nach Idee)</li> </ul> <p>Neben der Programmierung sollen sich die SuS auch ein geeignetes Aussehen überlegen. Dafür werden ggf. Bastelmaterialien benötigt (z.B. Wie würde man den Schrittzähler am Körper tragen)</p>	B8.5 B8.6 Schere, Kleber Pappe, farbige Papiere, Schnur, Krokodilklemmen, Alufolie, ...
	Präsentation	Plenum	Die SuS präsentieren ihre Ergebnisse in der Klasse, dabei wird auch die Programmierung mit Hilfe eines Beamers gezeigt	
120	Ggf. Vertiefung	Gruppenarbeit	Umsetzung eines eigenen Projekts	

## 7 Einbettung in verschiedene Fächer und Themen

Für den Bereich der digitalen Bildung gibt es bisher noch keine konkreten Kompetenzformulierungen für die Grundschule. Die Kultusministerkonferenz hat allerdings ihre Empfehlungen zur „Medienbildung in der Schule“ im Jahr 2016 präzisiert und Anforderungen für eine Bildung in der digitalen Welt formuliert.

Die Länder sollen dafür Sorge tragen, dass alle Schülerinnen und Schüler, die zum Schuljahr 2018/19 eingeschult werden oder in die Sek I eintreten die Kompetenzen in den Bereichen

- Suchen, Verarbeiten und Aufbewahren
- Kommunizieren und Kooperieren
- Produzieren und Präsentieren
- Schützen und sicher agieren
- Problemlösen und Handeln sowie
- Analysieren und Reflektieren

erwerben können (nähere Informationen siehe hier: <https://www.kmk.org/aktuelles/thema-2016-bildung-in-der-digitalen-welt.html>)

Durch die Bezüge zu unterschiedlichen Fächern kann das gesamte Modul oder können einzelne Teile in verschiedenen Fächern eingesetzt werden. Die folgenden Kompetenzen finden sich entweder in dem neuen Strategiepapier der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

### **Kompetenzbereich: Problemlösen und Handeln**

- Anforderungen an digitale Umgebungen formulieren
- Technische Probleme identifizieren
- Bedarfe für Lösungen ermitteln und Lösungen finden bzw. Lösungsstrategien entwickeln
- Anforderungen an digitale Werkzeuge formulieren
- Digitale Umgebungen und Werkzeuge zum persönlichen Gebrauch anpassen
- Funktionsweisen und grundlegende Prinzipien der digitalen Welt kennen und verstehen
- Algorithmische Strukturen in genutzten digitalen Tools erkennen und formulieren

### **Sachunterricht (GS)**

Die Schülerinnen und Schüler ...

- arbeiten am PC mit Textverarbeitungs-, Lern- und Übungsprogrammen
- erproben unterschiedliche Lösungen für technische Problemstellungen

## Informatik (Sek I)

Schülerinnen und Schüler ...

- benutzen die algorithmischen Grundbausteine zur Darstellung von Handlungsvorschriften
- entwerfen und testen einfache Algorithmen
- können erdachte Systeme in technische Systeme übertragen
- kennen sich in Entwicklungsumgebungen/Programmierungsumgebungen aus

## 8 Anschluss Themen

Als Anschluss Themen im Zusammenhang mit IT2School bieten sich folgende Bausteine an:

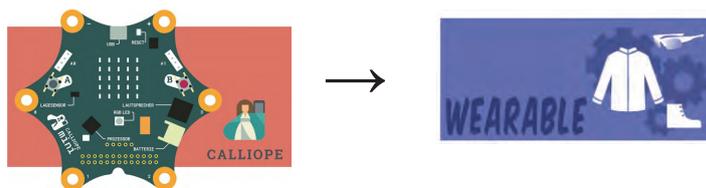
### Beispiel: Programmieren

Sollten Sie mit dem Calliope begonnen haben, können Sie auch zurück zum Modul B5 – Programmieren und B6 – Mein Anschluss springen:



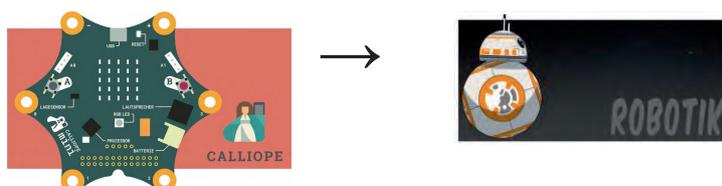
### Beispiel: Programmieren II

Um die Thematik „Wearables“ zu vertiefen kann das Modul E2 – Wearables herangezogen werden. Die Schülerinnen und Schüler erfahren etwas über tragbare und interaktive Systeme, wie sie in smarten Kleidungsstücken Anwendung finden.



### Beispiel: Robotik

Im Erweiterungsmodul E3 Robotik haben die Schülerinnen und Schüler die Möglichkeit einen eigenen BB8 (ein Roboter aus dem Star Wars Universum) zu bauen. Bei diesem Bauvorhaben kann auf einen Arduino oder das BluCoLight zurückgegriffen werden.



## 9 Literatur und Links

- Calliope – **Offizielle Webseite:** <https://calliope.cc>
- **MakeCode Editor:** <https://makecode.calliope.cc/>
- **Open Roberta Lab Editor** <https://lab.open-roberta.org/>
- **Calliope mini technische Dokumentation:** <https://calliope-mini.github.io/v10/>
- **Projektbeispiele:** <https://calliope.cc/projekte>
- Haase, Hans et al. (2017): **Calliope Lehrer-Handreichung** der Universität Wuppertal: [https://ddi.uni-wuppertal.de/www-madin//Calliope\\_Handreichung.pdf](https://ddi.uni-wuppertal.de/www-madin//Calliope_Handreichung.pdf)
- Bergner, Nadine et al. (2017): **Das Calliope Buch** - Spannende Bastelprojekte mit dem Calliope-Mini-Board. dpunkt.verlag
- **Programmieren ohne Computer:** <https://kinder-geben-kommandos.de/2014/12/06/programmieren-ohne-computer-ein-experiment-zur-hour-of-code/>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B8.1 GS	Die Robo-Lehrkraft	Programmierung der Lehrkraft und der Mitschülerinnen und Mitschüler, ohne Computer und ohne Calliope
 B8.2 GS	Der Calliope mini	Vorstellung des Calliope mini, Anleitung zur Übertragung der Programme auf den Calliope
 B8.3 GS	Kleine Programmieraufträge	Kleine Aufgaben zum Kennenlernen der einzelnen Bausteine
 B8.4 GS	Hanna muss ins Krankenhaus	Arbeitsauftrag zum Programmieren verschiedener Anwendungen
 B8.5 Sek	Tutorial	Einführung in den Calliope
 B8.6 Sek	Die Klasse 7a auf Klassenfahrt	Arbeitsauftrag zum Programmieren verschiedener Anwendungen
 B8 Muster	Musterlösung	Musterlösungen für die Programmieraufträge aus B8.2GS, B8.4GS und B8.6 Sek.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Calliope/ Kalliope	Eine der neun klassischen Musen, Göttin der Künste und Wissenschaften, Muse der epischen Dichtung und Namensgeberin für Calliope mini
Mikrocontroller	Ein-Chip-Computersystem: Alle Bauteile, die ein Computer benötigt, wie z.B. Prozessor, Taktgeber, Programm-/Arbeitsspeicher und Bussysteme, sind in einem einzigen Chip integriert.
Magnetometer	Sensor zur Messung der magnetischen Flussdichte
UART	Kommunikationsprotokoll zur seriellen Datenübertragung
I2C	Master-Slave Bussystem zur seriellen Datenübertragung
Entwicklungsumgebung	Computerprogramme zum Erstellen von Computerprogrammen
Flashen	Übertragen eines Programms auf den Programmspeicher eines Mikrocontrollers

## 12 FAQs und Feedback

Stolpersteine, Lessons learnt und Frequently Asked Questions (FAQs) finden Sie unter:



<https://tinyurl.com/IT2S-FAQ>

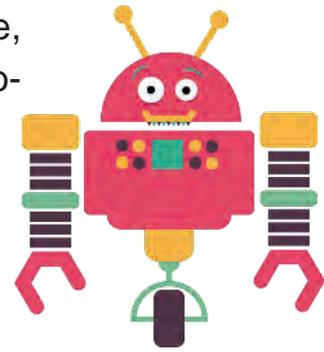
Wir sind auf Ihr Feedback zum Modul gespannt. Lassen Sie uns wissen, was Ihnen gefallen hat und wo Sie Verbesserungspotential sehen:



<https://www.surveymonkey.de/r/QM82XWN>

# Die Robo-Lehrkraft

Überall um uns herum gibt es Computerprogramme, z.B. Spiele auf dem Handy oder das Schreibprogramm auf dem Computer. Auch viele Maschinen, wie zum Beispiel Roboter, werden mit Hilfe von Programmen gesteuert. Die Programme werden zuvor von Menschen geschrieben. Ein Computerprogramm macht immer nur das, was vorher von Menschen programmiert wurde.



## Aufgaben

1. Teilt euch in Kleingruppen mit je 4 Schülerinnen und Schülern auf.
2. Stellt euch vor, eure Lehrkraft ist ein Roboter und ihr könnt sie programmieren.
3. Schreibt ein Computerprogramm, damit eure Lehrkraft einmal durch den Klassenraum läuft, ohne an Tische und Stühle zu stoßen. Benutzt folgende Befehle dafür:
  - a. Schritt gerade aus (bedeutet einen Schritt vorwärts)
  - b. Drehe links (bedeutet, dass sich die Lehrkraft auf der Stelle einmal um 90 Grad nach links dreht)
  - c. Drehe rechts (bedeutet, dass sich die Lehrkraft auf der Stelle einmal um 90 Grad nach rechts dreht)
  - d. Legt einen Start- und einen Endpunkt fest, z.B. die Tür des Klassenraumes oder ein bestimmtes Fenster. Überlegt dann genau, wie die Reihenfolge der Befehle lauten müsste, damit eure Lehrkraft ohne blaue Flecken durch die Klasse kommt. Schreibt eure Programmierung auf.
4. Testet eure Programme und lasst eure Lehrkraft durch den Klassenraum laufen. Hat alles geklappt?
5. Nun schreibt ein Computerprogramm für eine Schülerin oder einen Schüler aus einer andere Gruppe. Diesmal bekommt ihr ein paar mehr Befehle, zwei Knöpfe (A und B) und ihr könnt pfeifen.



Das ist der Start-Befehl



Das ist der Befehl für das Abspielen eines Tons (also einmal kurz pfeifen)



Programme reagieren auch auf Eingaben. Hier ist die Eingabe Knopf A. (Knopf A ist ein Post-It)



Eingabe Knopf B  
(Knopf B ist ein Post-It)



Dies ist ein „wenn-dann-Befehl“. Wenn z.B. der Knopf B gedrückt wurde, dann soll etwas passieren.



Diese Programmierung sagt, dass etwas 4 mal wiederholt werden soll, statt der 4 kann man auch eine andere Zahl eingeben



Gehe einen Schritt gerade aus



Drehe dich nach links



Drehe dich nach rechts

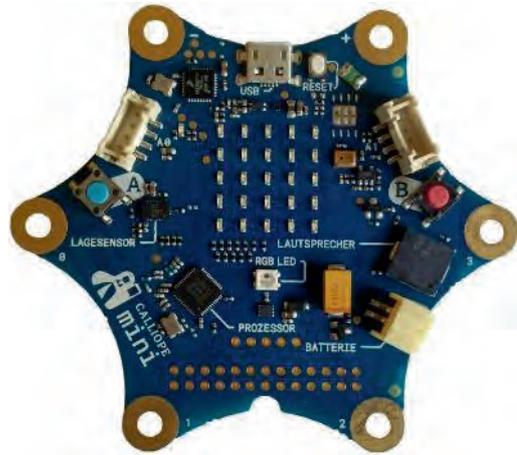
Wie sehen die folgenden Programme aus? Schreibt sie zuerst auf einem Blatt auf und testet sie danach mit euren Mitschülerinnen und Mitschülern.

- Wenn man Knopf A drückt, soll ein Ton abgespielt werden.
- Wenn man Knopf B drückt, soll die Mitschülerin oder der Mitschüler einen Schritt gerade aus, dann einen Ton abspielen und wieder einen Schritt gerade aus gehen. Das Ganze soll viermal hintereinander gemacht werden.
- Überlegt euch nun eine eigene Programme.
- Präsentiert euch gegenseitig eure Programme.

# Der Calliope mini

## Aufgabe 1 – der Calliope

Das ist ein Calliope mini, ein kleiner Computer mit dem man allerhand Sachen machen kann. Schaut ihn euch genau an.



- Was ist darauf zu erkennen? Beschreibt, was ihr seht.
- Damit der Calliope funktioniert, benötigt er Strom. Verbindet die Batterie mit dem Calliope. Schaut euch dafür den Stecker genau an. Wo passt er rein?

Wenn der Calliope richtig angeschlossen ist, leuchtet eine kleine LED (kleines Lämpchen) rechts oberhalb des A-Knopfes. Wird der Calliope das erste Mal verwendet, kann man sogar „Hallo“ auf dem Calliope mini lesen.

## Aufgabe 2 – die Programmierumgebung

So einen kleinen Computer nennt man auch Mikrocontroller. Damit der Mikrocontroller weiß, was er machen soll, muss man ihn programmieren. Einige Programmierbefehle habt ihr schon kennengelernt.

Nun schauen wir uns mal die Programmierumgebung an. Die Programmierumgebung ist ein Programm mit dem man neue Programme machen kann.

- Öffnet dazu den Browser und tippt folgendes ein:  
[makecode.calliope.cc](https://makecode.calliope.cc)

Klicke auf „Neues Projekt“, gib deinem Projekt einen Namen und klicke danach auf

Erstelle ✓

- b) Ganz links seht ihr ein Bild vom Calliope mini. In der Mitte findet ihr die bunten Bausteine zum Programmieren. Ganz rechts ist die Programmierfläche. Dorthin können die Blöcke mit der Maus gezogen werden. Klickt mit der linken Maustaste auf einen Programmierbaustein und haltet die Maustaste dabei gedrückt. Während ihr die Maustaste gedrückt haltet, könnt ihr den Baustein frei bewegen. Man kann Bausteine wie Puzzleteile verbinden. Bekommt ihr das hin?



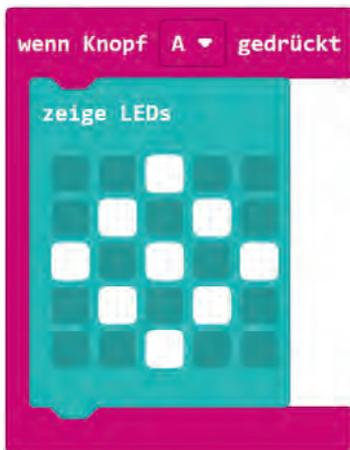
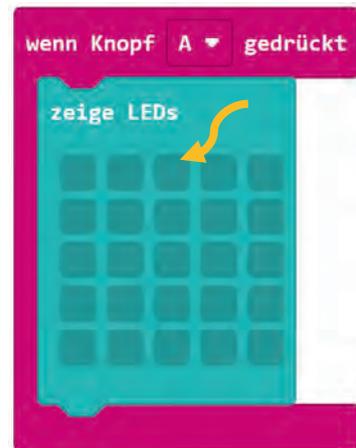
- c) Wenn ihr einen Baustein von der Programmierfläche löschen möchtet, dann haltet diesen Baustein mit der linken Maustaste gedrückt. Während ihr die linke Maustaste gedrückt haltet, bewegt ihr den Baustein zurück in die Sammlung der Programmierbausteine. Dabei erscheint eine Mülltonne. Legt den Stein in der Mülltonne ab und lasst die Maustaste wieder los. Schafft ihr es den blauen Baustein auf dem „dauerhaft“ steht, zu löschen?
- d) Damit euer Programm auf den echten Calliope drauf kommt, muss dieser mit dem Computer verbunden werden. Dazu nehmt ihr das USB-Kabel. Am einen Ende wird der Computer und am anderen Ende der Calliope angeschlossen.

## Aufgabe 3 – erste Programmierschritte

- a) Wir beginnen mit einer Eingabe (pinke Farbe). Wenn man auf einen Knopf drückt, soll etwas passieren. Geht dazu auf „Eingabe“ und wählt den ersten Baustein aus. Ihr könnt euch aussuchen, ob Knopf A oder B gedrückt werden soll, indem ihr den kleinen Pfeil neben dem Buchstaben A anklickt. Zieht den Baustein mit gedrückter, linker Maustaste nach rechts in das leere weiße Feld.

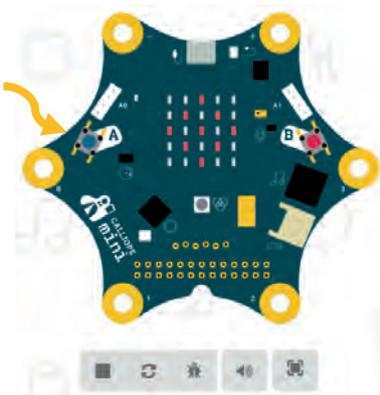


Danach klickt ihr auf den Baustein „Grundlagen“. Hier erhaltet ihr auch wieder ein Auswahl. Wählt den zweiten „zeige LEDs“ aus und zieht ihn auch nach rechts in den vorherigen Baustein.



Nun könnt ihr durch das Anklicken der kleinen Felder bestimmen, welche LEDs auf dem Calliope leuchten sollen. Links im Bild seht ihr ein Beispiel.

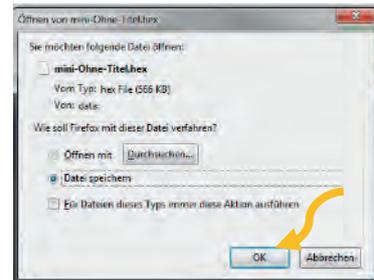
- b) Simulation: Nun kann das Programm getestet werden. Die Simulation im linken Teil des Bildschirms startet automatisch. Sobald der Simulator bereit ist, klickt auf den Knopf A direkt im Bild der Platine. Dann müssten die kleinen LEDs leuchten.



c) Hat alles geklappt? Dann kann das Programm auf den echten Calliope übertragen werden. Klickt dafür links unten auf „Herunterladen“.

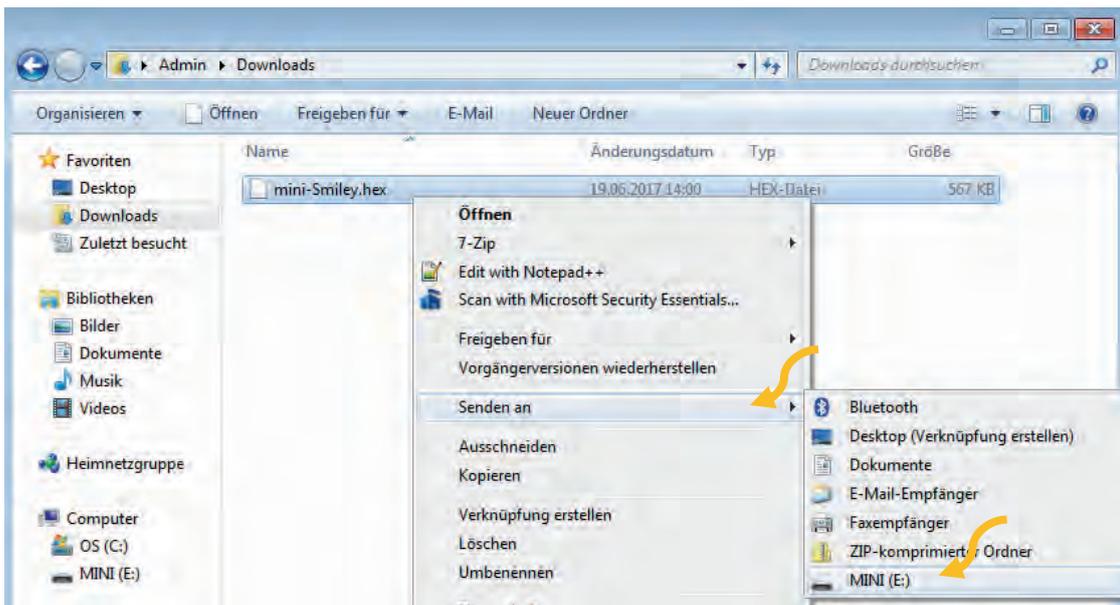


d) Es öffnet sich ein neues Fenster. Dort klickt ihr auf „OK“, um die Datei zu speichern.



e) Navigiert nun in den Downloads-Ordner. Klickt euer Programm mit der rechten Maustaste an, um das Kontextmenü zu öffnen. In diesem wählt ihr „Senden an“ aus, um in das nächste Kontextmenü zu gelangen. Dort sollte ganz unten das Laufwerk „MINI“ aufgelistet sein. Klickt dieses mit der linken Maustaste an, um das Programm auf dem Calliope mini zu speichern.

Dieses Bild kann dir dabei helfen:



f) Beim Übertragen des Programms beginnt eine gelbe LED auf der Platine zu blinken. Siehst du die LED? Sobald das gelbe Lämpchen aufgehört hat zu blinken, könnt ihr euer Programm testen. Falls euer Programm nicht startet, dann drückt auf den Reset-Taster. Das ist der weiße Taster rechts neben dem USB-Anschluss.

Für jedes neue Programm, das ihr erstellt, müsst ihr auch wieder so vorgehen.

# Kleine Programmieraufträge

## Aufgabe 1 – Wie heißt du?

Auf dem Calliope mini befindet sich eine LED-Matrix. Das sind die kleinen roten Lämpchen. Diese kann man ganz unterschiedlich programmieren. Wie wäre es mit einem Namensschild? Wählt dazu folgende Bausteine aus und verbinde sie miteinander:



## Aufgabe 2 – Hast du Töne?

Der Calliope hat einen Lautsprecher, daher kann er auch Töne abspielen. Versucht einmal das folgende Programm:

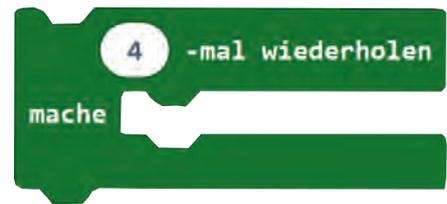


Wenn man nun Knopf A drückt, müsste euer Calliope drei verschiedene Töne abspielen.

Wie ihr seht, kann man auch mehrere Bausteine hintereinander setzen. Versucht nun eine eigene Komposition, wenn man *Knopf B* drückt.

### Aufgabe 3 – Schleifen

Immer wenn man möchte, dass der Computer etwas mehrmals hintereinander macht, braucht man Programmschleifen. Versucht es einmal mit eurem Programm aus Aufgabe 2. Die drei verschiedenen Noten sollen viermal hintereinander abgespielt werden.



Wie könnte das Programm aussehen?

### Aufgabe 4 – Schüttel mich!

Statt der Knöpfe A und B kann man als Eingabe auch den Befehl „wenn geschüttelt“ verwenden. Nun wollen wir programmieren, dass der Calliope mini die Temperatur anzeigt, wenn er geschüttelt wird. Wählt dazu folgende Bausteine aus und setzt sie zusammen:



Temperatur (°C)



Wie sieht die fertige Programm aus? Testet sie mithilfe des Simulators.

### Aufgabe 5 – Würfel

Nun versucht doch einmal, mit Hilfe des Calliope einen Würfel zu programmieren. Die wesentlichen Bausteine dafür kennt ihr schon. Ihr braucht aber auch noch diesen Baustein:

wähle eine zufällige Zahl von 0 bis 10

Funktioniert der Würfel?

### Aufgabe 6 – eigene Idee umsetzen

Nun setzt eure eigenen Ideen um. Probiert dabei auch Bausteine aus, die ihr bisher noch nicht verwendet habt. Wenn ihr interessante Dinge dabei herausfindet, teilt eure Erkenntnisse mit euren Mitschülerinnen und Mitschülern.

# Hanna muss ins Krankenhaus!

Hanna geht es gar nicht gut. Sie hat schon seit Wochen furchtbare Halsschmerzen und Fieber. Deshalb muss sie jetzt ins Krankenhaus. Die Mandeln müssen entfernt werden, ob Hanna will oder nicht.



Daher packen die Eltern ein paar Dinge zusammen und fahren sie in die Klinik.

Hanna soll am nächsten Tag operiert werden. Gut, dass sie ihr Lieblingsschaf Oskar dabei hat.

Die Zeit bis dahin findet sie sehr langweilig. Außerdem kommt ständig jemand rein zum Fieber messen. Das nervt ein bisschen. Dann muss sie auch noch zu einer weiteren Untersuchung. In dieser Zeit muss ihr Lieblingskuscheltier allein im Zimmer bleiben. Hoffentlich klaut niemand das Schaf – es ist doch ihr ein und alles.

Am nächsten Tag wird Hanna operiert. Sie ist sehr aufgeregt, aber alles verläuft gut. Nur das Sprechen fällt ihr nach der Operation etwas schwer, da es noch im Hals wehtut. Aber dafür kann sie so viel Eis essen, wie sie möchte!

## Aufgaben

1. Wie könnte der Calliope mini Computer Hanna im Krankenhaus helfen?
  - a. Unterstreicht Hannas Probleme im Krankenhaus im Text.
  - b. Was könnte ihr in der jeweiligen Situation helfen? Sammelt Ideen.
2. Sucht euch in eurer Gruppe eine Idee aus und versucht dazu etwas zu programmieren.
3. Wenn ihr fertig seid, präsentiert eure Programme vor der Klasse.

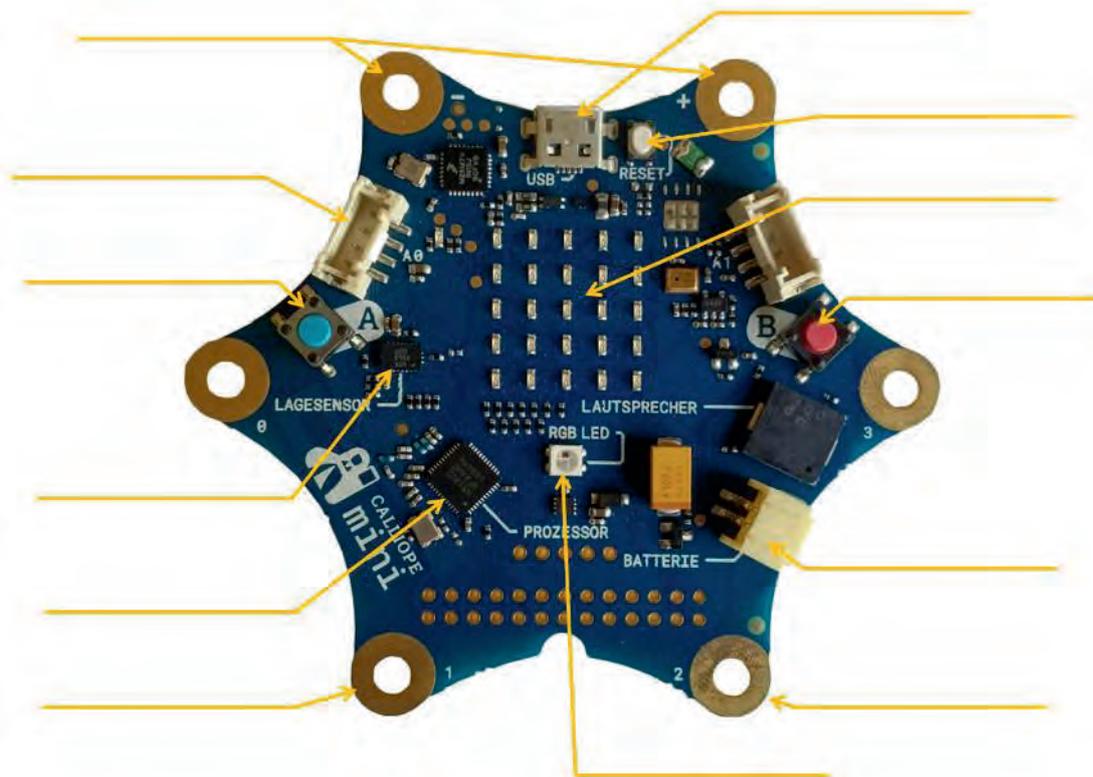
Toll! Danke, dass ihr Hanna helfen konntet!

# Der Calliope mini - Tutorial

Der Calliope mini ist ein Mikrocontroller, welcher mit viel zusätzlicher Hardware ausgestattet ist. Erfahre in diesem Tutorial wie man diesen kleinen Computer programmiert.

## Aufgabe 1

Schau dir den Calliope genau an und beschrifte die einzelnen Komponenten:

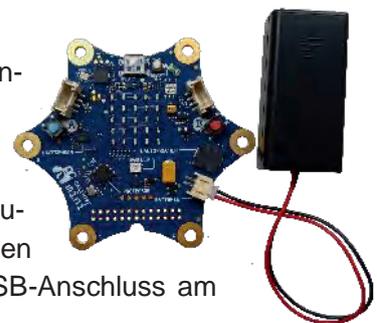


## Aufgabe 2 – die Stromversorgung

Der Calliope mini braucht Strom, um zu funktionieren. Dafür wird der Batterieblock an den richtigen Anschluss gesteckt.

Probiere es aus. Achte darauf, dass der Stecker richtig rum hineingesteckt wird. Der Stecker sollte ganz leicht hinein gehen.

Wenn du keine Batterien zur Verfügung hast, kannst Du den Calliope mini auch direkt mit einem Micro-USB-Kabel an einen Computer anschließen. Stecke dazu das Kabel in den dafür vorgesehenen Anschluss an den Calliope. Das andere Ende wird mit dem USB-Anschluss am Computer verbunden.



Diese Verbindung benötigst du auch, um den Calliope zu programmieren.

# Die Programmierumgebung

Der Calliope kann nun selbst von dir programmiert werden. Öffne dazu deinen Browser, um ins Internet zu gelangen. Öffne die folgende Internetseite: <https://makecode.calliope.cc>. Klicke auf „Neues Projekt“, gib dem Projekt einen treffenden Namen und bestätige mit einem Klick auf „Erstellen“. Du kommst dann direkt zur Programmierumgebung. Vielleicht kommt dir einiges bekannt vor, denn die Programmierblöcke erinnern ein wenig an Scratch.



## Die Programmierbausteine

Die einzelnen Programmier-Bausteine sind in Kategorien nach Funktion und Farbe zusammengefasst. Schau dir die einzelnen Kategorien genauer an. Wenn du darauf klickst, kommen die einzelnen Programmier-Bausteine zum Vorschein.

Suche...	Mit <b>Suche</b> findest du schneller deine Programmierbausteine.
<b>Grundlagen</b>	Unter <b>Grundlagen</b> findest Du Basisbausteine mit denen du direkt loslegen kannst, z.B. LEDs zum Leuchten bringen.
<b>Eingabe</b>	<b>Eingaben</b> helfen dir zu programmieren, wann etwas passieren soll z.B. wenn eine bestimmte Taste gedrückt wird.
<b>Musik</b>	<b>Musik</b> bringt Töne mit rein.
<b>LED</b>	Unter <b>LED</b> findest du tiefergehende Programmierbausteine für die LEDs.
<b>Funk</b>	<b>Funk</b> beinhaltet Bausteine zur kabellosen Datenübertragung.
<b>Schleifen</b>	<b>Schleifen</b> sagen, wie oft etwas wiederholt werden soll.
<b>Logik</b>	Mit <b>Logik</b> kannst du Bedingungen festlegen z.B. wenn- dann-Bedingungen.
<b>Variablen</b>	<b>Variablen</b> können Daten zwischenspeichern.
<b>Mathematik</b>	Unter <b>Mathematik</b> können Zahlen oder Variablen miteinander verknüpft werden.
<b>Motoren</b>	Unter <b>Motoren</b> findest du Bausteine, um angeschlossene Motoren anzusteuern.
<b>Fortgeschritten</b>	Und die Kategorie <b>Fortgeschritten</b> beinhaltet Bausteine für Profis, um zahlreiche Ein- und Ausgänge oder bestimmte Spielelemente zu programmieren.

## Das erste Programm

Zu Beginn befasst du dich mit den Grundlagen-Bausteinen. Diese dienen unter anderem dazu, die LEDs auf dem Calliope mini anzusteuern.

### Aufgabe 1

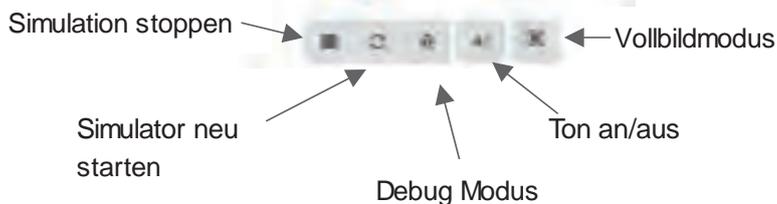
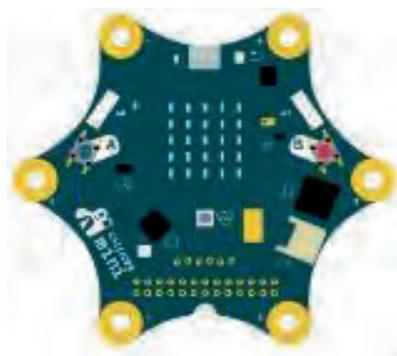
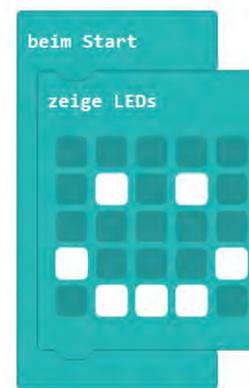
Versuche ein erstes eigenes Bild mit Hilfe der LEDs zu programmieren. Verwende dafür die folgenden beiden Bausteine und setze sie zusammen. Mit einzelnen Mausklicks kannst du die LEDs bestimmen, die leuchten sollen:



Wenn du dich schon etwas mit Scratch auskennst, dann ist das sicherlich kinderleicht für dich.

Das Programm könnte so aussehen: In dem Beispiel rechts im Bild wurde ein Smiley programmiert.

Nun kannst du das Programm mit Hilfe des Simulators testen.



## Das Programm auf den Calliope übertragen

Damit nun die LEDs des echten Calliope leuchten, muss das Programm übertragen werden. Verbinde den Calliope mit Hilfe des USB-Kabels mit deinem Computer. Der Calliope wird wie ein USB-Stick als Laufwerk erkannt und trägt die Berechnung „MINI“.

Lade im Anschluss dein Programm herunter. Klicke dafür unten links im Bildschirm auf Herunterladen.



Es öffnet sich ein neues Fenster, dort klickst du auf „Ok“. Dein Programm wird dann im Download-Ordner gespeichert.

Im nächsten Schritt navigierst du in den Downloads-Ordner und wählst dort dein Programm aus. Mit gedrückter, linker Maustaste ziehst du die Datei dann auf den Mini. Du kannst die Datei natürlich auch kopieren und in dem Laufwerk einfügen.

Während das Programm auf den Calliope übertragen wird, blinkt auf diesem eine gelbe Status LED rechts oberhalb des A-Knopfes. Sobald diese aufgehört hat zu blinken, kann das Programm getestet werden. Je nach Calliope Version kann es notwendig sein, den Mini per Hand neuzustarten. Falls dein Programm nicht startet, drücke auf den Reset-Taster. Das ist der weiße Taster rechts neben dem USB-Anschluss.

### Aufgabe 2

Probiere nun verschiedene Bausteine aus und teste sie. Verwende Bausteine aus den ersten drei Kategorien: **Grundlagen**, **Eingabe** und **Musik**. Schau was passiert, wenn du die Bausteine unterschiedlich zusammensetzt.

Wie wäre es mit einem Spiel, wie z.B. Schere, Stein, Papier oder einer witzigen Sound-Maschine?



# Die 7a auf Klassenfahrt

Die Klasse 7a plant eine Klassenfahrt zu einem ganz besonderen Zeltplatz. Sie werden mitten im Wald an einem See, genau wie amerikanische Ureinwohner, in Tipis schlafen. Die Schülerinnen und Schüler haben dort viel vor: Lagerfeuer machen, Fußball spielen, wandern, schwimmen und natürlich auch ein bisschen faulenzeln und daddeln.



Für die Klassenfahrt muss noch einiges besorgt werden: Proviant, Getränke, ggf. Schlafsack und Iso-Matte, usw. Bei den Vorbereitungen zur Klassenfahrt kommen aber auch noch ein paar Fragen und Wünsche auf, die die Klassenlehrerin Frau Schmidt an der Tafel sammelt.

Paul würde sich mit einem Kompass auf den Wanderungen wohler fühlen, um sich besser in der unbekannteren Umgebung zu orientieren. Er will sich schließlich nicht verlaufen.

Seine Klassenkameradin Julia macht sich Sorgen um ihre Wertsachen und fragt, ob es dort auch einen Tresor gäbe, denn dann könne sie ihre wichtigen Sachen einschließen. Tim, Julia und Ben, die Sportskanonen der Klasse, schlagen vor, einen Schrittzähler mitzunehmen. „Da könnten wir doch eine Challenge draus machen: Wer geht die meisten Schritte?!“, schlägt Ben vor.

Als alles notiert ist, stellt sich in der Runde die Frage, wer sich um all die Dinge kümmern kann. Einer seufzt: „Ach wäre das toll, wenn man einen kleinen Computer hätte, der das alles kann.“ „Na klar“, sagt Tina und schlägt sich an die Stirn, „der Calliope kann das. Man muss ihn nur richtig programmieren und vielleicht fallen uns ja auch noch weitere Funktionen ein“ und ganz leise sagt sie hinterher „... vielleicht ein Spiel?“

## Aufgaben:

1. Habt ihr noch mehr Ideen, was man auf einer solchen Klassenfahrt gebrauchen und mit dem Calliope mini programmieren könnte?
  - 
  - 
  -
2. Teil euch in Kleingruppen auf und überlegt, wie die Vorschläge mit dem Calliope programmiert werden können?
  - Kompass
  - Tresor/ Alarmanlage für die Wertsachen
  - Schrittzähler
3. Wählt einen der vorgegebenen oder einen eigenen Vorschlag aus und setzt ihn mit dem Calliope mini um.

# Programmierauftrag: Kompass

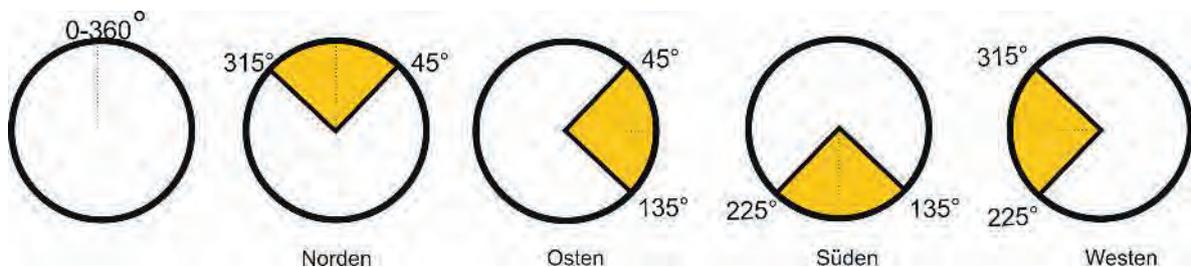
Nutzt den Calliope mini, um einen Prototypen für einen Kompass zu erstellen. Programmiert mit Hilfe der Entwicklungsumgebung MakeCode: <https://makecode.calliope.cc/>

## Aufgaben:

1. Wie funktioniert ein Kompass? Diskutiert, welche Bedingungen erfüllt werden müssen und versucht diese zu beschreiben:
  - a. Was zeigt ein Kompass an?
  - b. Wie kann ein Kompass aussehen?
  - c. Wie könnte das Kompass-Programm aufgebaut sein? (z.B. wenn x, dann y)
2. Ihr habt nun schon einige Erfahrung mit dem Calliope mini. Welche Programmierbausteine könnten dafür in Frage kommen?
3. Programmiert und testet eure Idee.
4. Überlegt euch nun ein interessantes und praktisches Design für euren Kompass. Wie soll der Kompass bedient werden? Wie und wo wird er getragen? Wird er vor Feuchtigkeit geschützt? Welche Kriterien könnten für das Design noch wichtig sein?
5. Stellt euren Kompass am Ende der Klasse vor.

## Tipps:

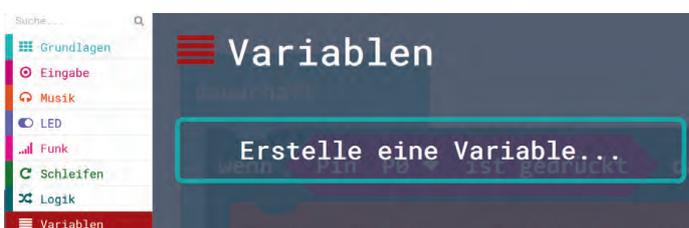
### Kompassausrichtung (°)



Die obige Abbildung hilft euch den Winkel der Kompassausrichtung in Himmelsrichtungen zu übersetzen.

### Kompass kalibrieren

Der Kompass muss nach dem Einschalten kalibriert werden. Folge den Anweisungen auf der LED-Matrix und bewege den Calliope so hin und her, dass der gesamte Bildschirm rot leuchtet.



Ihr benötigt einen Platzhalter für die Himmelsrichtung in eurem Programm. Einen Platzhalter bezeichnet man in der Mathematik und in der Informatik auch als Variable.

# Programmierauftrag: Tresor mit Alarmanlage

(Fortgeschritten)

Nutzt den Calliope mini, um einen Prototypen für einen Tresor mit Alarmanlage zu erstellen. Programmiert mit Hilfe der Entwicklungsumgebung MakeCode: <https://makecode.calliope.cc/>

## Aufgaben:

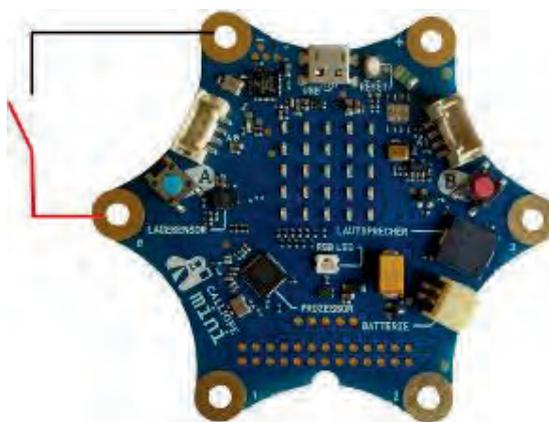
1. Wie funktioniert ein Tresor mit Alarmanlage?
  - a. Wie soll sich euer Tresor verhalten, wenn
    - i. dieser unauthorisiert geöffnet wird?
    - ii. dieser von euch geöffnet wird?
  - b. Wie könnte eurer Tresor-Programm aufgebaut sein? (z.B. wenn x passiert, dann soll y)
2. Ihr habt nun schon einige Erfahrung mit dem Calliope. Welche Programmierbausteine könnten dafür in Frage kommen?
3. Programmiert und testet eure Idee.
4. Überlegt euch nun ein interessantes und praktisches Design für euren Tresor mit Alarmanlage. Ihr könnt auch weitere Hilfsmittel verwenden, z.B. Karton, Krokodilklemmen, Alufolie, usw.
5. Stellt euren Tresor mit Alarmanlage am Ende der Klasse vor.

## Tipps:



Für dieses Projekt kann der Pin P0 auf der linken Seite der Platine verwendet werden. P0 wird als gedrückt registriert, wenn der Pin mit Masse (-) verbunden ist. Nutze dies für deine Alarmanlage.

Der Calliope mini kann offene und geschlossene Stromkreise erkennen. Um einen Stromkreis zu schließen, muss beispielsweise Pin 0 mit Masse (-) verbunden werden. Den Kontakt stellt man am besten mit Krokodilklemmen her.



# Programmierauftrag: Alarmanlage

Nutzt den Calliope mini, um einen Prototypen für eine Alarmanlage zu erstellen. Programmiert mit Hilfe der Entwicklungsumgebung MakeCode <https://makecode.calliope.cc/>

## Aufgaben:

1. Wie funktioniert eine Alarmanlage?
  - a. Wie soll eure Alarmanlage funktionieren? Wie aktiviert man den Alarm? Wie deaktiviert man den Alarm?
  - b. Was soll beispielsweise passieren, wenn das Zelt geöffnet wird oder eine Tasche genommen wird?
  - c. Wie könnte eurer Alarmanlagen-Programm aufgebaut sein? (z.B. wenn x passiert, dann soll y )
2. Ihr habt nun schon einige Erfahrung mit dem Calliope. Welche Programmierbausteine könnten dafür in Frage kommen?
3. Programmiert und testet eure Idee.
4. Überlegt euch nun ein interessantes und praktisches Design für eure Alarmanlage.
5. Stellt eure Alarmanlage am Ende der Klasse vor.

## Tipp:



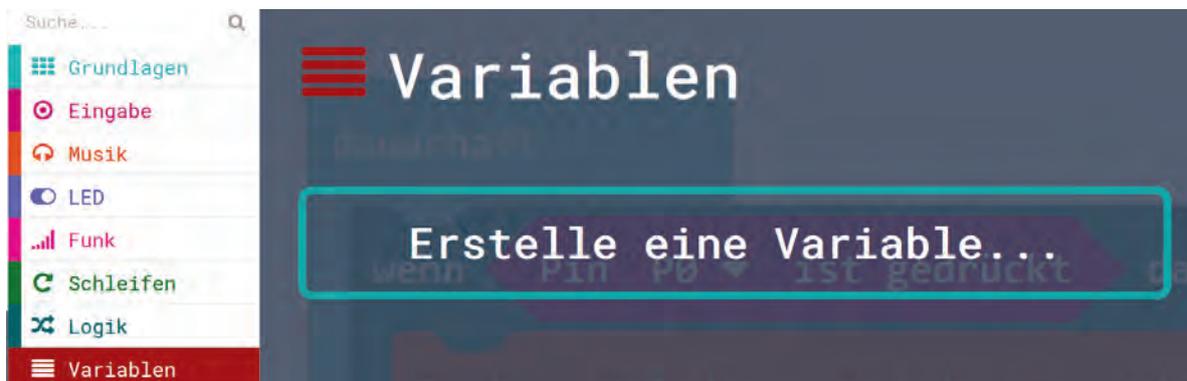
# Programmierauftrag: Schrittzähler

Nutzt den Calliope mini, um einen Prototypen für einen Schrittzähler zu erstellen. Programmiert mit Hilfe der Entwicklungsumgebung MakeCode <https://makecode.calliope.cc/>

## Aufgaben:

1. Wie funktioniert ein Schrittzähler?
  - a. Wie soll eurer Schrittzähler funktionieren? Wie startet man den Zähler und wie setzt man diesen z.B. wieder zurück? Gibt es Belohnungen für eine bestimmte Schrittzahl?
  - b. Wie könnte eurer Schrittzähler-Programm aufgebaut sein? (z.B. wenn x passiert, dann soll y )
2. Ihr habt nun schon einige Erfahrung mit dem Calliope. Welche Programmierbausteine könnten dafür in Frage kommen?
3. Programmiert und testet eure Idee.
4. Überlegt euch nun ein interessantes und praktisches Design für euren Schrittzähler. Wie und wo wird er getragen, wird er vor Feuchtigkeit geschützt, etc. Welche Kriterien könnten für das Design noch wichtig sein?
5. Stellt euren Schrittzähler am Ende der Klasse vor.

## Tipp:



Ihr benötigt einen Platzhalter in eurem Programm, der sich die bisherigen Schritte merkt. Einen Platzhalter bezeichnet man in der Mathematik und in der Informatik auch als Variable.

# Programmierauftrag: Eigene Idee

Ihr habt eine eigene Idee für die Nutzung des Calliope während der Klassenfahrt für die 7a? Programmiert mit Hilfe der Entwicklungsumgebung MakeCode <https://makecode.calliope.cc/>

## Aufgaben:

1. Was soll erstellt werden? Wie soll das Gerät funktionieren?
  - Diskutiert, welche Bedingungen erfüllt werden müssen und versucht es zu beschreiben (z.B. wenn x, dann y)
2. Ihr habt nun schon einige Erfahrung mit dem Calliope. Welche Programmierbausteine könnten dafür in Frage kommen?
3. Programmiert und testet eure Idee.
4. Überlegt euch nun ein interessantes und praktisches Design für eure Idee. Welche Kriterien könnten für das Design wichtig sein?
5. Stellt euren Prototypen am Ende der Klasse vor.

# Musterlösungen

## Arbeitsblatt B8.3 - Kleine Programmieraufträge

Aufgabe 1 – Wie heißt du?



[https://makecode.calliope.cc/\\_9MgLki6MeiEz](https://makecode.calliope.cc/_9MgLki6MeiEz)

Aufgabe 2 – Hast du Töne?



[https://makecode.calliope.cc/\\_YFYX74fie22R](https://makecode.calliope.cc/_YFYX74fie22R)

Aufgabe 3 – Schleifen



[https://makecode.calliope.cc/\\_3YFepc2DVc4H](https://makecode.calliope.cc/_3YFepc2DVc4H)

Aufgabe 4 – Schüttel mich!



[https://makecode.calliope.cc/\\_9A8U6C98L7Yb](https://makecode.calliope.cc/_9A8U6C98L7Yb)

## Aufgabe 5 – Würfel

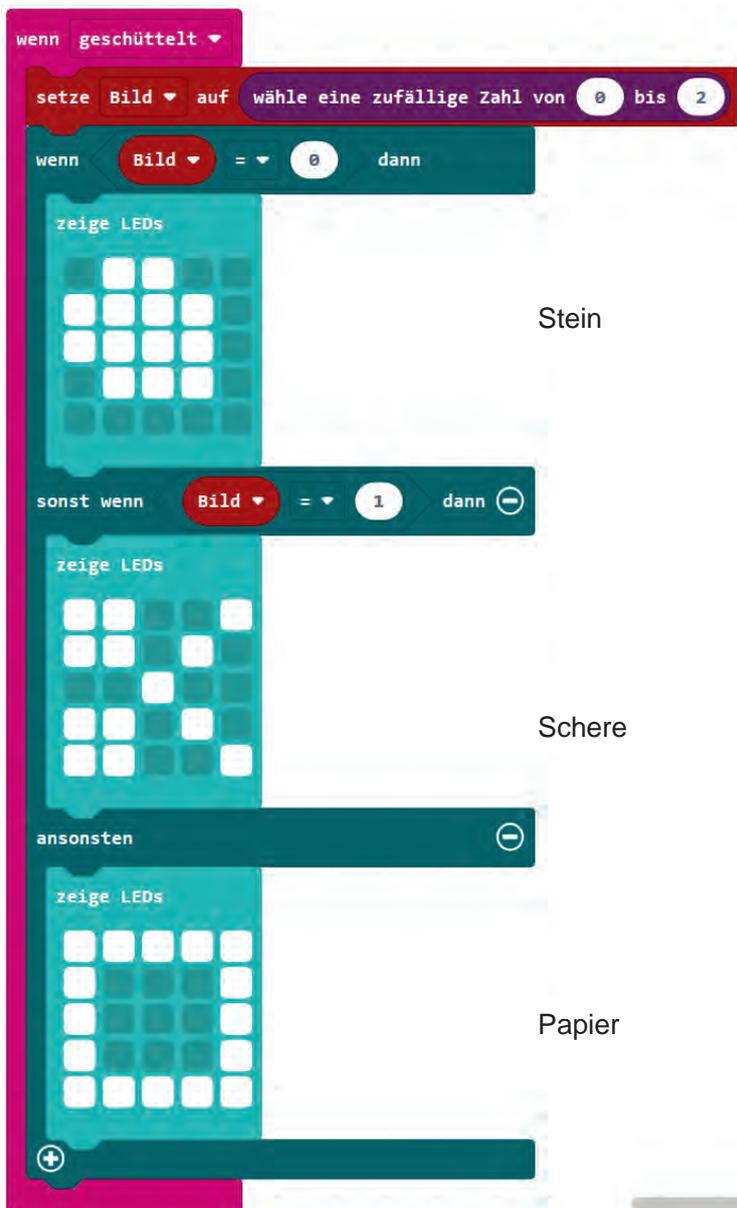


[https://makecode.calliope.cc/\\_eiT0dP40hJYr](https://makecode.calliope.cc/_eiT0dP40hJYr)

## Arbeitsblatt B8.4 Hanna muss ins Krankenhaus

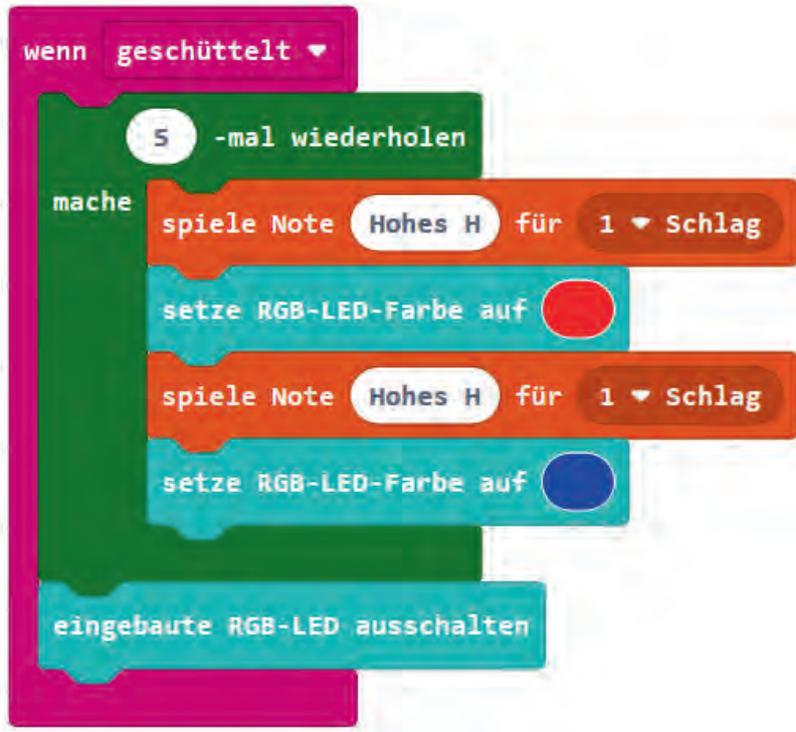
Für folgende Situationen können Programmierungen erstellt werden:

- a) Etwas gegen Langeweile z.B.
  - Ein Musikstück (siehe AB 8.3)
  - Würfel (siehe AB 8.3)
  - Kleines Spiel z.B. Stein, Schere, Papier



[https://makecode.calliope.cc/\\_DCE08zfkAgPF](https://makecode.calliope.cc/_DCE08zfkAgPF)

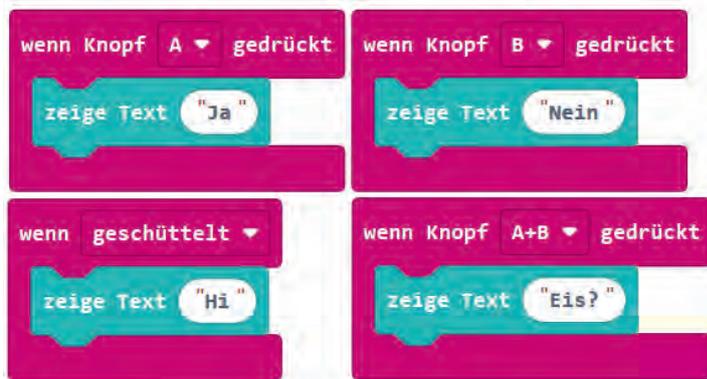
Eine Alarmanalge für das Schaf:



Der Calliope mini kann am Schaf befestigt werden. Wird das Schaf bewegt, werden Töne abgespielt und die RGB-LED blinkt in verschiedenen Farben.

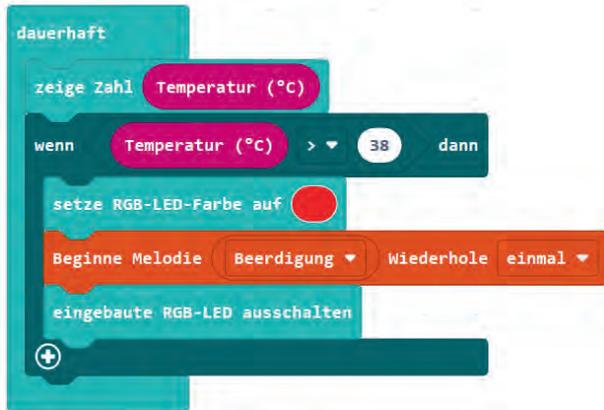
[https://makecode.calliope.cc/\\_YRA1xsTUHhoj](https://makecode.calliope.cc/_YRA1xsTUHhoj)

b) Zum Antworten, wenn Hanna das Sprechen schwer fällt z.B.



[https://makecode.calliope.cc/\\_ckpD0eHwR4Kt](https://makecode.calliope.cc/_ckpD0eHwR4Kt)

## c) Zum Fiebermessen



Die Temperatur soll dauerhaft angezeigt werden

Wenn die Temperatur über 38 steigt, soll ein Alarm ertönen und die LED rot leuchten

[https://makecode.calliope.cc/\\_Wj75yobbu9em](https://makecode.calliope.cc/_Wj75yobbu9em)

### Tipp:

Zum Temperaturmessen am eigenen Körper kann der Calliope direkt auf den Bauch, Brust oder den Unterarm auf die Haut gelegt werden. Da hoffentlich keine Schülerin oder Schüler tatsächlich Fieber hat, kann man den Alarm auch für eine Temperatur von 30 Grad einstellen. Dadurch geht der Alarm schneller los und mehrere Schülerinnen und Schüler können testen, wie der Temperatursensor des Calliope funktioniert.

## Arbeitsblatt B8.7 Die 7a auf Klassenfahrt

### Programmierauftrag: Kompass



Je nach Gradzahl wird, N (Norden), O (Osten), S (Süden) oder W (Westen) angezeigt.

Statt der Kurzbezeichnungen sind natürlich auch Piktogramme oder Ähnliches möglich.

[https://makecode.calliope.cc/\\_CkcamahX19Ps](https://makecode.calliope.cc/_CkcamahX19Ps)

## Kalibrierung des Kompasses:

Um die Kalibrierung durchzuführen, wird man nach dem Übertragen des Programms aufgefordert den Bildschirm komplett auszufüllen – auf dem Display erscheint die Aufforderung: TILT TO FILL SCREEN

Es leuchtet jedes mal ein einziges Pixel in der Matrix auf. Durch das Schaukeln des Calliope bewegt sich das Pixel. Sobald sich das Pixel auf einer freien Stelle befindet, erscheint ein neues blinkendes Pixel. Diesen Vorgang wiederholt man insgesamt 25 mal. Sobald die gesamte Matrix rot leuchtet, wartet man einen Moment ab. Ein Smiley bestätigt die erfolgreiche Kalibrierung.

## Programmierauftrag: Schrittzähler



Beim Starten soll der Schrittzähler immer bei „0“ beginnen.

Möchte man zwischendurch neu starten, kann man auch einbauen, dass bei einem bestimmten Tastendruck die Zählung wieder bei „0“ beginnt.

Wenn der Calliope geschüttelt wird, wird der Schrittzähler aktiv; es wird immer um einen Schritt vergrößert.

Bei 10.000 Schritten erhält man einen Signalton

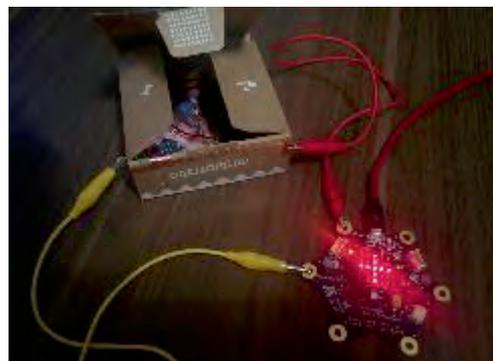
[https://makecode.calliope.cc/\\_gP73iz6vT686](https://makecode.calliope.cc/_gP73iz6vT686)

## Programmierauftrag: Alarmanlage

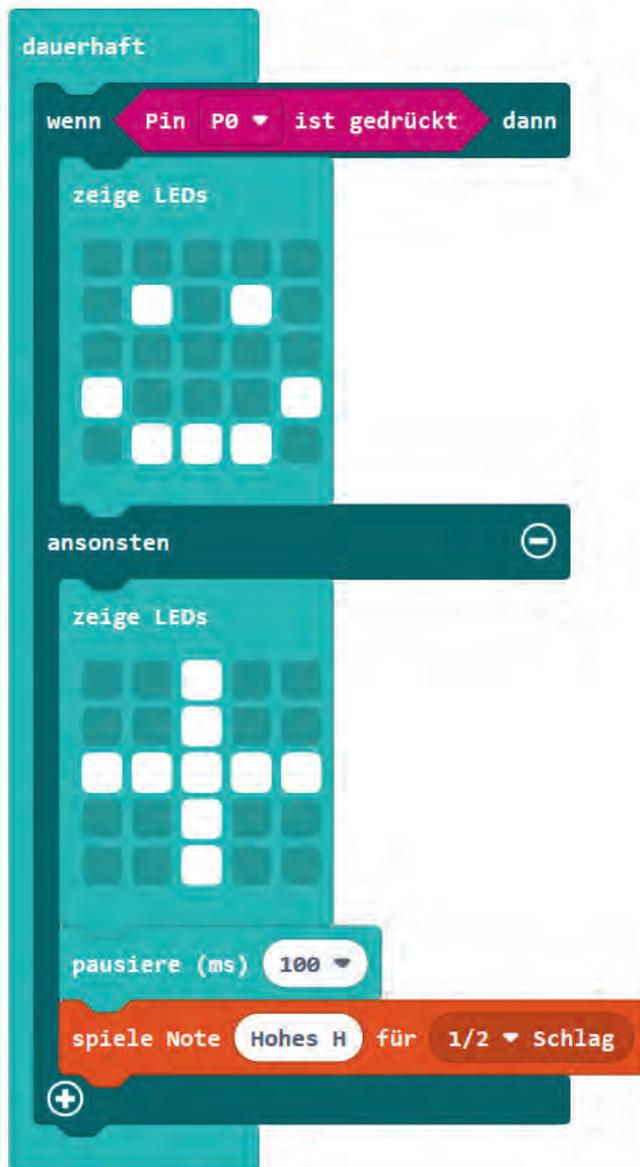
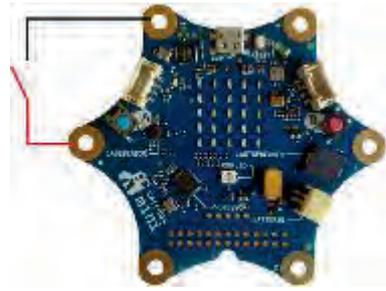
Siehe beispielhafte Lösung Aufgabe B8.4 (b) (einfach)

## Tresor mit Alarmanlage - Alternative für Fortgeschrittene:

Für diese Programmierung kann ein Tresor gebaut werden. Dafür benötigt man eine Box, Krokodilklemmen und die Kupferfolie aus der Calliope Mini Box. Alternativ kann auch Alufolie verwendet werden. Der Calliope erkennt, ob der Stromkreislauf geschlossen oder offen ist - je nachdem kann der Alarm ausgelöst werden.



Hierfür muss beispielsweise Pin 0 mit Masse (-) verbunden werden. In diesem Beispiel wurde die Verbindung über die Kupferfolie, welche an einer Box angebracht ist, hergestellt. Wenn die Box geschlossen ist, ist auch der Stromkreislauf geschlossen und der Calliope mini zeigt einen Smiley an. Wenn die Box geöffnet wird, erscheint ein Kreuz und ein Signal ertönt.



[https://makecode.calliope.cc/\\_0hPLcehURXDr](https://makecode.calliope.cc/_0hPLcehURXDr)

Man könnte zusätzlich noch einen Schlüssel mit einem anderen Calliope via Funk übertragen, um den Alarm zu deaktivieren. Details zu diesem Projekt sind hier verlinkt: <https://make.tech-willsaveus.com/microbit/activities/security-box>