



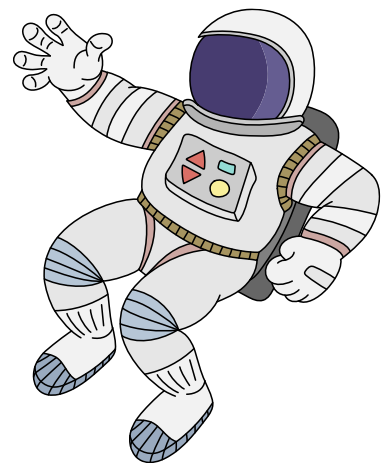
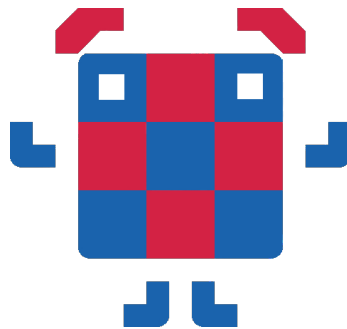
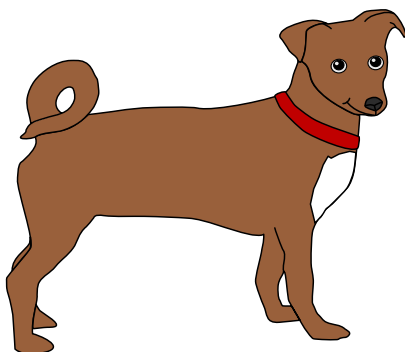
5. – 7. Klasse

Teil 1

Informatik erleben

– Pseudocode –

Hier gibt es Unterrichtsverlaufspläne, Arbeitsblätter,
Kopiervorlagen und Programmieraufgaben für den
Einstieg in die Welt der Algorithmen und der Informatik.



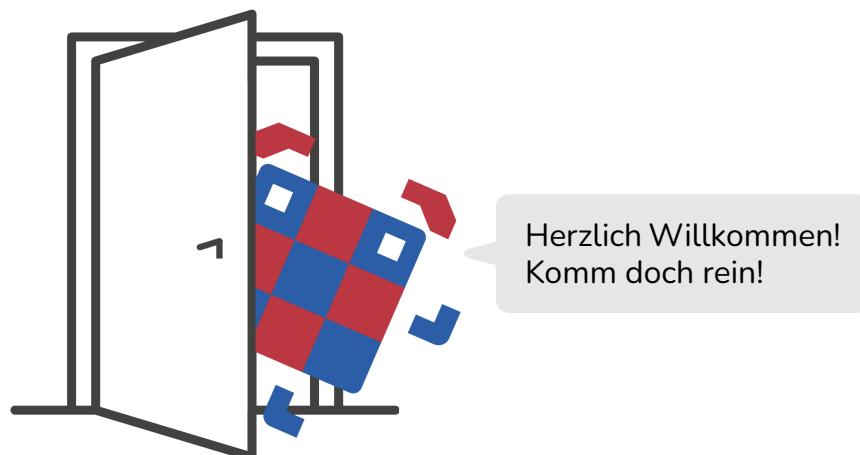
Herzlich Willkommen zu unserer Lernreihe

Wie schön, dass Du da bist! 😊 Mit dieser Unterrichtsreihe bekommst Du alles, was Du brauchst, um den Einstieg in die Programmierung mit Deiner Lerngruppe sorglos zu gestalten an die Hand. Mithilfe der für den Unterricht konzipierten Lernsoftware **Cubi** kannst Du das Thema **Programmierung** kleinschrittig und ganzheitlich mit Deiner Klasse entdecken.

Das IT4Kids-Material zu **Schleifen**, **Verzweigungen**, **Variablen** und Co. vermittelt die grundlegenden Programmierkenntnisse, um das Informatik-Thema **Algorithmen** vollständig zu behandeln.

Keine Sorge: Es wird kein Vorwissen benötigt. Durch unsere Materialien kannst Du Dir die Welt der Programmierung Schritt für Schritt erschließen. Mithilfe vorgefertigter Programmieraufgaben für die Schüler*innen und ausgearbeiteter Unterrichtsverlaufspläne für Dich als Lehrkraft, wollen wir Dir so viel Unterrichtsvorbereitung abnehmen wie möglich. Dazu stellen wir Dir auch Arbeitsblätter, Kopiervorlagen und Musterlösungen zur Verfügung.

Du möchtest Dich erst einmal mit unserer Lernsoftware vertraut machen? Kein Problem! Du findest den Cubi-Editor unter `editor.i4k.org`. Das **Benutzerhandbuch für die Lernsoftware Cubi** verrät Dir alles, was Du bei der Nutzung der Lernsoftware wissen solltest. Du findest es im Begleitmaterial.



Das vorliegende Lehrmaterial von IT4Kids und zugehörige Begleitmaterialien für Schüler*innen stehen, soweit nicht anders angegeben, unter der Creative Commons-Lizenz CC BY-NC-SA 4.0. Weitere Informationen zu der Lizenz findest Du hier: <http://creativecommons.org/licenses/by-nc-sa/4.0/>



Informatik als Fachunterricht in der Sekundarstufe I

In immer mehr Bundesländern erhält das Fach Informatik einen festen Platz im Stundenplan der Jahrgangsstufen 5 bis 7 oder wird dort erprobt. Das Ziel der vorliegenden Lernreihe von IT4Kids ist, Schüler*innen einen ganzheitlichen Einstieg in den **Inhaltsbereich Algorithmen** zu bieten. Dabei vermitteln wir insbesondere die Prozessbereiche **Modellieren und Implementieren, Begründen und Bewerten** und **Kommunizieren und Kooperieren**. Wir folgen hiermit den *Grundsätze[n] und Standards für die Informatik in der Schule* der Gesellschaft für Informatik e.V.

Schon gewusst?

Die Inhalte von IT4Kids entsprechen dem Strategiepapier der KMK für *Bildung in der digitalen Welt* und den Zielen für nachhaltige Entwicklung.

Im Laufe der vorliegenden Unterrichtsstunden lernen die Schüler*innen verschiedene Anweisungen in der grafischen Programmierumgebung **Cubi** kennen. Mit diesen können sie **sequentielle Algorithmen** und Algorithmen mit **Schleifen** und **bedingten Anweisungen** modellieren und implementieren. Im zweiten Teil der Lernreihe kommen **Variablen** und **Funktionen** hinzu. Außerdem wird das große Thema **Fehlersuche und Testen** aufgearbeitet und Programme werden mit Stift und Papier geplant. Den Abschluss bildet ein kreatives Projekt, in dem eigene Spiele entwickelt werden.

Die Lernentwicklung der Schüler*innen wird über die gesamte Lernreihe hinweg auch durch **überfachliche Kompetenzen** gefördert. Dadurch, dass sie die Konsumperspektive verlassen und erfahren, wie sie die digitale Welt kreativ mitgestalten können, werden **personale Kompetenzen** gestärkt, die auf die Förderung der Selbstwirksamkeit, -behauptung und -reflexion abzielen.

Auch die **motivationale Einstellung** der Schüler*innen wird mit den Lehrinhalten gesteigert. Die Neugierde der Schüler*innen für den neuen Themenbereich der Informatik wird geweckt, sodass sie sich für diesen begeistern und neuen Problemstellungen ausdauernd begegnen können. Dabei wird eine positive Einstellung gegenüber experimentellem Lernen und die Frustrationstoleranz der Schüler*innen ausgebaut.

Durch eine Varianz an Sozialformen und die Integration von Partner- und Gruppenarbeiten werden **soziale Kompetenzen** wie das Agieren in kooperativen Lernprozessen oder der konstruktive Umgang mit Konflikten und Vielfalt gefordert und gefördert.

Die Schüler*innen erweitern ihre **Methodenkompetenz**, indem sie beim Lernen strukturiert sowie systematisch vorgehen und eigene Arbeitsprozesse planen und organisieren. Das Lösen von Programmieraufgaben fordert ein hohes Maß an Problemlösefähigkeit, das im Verlauf der Lernreihe auf- und ausgebaut wird. Bei der Arbeit an ebendiesen Programmieraufgaben sowie den damit verbundenen Recherche- und Präsentationsaufträgen ist die Förderung der Medienkompetenz der Schüler*innen allgegenwärtig.

Verankerungen von Inhalten zu Algorithmen in Bildungsplänen

Die Inhalte wurden für die verschiedenen Anforderungen der länderspezifischen Bildungspläne entwickelt. Um deren Varianz gerecht zu werden, wurden auch Unterrichtsstunden konzipiert, dessen Kernkompetenzen nur in einzelnen Bundesländern gefordert sind. In der folgenden Tabelle findest Du eine Übersicht über die Unterrichtsstunden. Aus ihr kannst Du entnehmen, welche Unterrichtsstunden im Bildungsplan Deines Bundeslandes verankert sind.


Zuordnung der Unterrichtseinheiten zu den landesspezifischen Bildungsplänen (Stand: Juli 2024)

Bundesland	Unterrichtseinheit						
	Einführung in die Algorithmen	Sequenzen	Schleifen	Verzweigungen	Pseudocode	Fehlersuche & Testen	Eigenes Spiel
Baden-Württemberg		X	X	X		X	
Bayern	X	X	X	X	X	X	X
Berlin/Brandenburg		X	X	X			
Hamburg	X	X	X	X		X	
Hessen	X	X	X	X	X	X	X
Mecklenburg-Vorpommern	X	X	X	X	X	X	X
Niedersachsen	X	X	X	X	X	X	X
Nordrhein-Westfalen	X	X	X	X	X	X	X
Rheinland-Pfalz	X	X	X	X	X	X	X
Saarland	X	X	X	X			X
Sachsen	X	X	X	X	X		X
Schleswig-Holstein	X	X	X	X	X	X	
Thüringen	X	X	X	X	X	X	X

Anmerkung: In Bremen gibt es keinen Informatikunterricht. In Sachsen-Anhalt gibt es Informatik im Wahlpflichtbereich nur in höheren Jahrgangsstufen.

Inhaltsverzeichnis

Algorithmen



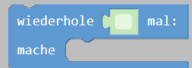
90 Minuten
unplugged

Sequenzen




90 Minuten
plugged

Schleifen




45 Minuten
plugged

Verzweigungen



45 Minuten
plugged

Pseudocode



45 Minuten
unplugged

Seite 6

Fehlersuche & Testen





45 Minuten
plugged

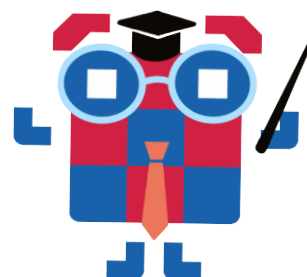
Eigenes Spiel



90 Minuten
plugged

 = plugged

 = unplugged





Pseudocode

Künstler*innen fertigen eine Skizze an, bevor sie ein Bild malen, und Journalist*innen machen sich Stichpunkte, ehe sie einen Artikel schreiben. Auch in der Programmierung ist eine solche Planung des Programms möglich. Dabei wird häufig Pseudocode verwendet. In dieser Unterrichtsstunde lernen die Schüler*innen Pseudocode kennen und schreiben eigenen Pseudocode in der deutschen Variante der Programmiersprache **Pascal**.

Anknüpfung an Bildungspläne

Bayern (Informatik, Jgs. 5/6), **Hessen** (Wahlfach Informatik, Jgs. 7), **Mecklenburg-Vorpommern** (Informatik und Medienbildung, Jgs. 6), **Niedersachsen** (Informatik, Jgs. 5 – 7), **Nordrhein-Westfalen** (Informatik, Jgs. 5/6), **Rheinland-Pfalz** (IPS, Jgs. 5/6), **Sachsen** (Informatik, Jgs. 8), **Schleswig-Holstein** (Informatik, Jgs. 5 – 7), **Thüringen** (Medienbildung und Informatik, Jgs. 5/6)

Überfachliche Kompetenzen

Die Schülerinnen und Schüler...

- ... sind motiviert, Neues zu lernen und Dinge zu verstehen, strengen sich an, um sich zu verbessern.
- ... gehen beim Lernen strukturiert und systematisch vor.
- ... entwickeln eine eigene Meinung, treffen eigene Entscheidungen und vertreten diese gegenüber anderen.
- ... zeigen Toleranz und Respekt gegenüber anderen und gehen angemessen mit Widersprüchen um.

Fachliche Kompetenzen

Die Schülerinnen und Schüler...

- ... beschreiben die Idee eines gegebenen Algorithmus.
- ... stellen Algorithmen in verschiedenen Darstellungsformen dar.
- ... skizzieren die Idee eines Programms mithilfe von Pseudocode (deutsches Pascal).

Neue Bausteine

Es werden keine neuen Bausteine eingeführt.

Weitere verwendete Bausteine

Schleifen: **Wiederhole fortlaufend**

Kontrolle: **Wenn ... dann ... sonst**

Vorbereitung

Mache Dich mit der Thematik **Pseudocode** vertraut. Dazu kannst Du beispielsweise das *Skript des Studienvorkurs Informatik der Universität Osnabrück*¹ hinzuziehen.

Schreibe den folgenden Pseudocode zum Thema **Waffeln backen** an die Innenseite der Tafel und klappe sie zu:

Algorithmus 1 : Tafelbild für die Erarbeitungsphase zum Thema Pseudocode.

Starte das Programm

Heize Waffeleisen vor

Rühre Teig an

Wiederhole fortlaufend

wenn *Ist noch Teig übrig?* **dann**

 Fette Waffeleisen ein

 Fülle Teig in Waffeleisen

 Warte bis Waffel fertig

 Nimm Waffel heraus

sonst

 Schalte das Waffeleisen aus

 Serviere die Waffeln

 Iss die Waffeln

 Beende das Programm

Versuche für Dich einen Weg zu finden, wie Du die verschiedenen Befehle an der Tafel voneinander abgrenzen kannst. Was hier durch **fett** oder *kursiv* hervorgehoben ist, kannst Du an der Tafel beispielsweise durch Unterstreichen, Kästen oder Schreibschrift vom Rest des Pseudocodes abheben.

¹Quelle: Brinkmeier; Haldenwang (2018/19): *Skript. Studienvorkurs Informatik*. Universität Osnabrück: <https://www.informatik.uni-osnabrueck.de/fileadmin/documents/Arbeitsgruppen/Didaktik/Vorkurs/Skript.pdf> (zuletzt aufgerufen am 29.05.2024).

Unterrichtsverlaufsplan

Zeit	Phase	Unterrichtsschritte	SF	Material
5	Einstieg	Beispiele für Skizzierungen von Projekten	P	
10	Erarbeitung	Betrachtung eines Pseudocodes	P	<input type="checkbox"/> Tafelbild
20	Arbeitsphase	Bearbeitung AB Bens Tag	EA/ PA	<input type="checkbox"/> AB Bens Tag <input type="checkbox"/> ggf. KV Freitext
10	Reflexion	Thematisierung von Herausforderungen und Erkenntnissen	P	

EA = Einzelarbeit, GA = Gruppenarbeit, PA = Partnerarbeit, P = Plenum,
S = Sitzkreis, SF = Sozialform

Einstieg

Überlegt gemeinsam, wie man ein Programm planen und skizzieren kann. Führt euch weitere Beispiele für Planungsphasen vor Augen. Beispielsweise machen sich Journalist*innen Stichpunkte, bevor sie einen Artikel schreiben, und Künstler*innen fertigen vor ihrem Kunstwerk eine Skizze an.

Stellt euch die Frage, wie Stichpunkte oder Skizzen für ein Programm aussehen könnten.

Erarbeitung

Klappe nun die Tafel auf und zeige den Schüler*innen den Pseudocode, den Du im Vorfeld dort notiert hast. Lass ein Kind den Code vorlesen. Schreibe **Pseudocode** als Überschrift über das Tafelbild. Frage die Schüler*innen, was sie glauben, was es mit **Pseudocode** auf sich hat. Kennen sie noch weitere Begriffe mit dem Präfix **Pseudo**? Mit welchem Wörtern aus der Alltagssprache kann man es übersetzen? **Pseudo** bedeutet soviel wie **scheinbar** oder **unecht**.

Interpretiert den Pseudocode gemeinsam und formuliert ihn in vollständigen Sätzen aus. Besprecht die Struktur des Pseudocodes. Sprecht über folgende Auffälligkeiten:

- Anweisungen, die in eine andere Anweisung eingebettet sind, werden eingerückt.
- Bedingungen werden in Klammern geschrieben.

Markiert die Auffälligkeiten im Pseudocode.

Arbeitsphase

In der Arbeitsphase bearbeiten die Schüler*innen das Arbeitsblatt **Bens Tag**. Dazu lesen sie den Text durch und markieren sich zunächst alle Aussagen, die für den Pseudocode informativ sind. Danach schreiben sie den Pseudocode zu dem Text auf. Dieser kann wie in Algorithmus 2 aussehen.

Algorithmus 2 : Beispiellösung für das Arbeitsblatt **Bens Tag**.

```

Starte das Programm
Ziehe Badesachen an
wenn Ist es ein kalter Tag? dann
  | Ziehe Bademantel an
Wiederhole fortlaufend
  | Gehe um Beckenrand herum
  | wenn Springt jemand vom Beckenrand? oder Rennt jemand am Beckenrand?
  | dann
  | | Pfeifen
  | | Person ermahnen
  | wenn Gibt es einen Notfall? dann
  | | Ins Wasser springen
  | | Zur Person schwimmen
  | | Person aus Wasser ziehen
  | | wenn Ist die Person ansprechbar? dann
  | | | Person beruhigen
  | | | Person fragen, was sie braucht
  | | sonst
  | | | Notruf wählen
  | | | Erste Hilfe leisten
  | wenn Ist Feierabend? dann
  | | Ziehe Alltagskleidung an
  | | Kollegen verabschieden
  | | Nach Hause gehen
  | | Beende das Programm

```

Lass die Schüler*innen, die einen Pseudocode zum Arbeitsblatt **Bens Tag** verfasst haben, in Partnerarbeit gehen. Dort vergleichen sie ihre Arbeitsergebnisse. Dabei werden sie wahrscheinlich feststellen, dass ihr Pseudocode nicht identisch ist, da sie beispielsweise unterschiedliche Formulierungen gewählt haben. Animiere die Schüler*innen dazu, Vermutungen aufzustellen, wie es zu den Unterschieden kam, und was das für die Umsetzung eines Pseudocodes in ein richtiges Programm bedeuten kann.



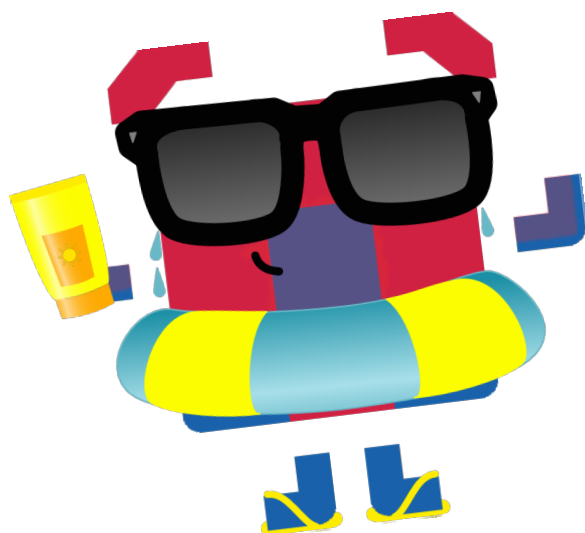
Schnelle Kinder können als Vertiefung überlegen, ob sie das Programm in der Lernsoftware **Cubi** umsetzen könnten. Die Umsetzung ist mit dem derzeitigen Wissensstand der Schüler*innen schwierig, da der Befehlssatz in Cubi begrenzt ist. Auch kannst Du zwei Schüler*innen mit unterschiedlichen Arbeitsergebnissen ihre Pseudocodes an die Tafel schreiben lassen, sodass ihr sie in der Reflexionsphase im Plenum vergleichen könnt.

Reflexion

Zum Abschluss der Stunde können die Schüler*innen ihre Arbeitsergebnisse vorstellen und darauf eingehen, was gut geklappt hat oder was ihnen schwer gefallen ist. Wenn ihr zwei Pseudocodes an die Tafel geschrieben habt, stellt diese in den direkten Vergleich. Welche Unterschiede gibt es in den Pseudocodes der Kinder? Was sind ihre Vermutungen, wie diese zustande kamen, und welche Herausforderungen bringen sie mit sich? Als Fazit kann gezogen werden, dass Pseudocode helfen kann, Gedanken zu sortieren und zu visualisieren. Es ist allerdings kein Garant dafür, dass der Code in der Programmiersprache auch umgesetzt werden kann. Beende die Stunde mit einem Ausblick auf das nächste Thema.

Geschafft!

Großartig, Du hast es durch **Teil 1 der Lernreihe** geschafft! Was eine tolle Leistung!



Jetzt kannst Du Dich zurücklehnen, während Deine Klasse fleißig programmiert.

Baustein-Lexikon

Start

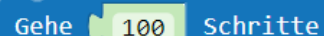
Der erste Baustein eines Blocks aus mehreren Bausteinen ist immer ein **Starbaustein**. Ein Programm einer Figur kann beliebig viele **Startbausteine** haben. **Startbausteine** zeichnen sich durch die Rundung am oberen Teil aus. Diese sagt aus, dass **Startbausteine** nicht an andere Bausteine angehängt werden können.



Wenn Start ▶ geklickt

Der **Startbaustein Wenn Start geklickt wurde** ist der erste Baustein, den die Schüler*innen kennenlernen. Nachfolgende Bausteine werden nacheinander ausgeführt, unmittelbar nachdem das Level gestartet ▶ wurde.

Bewegung



Gehe 100 Schritte

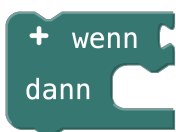
Der Baustein **Gehe ... Schritte** bewegt die Figur die entsprechende Anzahl an Pixel in die aktuelle Richtung der Figur. Im Normalfall ist dies bei Programmstart nach rechts.



Drehe rechts um 90 Grad

Mit dem Baustein **Drehe rechts/links um ... Grad** dreht sich die Figur in die ausgewählte Richtung um die entsprechende Gradzahl. In den ersten Leveln brauchen die Schüler*innen nur den rechten Winkel.

Kontrolle



+ wenn dann

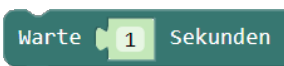
Der Baustein **Wenn dann** leitet eine Verzweigung ein. Oben an das **wenn** wird eine Bedingung angepuzzelt. Diese ist entweder **wahr** oder **falsch**. Wenn die Bedingung wahr ist, werden die Bausteine, die neben dem **dann** stehen ausgeführt. Ist die Bedingung falsch, also nicht erfüllt, werden die Bausteine bei **dann** übersprungen und nicht ausgeführt. Drückt man oben links auf dem Baustein auf das weiße Plus, wird ein neuer Verzweigungsarm hinzugefügt, an den eine

weitere Bedingung angepuzzelt werden kann. Diese wird jedoch nur überprüft, wenn die erste Bedingung falsch war.



Der Baustein **Wenn dann sonst** leitet eine Verzweigung ein. Oben an das **wenn** wird eine Bedingung angepuzzelt. Diese ist entweder **wahr** oder **falsch**. Wenn die Bedingung wahr ist, werden die Bausteine, die neben dem **dann** stehen ausgeführt. Die Bausteine hinter **sonst** werden übersprungen. Ist die Bedingung falsch, also nicht erfüllt, ist es genau andersherum und die Bausteine bei **dann** werden übersprungen und an ihrer Stelle werden die Bausteine, die hinter **sonst** stehen, ausgeführt.

Drückt man oben links auf dem Baustein auf das weiße Plus, wird ein neuer Verzweigungsarm hinzugefügt, an den eine weitere Bedingung angepuzzelt werden kann. Diese wird jedoch nur überprüft, wenn alle vorherigen Bedingungen falsch waren.



Gelangt ein Programm zu einem **Warte**-Baustein, dann bleibt es hier für die Anzahl der eingegebenen Sekunden stehen. Andere Programmteile der Figur, die ihren eigenen **Startbaustein** haben, werden hierdurch nicht unterbrochen. Erst wenn die Zeit um ist, wird der nächste Baustein ausgeführt.

Fühlen

Bausteine der Kategorie **Fühlen** werden als Bedingungen in **Verzweigungen** oder **Schleifen mit Bedingungen** angepuzzelt. Das Programm prüft, ob die Bedingung **wahr** oder **falsch** ist. Ist die Bedingung wahr, werden die Bausteine in der **Verzweigung** oder **Schleife** ausgeführt.



Mit diesem Baustein wird geprüft, ob die Figur eine bestimmte Farbe berührt. Durch Klicken auf das Farbfeld kann die Farbe geändert werden, die geprüft wird.

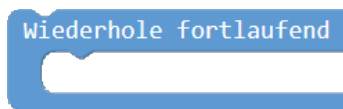


Mit diesem Baustein kann die Figur reagieren, wenn sie etwas berührt. Durch Klicken auf den kleinen Pfeil kann ausgewählt werden, ob die Figur auf den Rand oder eine andere Figur reagieren soll, wenn es noch weitere Figuren in dem Level gibt.

Schleifen



Mit der **Wiederhole ... mal**-Schleife können Bausteine, die in die Schleife eingefügt werden, wiederholt werden. Klicke auf die Zahl, um die Anzahl der Wiederholungen zu ändern.



Alle Bausteine, die in die **Wiederhole fortlaufend**-Schleife eingefügt werden, werden endlos lange wiederholt. Das Level stoppt erst, wenn man es über die Pause-Taste unterbricht oder über den Zurücksetzen-Button stoppt. Unter dieser Schleife können keine Bausteine angehängt werden, weil diese nie ausgeführt würden.

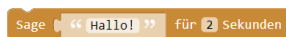
Aussehen



Mithilfe dieses Bausteins wechselt die entsprechende Figur das Kostüm und ändert so ihr Aussehen. Die Kostüme sind unterhalb der Bühne zu sehen. Hier wird zu dem Kostüm gewechselt, welches in dem Feld ausgewählt wurde.



Mithilfe dieses Bausteins wechselt die entsprechende Figur das Kostüm und ändert so ihr Aussehen. Die Kostüme sind unterhalb der Bühne zu sehen. Hier wird zu dem Kostüm gewechselt, welches unter der Bühne nach dem aktuellen Kostüm abgebildet ist.



Wenn der **Sage**-Baustein ausgeführt wird, erscheint für die Figur eine Sprechblase. Dort wird der Text angezeigt, welcher in das Textfeld geschrieben wurde. Nach der angegebenen Zeit verschwindet die Sprechblase und der nächste Baustein wird ausgeführt.

KV Kunstgalerie

IT 4 KIDS

Schreibe eine Anleitung für diese **Tür**.



Schreibe eine Anleitung für diese **Brille**.



Schreibe eine Anleitung für

Schreibe eine Anleitung für diesen **Menschen**.



Schreibe eine Anleitung für diese **Sonne**.



Schreibe eine Anleitung für

Alle Arbeitsblätter und Kopiervorlagen zu diesem Modul findest Du auf der Webseite von IT 4 KIDS: material.i4k.org/ab

Astrid ist Astronautin. Heute steht ihr großer Raketenstart an. Endlich kann der Versuch starten!

Bevor sie die Rakete betritt, schlüpf sie in ihren Raumanzug. Dann geht sie ins Cockpit und schnallt sich an.

Damit Astrid weiß, dass sie die Rakete starten kann, muss ein grünes Licht aufleuchten. Deshalb fragt sie, ob alles startklar ist.

Wenn das grüne Licht nicht erscheint, wartet Astrid zwei Minuten und fragt dann nochmal nach. Das macht sie so lange, bis sie grünes Licht bekommt und starten kann.

Bei grünem Licht atmet sie nochmal tief durch und startet dann die Rakete.

Damit ist ihr Job für den Raketenstart getan.



Dieses Material ist lizenziert unter CC BY-NC-SA 4.0. Weitere Informationen findest Du hier: <http://creativecommons.org/licenses/by-nc-sa/4.0/> © 0 0 0

IT 4 KIDS
 Namen: _____
 Datum: _____
Aufgabe: Schreibe eine Anleitung zu deinem Bild. Sei so genau wie möglich, damit ein anderes Kind das Bild malen kann. Das Bild ist gelblich, lass es niemanden sehen!

IT 4 KIDS
 Namen: _____ Datum: _____
Aufgabe: Befolge die Anleitung deines Partners (als das gelbliche Bild)?
 Das habe ich gemacht: _____

IT 4 KIDS
 Namen: _____ Datum: _____
Eigenes Level 1

Worum geht es in dem Level?
 Thema: _____
 Ort: _____
 Geschichte: _____

Um wen geht es in dem Level?
 Name: _____ Name: _____
 Aufgabe: _____ Aufgabe: _____
 Steuerbar: Nein Ja, und zwar so: _____
 Steuerbar: Nein Ja, und zwar so: _____

Aussehen: _____ **Aussehen:** _____

Dieses Material ist lizenziert unter CC BY-NC-SA 4.0. Weitere Informationen findest Du hier: <http://creativecommons.org/licenses/by-nc-sa/4.0/> © 0 0 0

Weitere Angebote von IT4Kids

Du willst weiter mit IT4Kids arbeiten? Mach gerne mit **Teil 2 der Levelreihe** weiter! Hier kannst Du gemeinsam mit Deinen Schüler*innen tiefer in die Welt der Algorithmen eintauchen und lernst unter anderem **Variablen**, **verschachtelte Verzweigungen** und **Funktionen** kennen.

Außerdem bieten wir neben einer Modulreihe zu den Themen **Sequenzen**, **Schleifen** und **Verzweigungen** mit der Zielgruppe **Primarstufe** auch Fortbildungen für Dich und Dein Kollegium an. Hier lernen wir gemeinsam den Cubi-Editor kennen, sammeln grundlegende Programmiererfahrungen und planen eine erste Unterrichtsstunde mit Cubi speziell für Deine Klasse. Nach der Fortbildung kannst Du direkt am nächsten Tag eine Stunde Programmierung mit Deiner Klasse ausprobieren, weil wir alles gemeinsam in der Fortbildung vorbereitet haben. Alle aktuellen Informationen zu unserem Fortbildungsangebot findest du auf unserer Webseite unter www.i4k.org/fortbildung. 😊

Du bist noch unschlüssig? Dann schau Dich gerne auf unserer Webseite www.i4k.org um, stöbere durch unser Material und lerne uns ein bisschen besser kennen. Wenn auf dem Weg Fragen aufkommen oder Du mit uns ins Gespräch über die Materialien kommen möchtest, dann melde Dich gerne per E-Mail unter info@it-for-kids.org bei uns oder nimm über unsere Webseite www.i4k.org/kontakt mit uns Kontakt auf. 💬

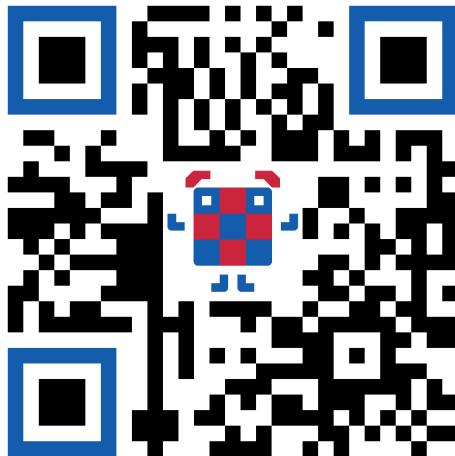
Wir freuen uns auf Dich! 😊



Hilf uns, besser zu werden!

Dir sind Fehler in dem Material aufgefallen?
Du hast Verbesserungsvorschläge?
Du möchtest mehr zum Einstieg in die Programmierung?

Wir freuen uns über Dein Feedback:



feedback.i4k.org/lk/sek1/teil1