

FRANZIS

Programmieren

lernen

mit

Scratch

So wirst du ein echter **Scratcher** der weltweiten **Scratch-Community**

CHRISTIAN IMMLER

Bibliografische Information der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

Hinweis: Alle Angaben in diesem Buch wurden vom Autor mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einhaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. Der Verlag und der Autor sehen sich deshalb gezwungen, darauf hinzuweisen, dass sie weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernehmen können. Für die Mitteilung etwaiger Fehler sind Verlag und Autor jederzeit dankbar. Internetadressen oder Versionsnummern stellen den bei Redaktionsschluss verfügbaren Informationsstand dar. Verlag und Autor übernehmen keinerlei Verantwortung oder Haftung für Veränderungen, die sich aus nicht von ihnen zu vertretenden Umständen ergeben. Evtl. beigefügte oder zum Download angebotene Dateien und Informationen dienen ausschließlich der nicht gewerblichen Nutzung. Eine gewerbliche Nutzung ist nur mit Zustimmung des Lizenzinhabers möglich.

Der Autor

Christian Immler, Jahrgang 1964, war bis 1998 als Dozent für Computer Aided Design an der Fachhochschule Nienburg und an der University of Brighton tätig. Einen Namen hat er sich mit diversen Veröffentlichungen zu Spezialthemen wie 3D-Visualisierung, Smartphone-Betriebssysteme, Linux und Windows gemacht. Seit mehr als 20 Jahren arbeitet er als erfolgreicher Autor mit mehr als 200 veröffentlichten Computerbüchern.

© 2020 FRANZIS Verlag GmbH, 85540 Haar bei München, komplett durchgesehene und aktualisierte Ausgabe.

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Das Erstellen und Verbreiten von Kopien auf Papier, auf Datenträgern oder im Internet, insbesondere als PDF, ist nur mit ausdrücklicher Genehmigung des Verlags gestattet und wird widrigenfalls strafrechtlich verfolgt.

Die meisten Produktbezeichnungen von Hard- und Software sowie Firmennamen und Firmenlogos, die in diesem Werk genannt werden, sind in der Regel gleichzeitig auch eingetragene Warenzeichen und sollten als solche betrachtet werden. Der Verlag folgt bei den Produktbezeichnungen im Wesentlichen den Schreibweisen der Hersteller.

Lektorat: Ulrich Dorn

Satz: PC-DTP-Satz und Informations GmbH, Alexandra Kugge

Covergestaltung: Julia Harrer

Druck: Graspö CZ, a.s.

ISBN: 978-3-645-60672-1



Inhalt

So geht Scratch.....	6
1 Scratch installieren oder einfach nutzen.....	10
Scratch 3.0 im Browser	10
Scratch Desktop offline nutzen	11
Scratch 3.0 auf dem Raspberry Pi	12
2 Das erste eigene Spiel.....	14
Der Hintergrund	14
Der Ball.....	16
Der Schläger.....	20
Die Spielregeln	24
Punkte zählen.....	26
3 Der kleine Hacker und die Bananen.....	32
Der kleine Hacker in Scratch	32
Die Bananen	36
Die Bananen fallen herunter.....	38
Die Bananen fallen noch echter herunter	40
4 Spielwürfel.....	42
Würfel zeichnen.....	42
Das Programm für den Würfel.....	49
5 Space Race - oder auf Deutsche Raumschiffrennen.....	50
Der Sternenhimmel.....	50
Das Raumschiff	52
Die Steuerung.....	55
Kollisionserkennung	57
Der Rundenzähler	59



6 Ein Käfer sucht sich seinen Weg	62
7 Scratch malt Retro-Computergrafiken	66
So entstehen die Grafiken	66
Erweiterungen installieren	67
Das erste Grafikprogramm	68
Winkel über Variable einstellen	72
Winkel interaktiv einstellen	73
Schnellere Grafik.....	75
Der Turbo-Modus	75
8 Musik mit Scratch	76
9 Flappy Bird	82
Der Vogel fliegt	83
Die Rohre kommen.....	86
Kollisionserkennung und Punkte.....	89
10 Labyrinth	92
Das Koordinatensystem des Labyrinths	94
Zeichne das Labyrinth	94
Finde den Weg durch das Labyrinth	101
Automatisch den Weg durch das Labyrinth finden.....	109
11 Analoguhr	114
12 Simon - Senso - Einstein	120
Die Grafik	120
Eigenen Block definieren	122
Das Spiel.....	124
13 Die Scratch-Gemeinschaft	130
Die coolsten Scratch-Projekte und wie sie funktionieren.....	134
14 micro:bit mit Scratch steuern	140
micro:bit zur Verwendung mit Scratch einrichten	140
Einfaches Programm zum Ausprobieren	144
Pong mit dem micro:bit steuern	145



15 Elektronik mit Scratch auf dem Raspberry Pi steuern.....	148
GPIO-Steuerung mit Scratch auf dem Raspberry Pi	149
Speicherkarte zur Betriebssysteminstallation vorbereiten	150
Der erste Start des Raspberry Pi.....	152
Scratch-Programme aus dem Download nutzen	155
Diese Teile brauchst du für die Hardwareexperimente	157
Die erste LED blinkt	160
LED-Würfel mit Scratch	162
16 Das SenseHAT Für den Raspberry Pi.....	166
Joystick und Symbole auf dem SenseHAT	166
Temperatur und Text auf dem SenseHAT.....	167
Für Profis	170
Referenz: Alle Scratch-Blöcke im Überblick.....	170
Referenz: Die Blöcke der wichtigsten Scratch-Add-ons im Überblick.....	184

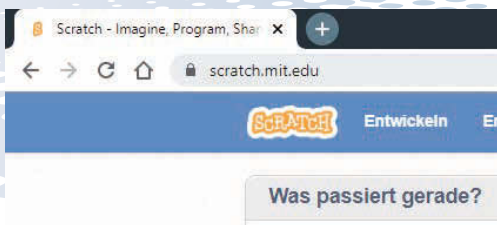
So geht Scratch



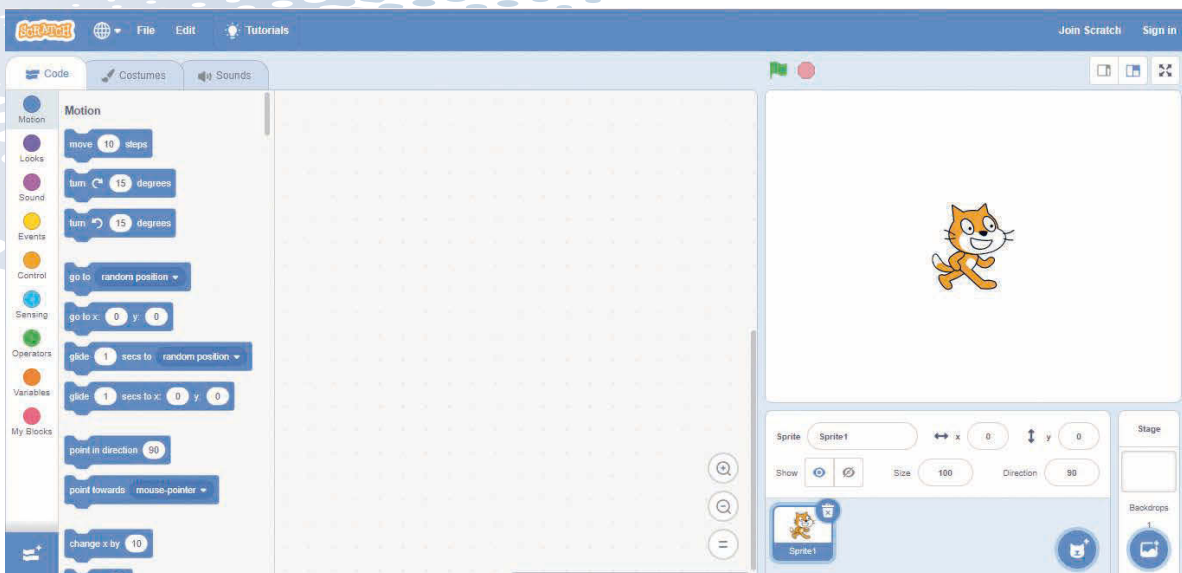
Bevor wir mit den ausführlichen Erklärungen beginnen, wie Scratch funktioniert und was man alles damit machen kann, probiere doch dieses kleine einfache Beispiel aus.

1 Starte einen Webbrowser (z. B. Firefox oder Chrome) auf dem PC und besuche die Webseite scratch.mit.edu.

2 Klicke oben links auf **Entwickeln**. Damit kannst du ein neues Scratch-Programm erstellen.

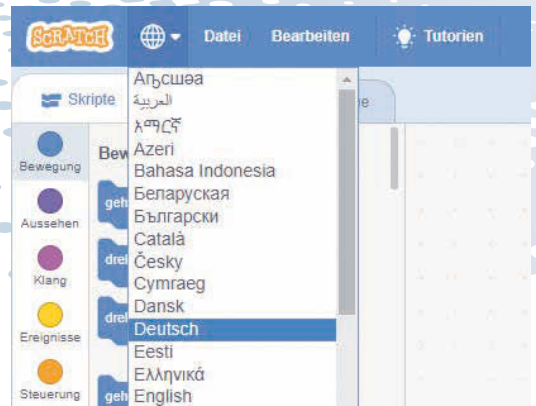


3 Jetzt erscheint die Scratch-Programmieroberfläche mit der typischen Katze, der Symbolfigur von Scratch.

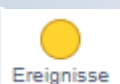


6 Der kleine Hacker – Programmieren für Einsteiger

4 Klicke links oben auf die Weltkugel und wähle **Deutsch** als Sprache für die Benutzeroberfläche.



5 In einem ganz einfachen Skript soll die Katze einmal im Kreis herum laufen und dabei ihre Farbe verändern.



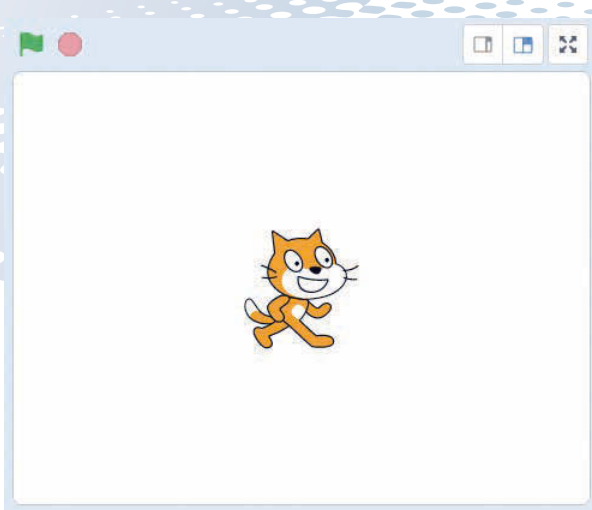


6 Klicke in der Blockpalette ganz links auf das gelbe Symbol **Ereignisse**. Dann werden die Blöcke angezeigt, die auf bestimmte Ereignisse reagieren.

7 Ziehe den abgebildeten Block **Wenn Fahne angeklickt wird** aus der Blockpalette in das Skriptfenster in der Mitte.

Wenn angeklickt wird

8 Auf der Scratch-Bühne ist oben links ein grünes Fähnchen. Es dient üblicherweise dazu, ein Programm zu starten. Die Bühne ist das Fenster rechts oben, in dem die Katze in der Mitte steht.



9 Der Block **Wenn Fahne angeklickt wird** bewirkt, dass die folgenden Blöcke ausgeführt werden, wenn der Benutzer auf das grüne Fähnchen klickt. Der Block ist oben rund, passt also unter keinen anderen Block. Er muss immer als Erstes gesetzt werden.

10 Klicke jetzt in der Blockpalette auf das orangefarbene Symbol **Steuerung**.



11 Die kreisförmige Bewegung der Katze wird aus einzelnen Gehen- und Drehen-Schritten zusammengesetzt. Diese werden so oft wiederholt, bis die Katze einen ganzen Kreis gegangen ist. Ziehe für die Wiederholungsschleife den Block **wiederhole ... mal** in das Skriptfenster und docke ihn unten an den dort bereits vorhandenen Block an. Wenn du zwei Blöcke nahe genug aneinanderziehst, verbinden sie sich automatisch.

12 In jedem Bewegungsschritt soll sich die Katze um 15 Grad drehen. Dabei dreht sie sich in 24 Schritten genau einmal. Tippe in das weiße Zahlenfeld des **wiederhole ... mal**-Blocks und ändere den vorgegebenen Wert auf **24**.

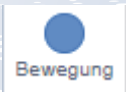


13 Damit die Katze eine Kreisbewegung ausführt, muss sie zunächst einen Schritt gehen, sich danach um 15 Grad drehen, wieder einen Schritt gehen usw. Um eine Bewegung zu

So geht Scratch



programmieren, klicke in der Blockpalette auf das blaue Symbol **Bewegung**.



14 Ziehe den Block **gehe ...er Schritt** in das Skriptfenster und docke ihn innerhalb der Schleife an.



15 Ändere dann noch die Schrittweite auf **20**.



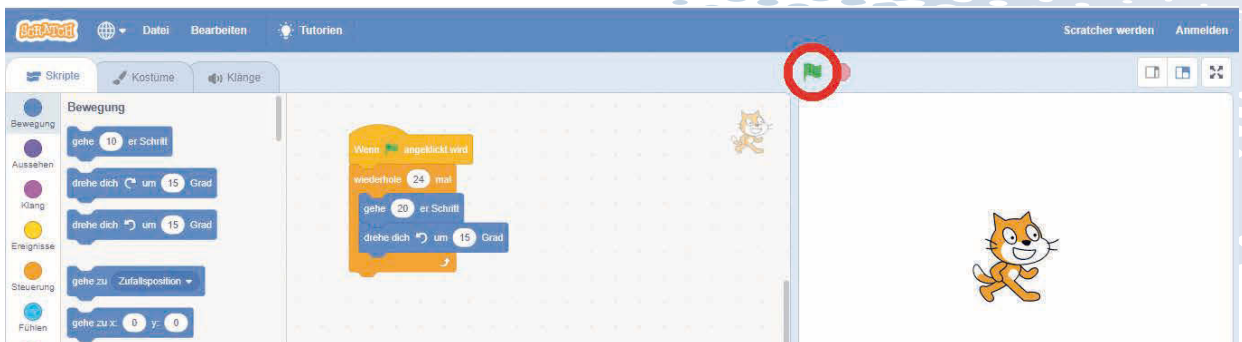
16 Ziehe danach den Block für die Drehbewegung gegen den Uhrzeigersinn in das Skriptfenster.



17 Platziere ihn so über den vorhandenen Blöcken, dass er sich innerhalb der Schleife einklinkt.

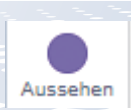


18 Das Skript sollte jetzt wie abgebildet aussehen. Probiere es aus, ob es auch wie erwartet funktioniert. Klicke dazu links oben in der Bühne auf das grüne Fähnchen. Die Katze wird eine Kreisbewegung gehen.





19 Jetzt fehlt nur noch die gewünschte Veränderung der Farbe. Klicke dazu in der Blockpalette auf das dunkelblaue Symbol **Aussehen**. Dann werden Blöcke angeboten, die das Aussehen des aktiven Objekts (der Katze in unserem Fall) beeinflussen.



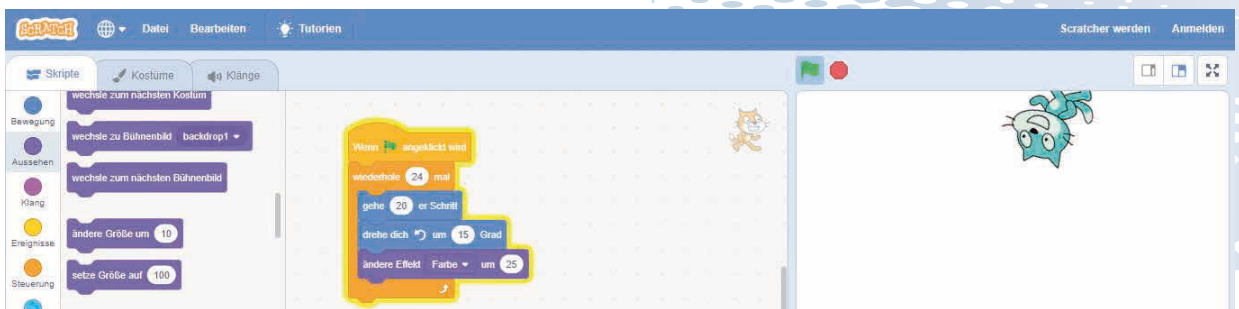
20 Ziehe den abgebildeten Block **ändere Effekt Farbe um ...** in das Skriptfenster in die Wiederholung hinter den **drehe dich um ... Grad**-Block.



21 Damit wird nach jeder Bewegung die Farbe um 25 Einheiten geändert.



22 Lässt du das Skript jetzt wieder ablaufen, ändert die Katze im Laufe ihrer Bewegung zyklisch ihre Farbe. Am Ende der 24 Wiederholungen steht die Katze erneut an ihrer ursprünglichen Position und hat auch wieder ihre ursprüngliche Farbe.





1 Scratch installieren oder einfach nutzen



Was ist Scratch? Fragen wir einfach die Entwickler. Das Scratch-Team beschreibt es so: *„Scratch ist eine Programmiersprache und Onlinegemeinschaft, in der du deine eigenen interaktiven Geschichten, Spiele und Animationen erstellen und deine Werke mit anderen überall auf der Welt teilen kannst. Beim Entwickeln und Programmieren von Scratch-Projekten lernen junge Menschen kreativ zu denken, systematisch vorzugehen und kooperativ mit anderen zusammenzuarbeiten – grundlegende Fähigkeiten für das Leben im 21. Jahrhundert.“*

In dem YouTube-Video *What most Schools don't teach* (Was die meisten Schulen nicht unterrichten) erklären berühmte „Tech-Helden“ wie Mark Zuckerberg (Facebook), Bill Gates (Microsoft) und andere, warum es wichtig ist, programmieren zu können: youtu.be/vwDSNbR-6UE.

Scratch war 15 Jahre lang ein Projekt der Life-long-Kindergartengruppe am MIT-Media-Lab und wurde aufgrund seines großen Erfolgs in eine eigene gemeinnützige Stiftung, die Scratch Foundation, überführt. Es ist und bleibt kostenlos.

Scratch möchte für jeden nutzbar sein, egal welche Sprache du sprichst, welche Hautfarbe oder

Religion du hast, ob du ein Junge oder ein Mädchen bist. Die meisten Scratcher sind übrigens zwischen 11 und 14 Jahre alt. Im vergangenen Jahr haben über 20 Millionen Kinder und Jugendliche auf der ganzen Welt Projekte mit Scratch entwickelt.

Um möglichst viele Menschen anzusprechen, ist Scratch in 50 Sprachen verfügbar und läuft auf vielerlei verschiedenen Computern. Scratch wird in unterschiedlichen Varianten angeboten:

SCRATCH 3.0 IM BROWSER

Am einfachsten funktioniert Scratch im Browser auf einem PC. Du brauchst nichts zu installieren, sondern einfach nur die Seite scratch.mit.edu zu besuchen, und los geht's mit der aktuellsten Version 3.0 von Scratch.

SCRATCH 3.0 IM BROWSER LÄUFT:

- mit den aktuellen Versionen von Firefox (57 und höher), Chrome (63 und höher), Edge (den Chromium-basierten Versionen) auf Windows-PCs mit mindestens 1.024 × 768 Pixeln Bildschirmauflösung,
- mit den aktuellen von Firefox (57 und höher), Chrome/Chromium (63 und höher) auf den meisten Linux-Distributionen auf PCs und auf macOS,
- auf Tablets mit Android (6 und höher) oder Windows 10 mit mindestens 1.024 × 768 Pixeln Bildschirmauflösung,





- auf Chromebooks mit Chrome (63 und höher).

SCRATCH 3.0 IM BROWSER LÄUFT NICHT:

- auf Smartphones,
- auf dem Raspberry Pi,
- im Browser auf Fernsehern und Spielkonsolen,
- mit dem Microsoft Internet Explorer,
- auf Windows-RT-Tablets (nur bestimmte Effekte werden nicht unterstützt).

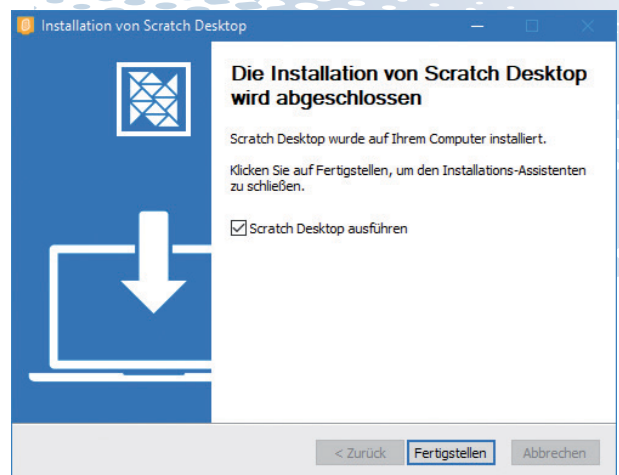
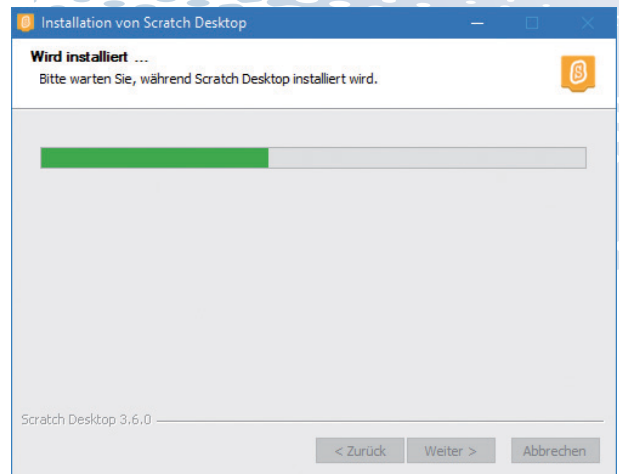
SCRATCH DESKTOP OFFLINE NUTZEN

Nicht überall hat man ständig Internetzugang. Möchtest du zum Beispiel auf einer längeren Bahnfahrt etwas mit Scratch machen, kannst du dir bei scratch.mit.edu/download den Scratch Desktop herunterladen. Viele Schulen haben zwar Computer in Klassenzimmern oder Aufenthaltsräumen, die aber häufig nicht mit dem Internet verbunden sind. Auch dort kann man mit dem Scratch Desktop programmieren.

SCRATCH-3.0-DESKTOP (OFFLINE-EDITOR) INSTALLIEREN

Der Scratch-3.0-Desktop läuft offline unter Windows 10 und macOS (10.13 und höher), auf Android-Tablets (6 und höher) sowie auf aktuellen Chromebooks mit Google-Play-Store-Unterstützung, nicht aber unter Linux, nicht auf Smartphones und nicht unter iOS.

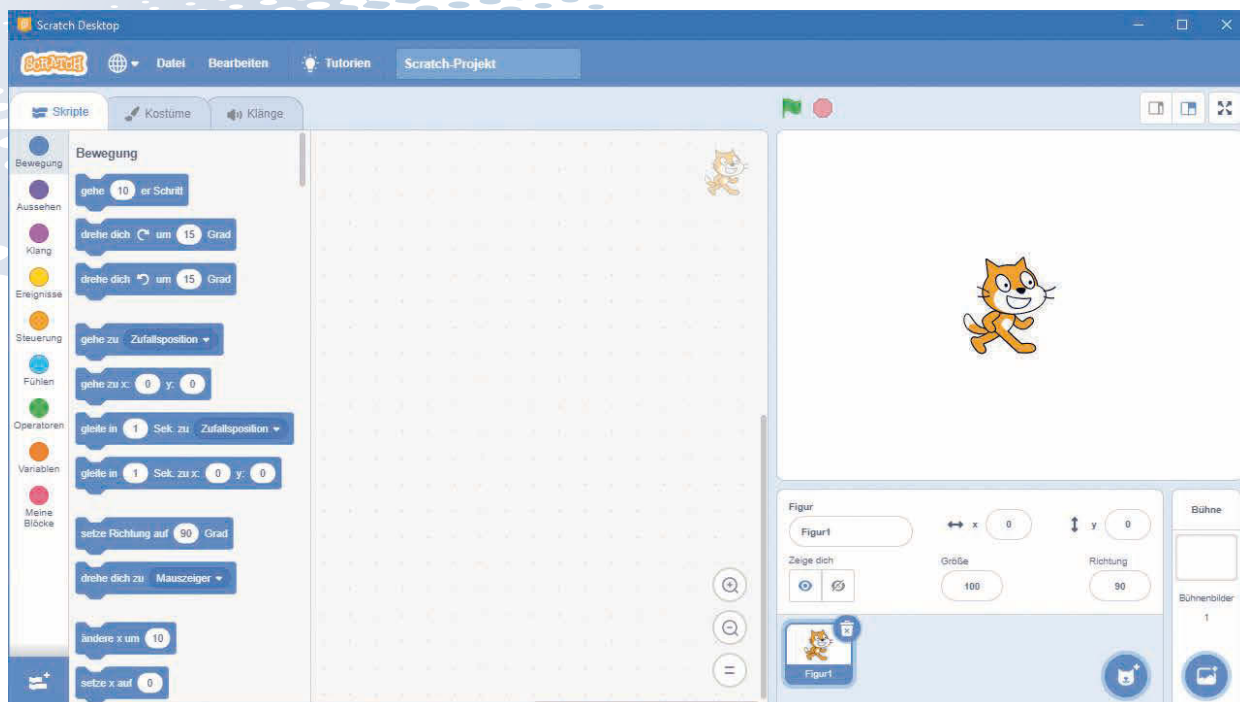
Installiere die heruntergeladene Datei auf deinem PC.





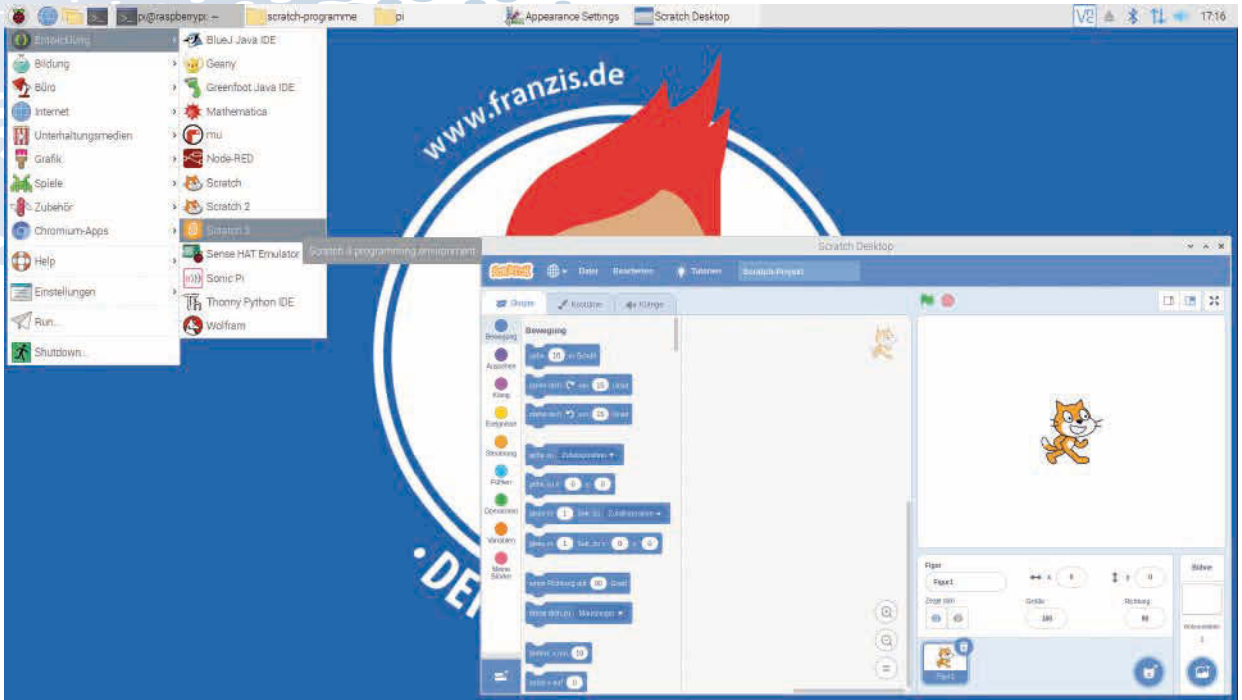
Scratch installieren oder einfach nutzen

Nach der Installation kannst du den Scratch Desktop ohne Internetverbindung nutzen.



SCRATCH 3.0 AUF DEM RASPBERRY PI

Auf dem Minicomputer Raspberry Pi ist die aktuelle Version Scratch 3.0 bereits im Raspbian-Betriebssystem vorinstalliert. In der neuesten Raspbian-Version gibt es standardmäßig keine Desktopsymbole mehr. Wähle hier [Scratch 3.0](#) aus dem Menü [Entwicklung](#). Der Raspberry Pi verwendet den Scratch Desktop ohne Browser. Diese Software kann offline ohne Internetverbindung genutzt werden.



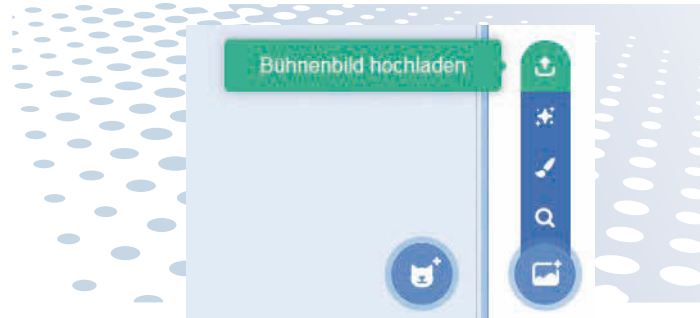
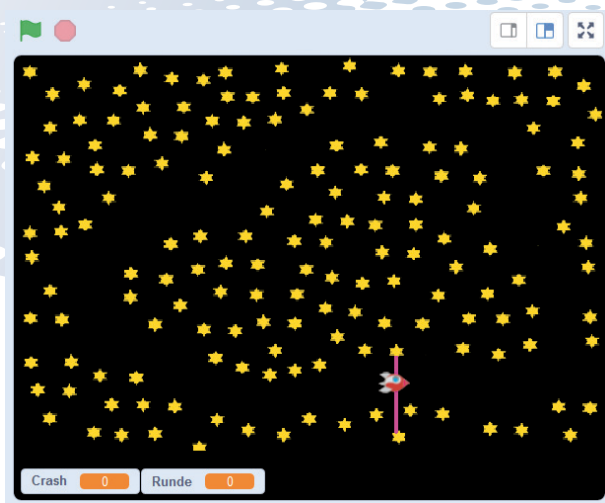
Scratch 3.0 läuft auf dem aktuellen Raspberry Pi 4 sehr flüssig. Für ältere Modelle sind auch noch die früheren Versionen Scratch 2.0 und Scratch 1.4 verfügbar. Scratch im Browser läuft auf dem Raspberry Pi nicht.

Wie Scratch auf dem Raspberry Pi genau funktioniert und wie du damit Elektronik steuern kannst, wird dir am Ende dieses Buchs beschrieben.

5 Space Race - oder auf Deutsch: Raumschiffrennen




Ein Raumschiff rast durchs All und darf dabei an keinem Stern anstoßen. Wie viele Runden mit möglichst wenigen Crashes schaffst du? In diesem Projekt programmieren wir ein schnelles Rennspiel.

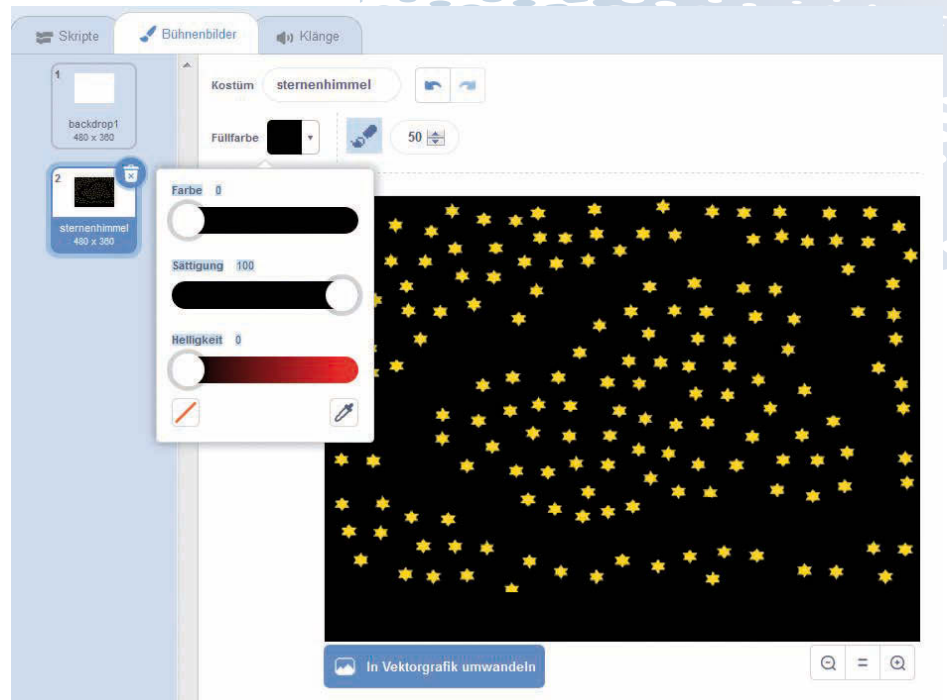
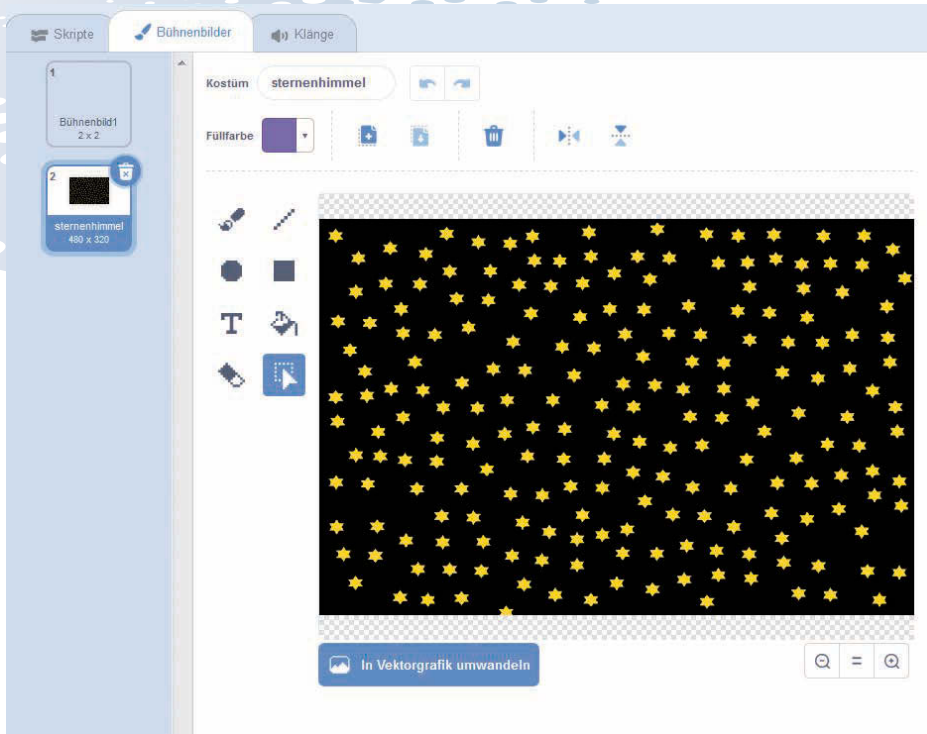


- 2 Der Sternenhimmel erscheint als Bühnenbild und öffnet sich gleichzeitig im Grafikprogramm.
- 3 Der Sternenhimmel ist gleichmäßig mit Sternen übersät. Übermale einige Sterne mit einem dicken schwarzen Pinsel, um Platz für einen Rundkurs zu machen, auf dem das Raumschiff fliegen kann. Wähle dazu das Pinselsymbol und stelle die Breite auf etwa **50**.
- 4 Lösche gleich noch die Katze, sie kommt in diesem Spiel nicht vor.

DER STERNENHIMMEL

 Bei den Downloads zum Buch findest du ein Bild **sternenhimmel.png**, das du als Hintergrund verwenden kannst.

 Starte ein neues Projekt in Scratch, klicke in der Figurenpalette auf die Bühne und fahre mit der Maus auf das Symbol **Bühnenbild wählen**. Klicke dann auf **Bühnenbild hochladen**. Wähle hier den Sternenhimmel aus.



5 Space Race - oder auf Deutsch: Raumschiffrennen



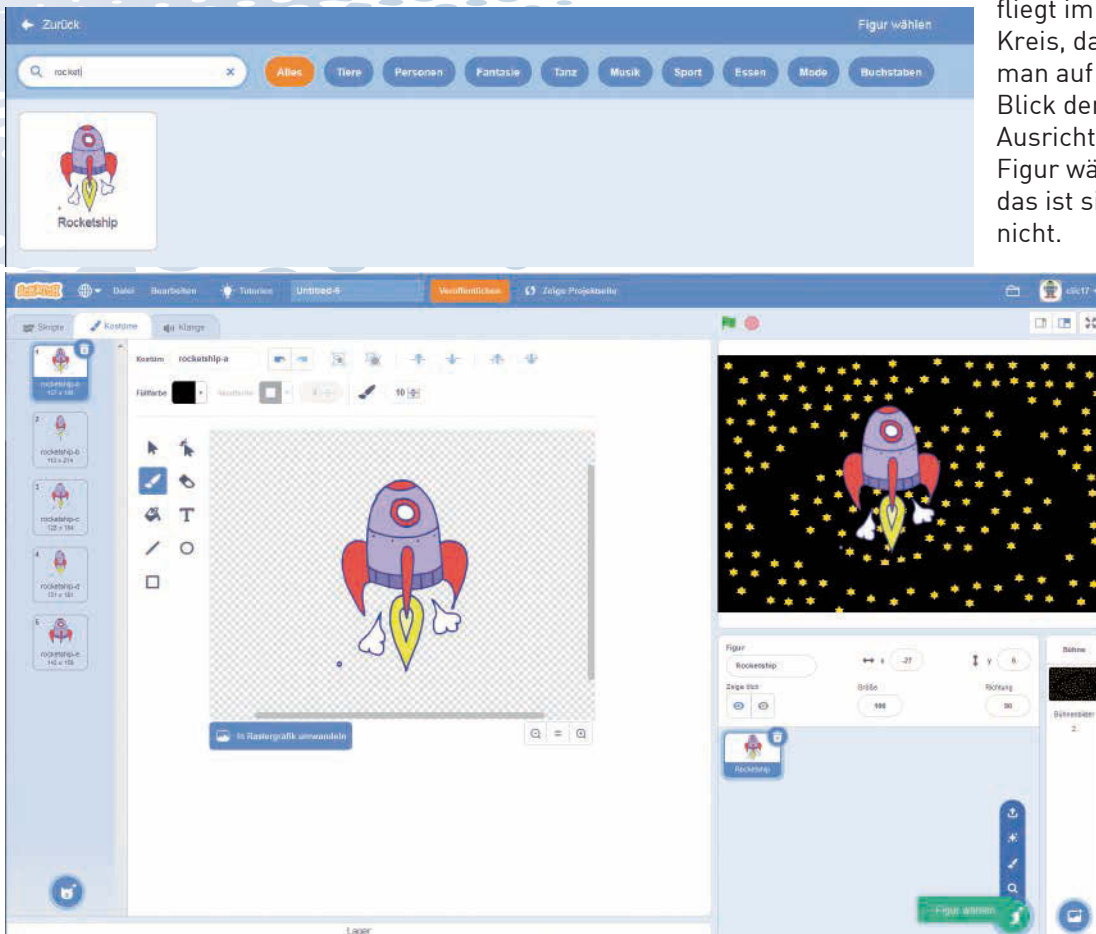
DAS RAUMSCHIFF

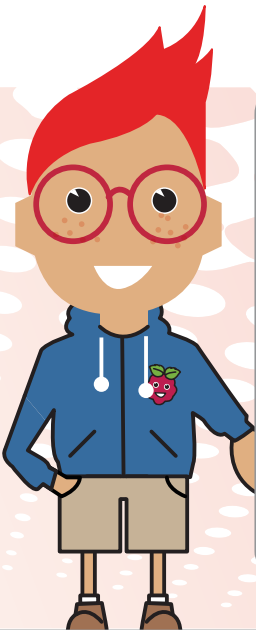
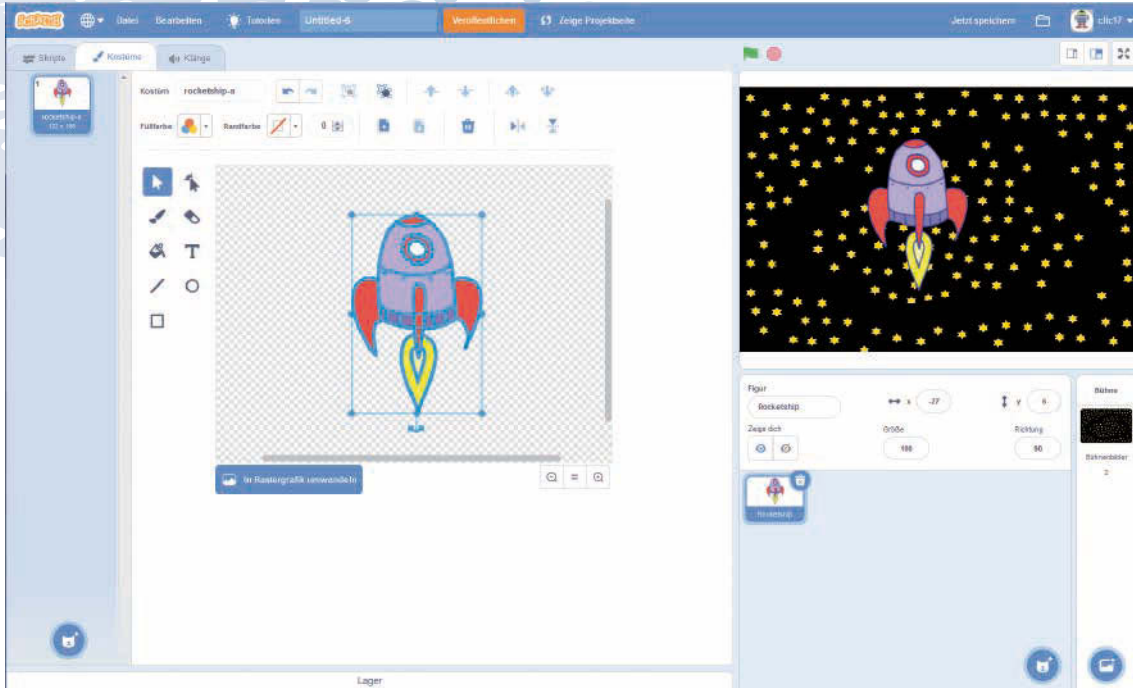
- 1 Scratch liefert bereits ein Raumschiff mit, das sich für dieses Spiel leicht modifizieren lässt. Klicke auf das Symbol **Figur wählen**. Suche im Suchfeld die Figur **Rocketship**.
- 2 Das Raumschiff ist als Vektorgrafik gezeichnet. Lösche alle Kostüme des Raumschiffs mit Ausnahme des ersten, da wir sie in diesem Spiel nicht brauchen.

3 Das übrig gebliebene Kostüm besteht aus zwei getrennten Objekten. Wähle die Wolken aus und lösche sie mit dem Papierkorbsymbol. Jetzt ist das Raumschiff nur noch ein Objekt, das sich leichter bearbeiten lässt.

4 Das Raumschiff ist für unser Spiel viel zu groß. Ziehe es an einer der Ecken kleiner, bis es auf der Bühne auf den zu fliegenden Kurs passt. Fasse es jetzt an dem runden Griff unten mit der Maus an und drehe es in die Waagerechte mit Flugrichtung nach rechts. Das Raumschiff

fliegt im Spiel im Kreis, daher mag man auf den ersten Blick denken, die Ausrichtung der Figur wäre egal – das ist sie aber nicht.



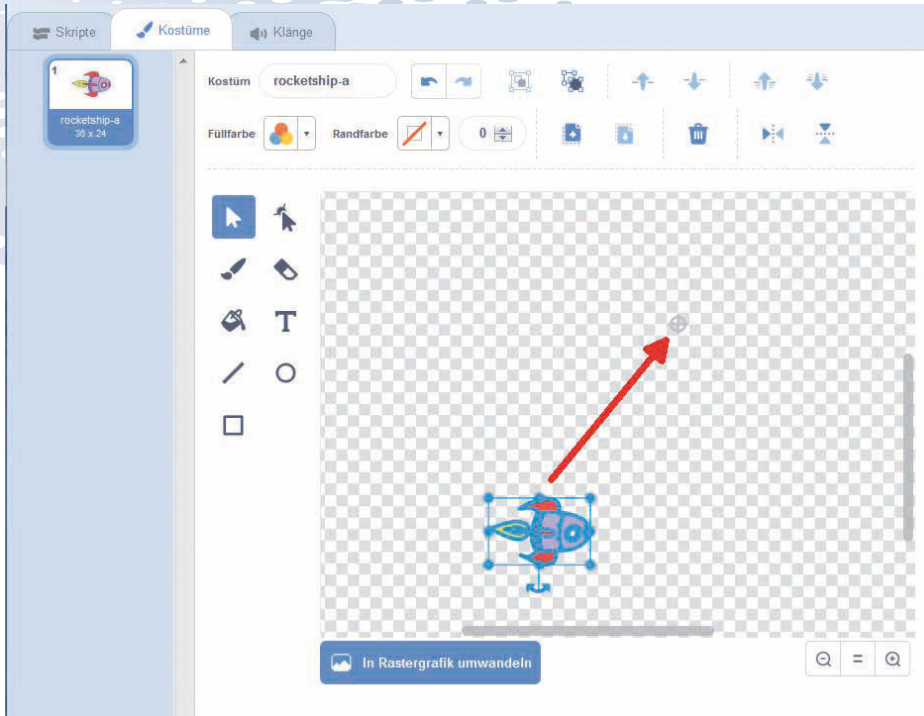


BEWEGUNGEN UND RICHTUNGEN IM SCRATCH-KOORDINATENSYSTEM

Die Bewegungsrichtung einer Figur zeigt standardmäßig immer nach rechts, wie an der Katze zu sehen ist. Das Raumschiff würde sich also seitlich zur eigentlichen Flugrichtung bewegen. Indem du das Kostüm drehst, kann sich das Raumschiff weiterhin nach rechts bewegen, es sieht aber aus, als würde es richtig fliegen. Drehst du dann die ganze Figur auf der Scratch-Bühne, kann das Raumschiff auch nach oben und in alle anderen Richtungen fliegen, da das interne Koordinatensystem der Figur mit gedreht wird.

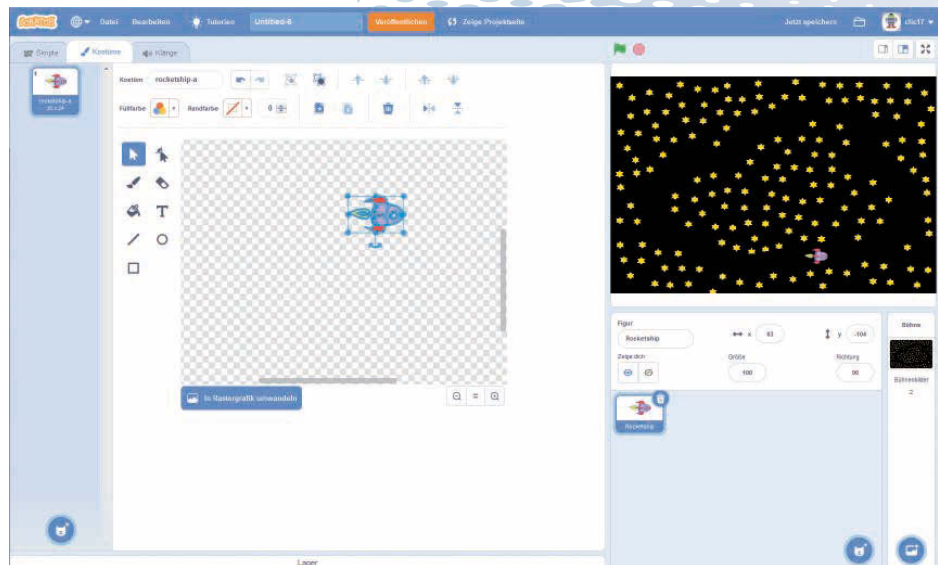
Beim Drehen ist außerdem der Drehpunkt wichtig, der für jede Figur festgelegt ist. Er befand sich vor der Verkleinerung mitten im Raumschiff und ist jetzt nach außen gerutscht. Er wird als kleines Kreuz im Grafikfenster dargestellt.

5 Space Race - oder auf Deutsch: Raumschiffrennen



5 Verschiebe das Raumschiff so, dass sein Mittelpunkt auf dem im Grafikfenster gezeigten Drehpunkt liegt. Eventuell musst du etwas zoomen, um den Drehpunkt zu finden.

6 Ziehe das Raumschiff auf der Bühne an die Startposition. In der Figurenliste siehst du die Koordinaten und kannst auch die Drehrichtung einstellen.





DIE STEUERUNG

Das Raumschiff wird mit den Pfeiltasten gesteuert. Pfeil nach links und Pfeil nach rechts lenken nach links und rechts, Pfeil nach oben beschleunigt, Pfeil nach unten bremst.

1 Wenn du später im Spiel auf das grüne Fähnchen klickst, soll das Raumschiff an die Startposition gesetzt werden, egal wo es sich gerade befindet. Ziehe dazu zwei Blöcke **gehe zu x: ... y: ...** und **setze Richtung auf 90 Grad** ins Skriptfenster. Wenn das Raumschiff an der richtigen Stelle steht, werden die Koordinaten automatisch in den Programmblock übernommen.



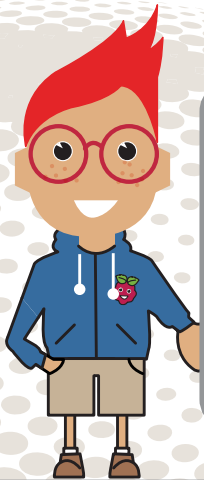
2 Zur Einstellung der Fluggeschwindigkeit verwenden wir eine Variable. Lege auf der Blockpalette **Variablen** mit einem Klick auf **Neue Variable** eine neue Variable mit Namen **Flug** an und hänge anschließend den Block **setze Flug auf 0** an das Programm an. Am Anfang steht das Raumschiff still, hat also eine Fluggeschwindigkeit von **0**.



3 Anschließend folgt eine **wiederhole fortlaufend**-Schleife, in der die Pfeiltasten abgefragt werden. Baue hier zunächst hintereinander zwei **falls ..., dann**-Blöcke ein, die prüfen, ob die Tasten **Pfeil nach links** oder **Pfeil nach rechts** gedrückt wurden. Wurde die Taste **Pfeil nach links** gedrückt, soll sich das Raumschiff um 5 Grad nach links (gegen den Uhrzeigersinn) drehen, bei der Taste **Pfeil nach rechts** um 5 Grad nach rechts (im Uhrzeigersinn).



5 Space Race - oder auf Deutsch: Raumschiffrennen



ZAHLENSCHREIBWEISE

Scratch verwendet wie viele amerikanische Programme den Punkt statt – wie im Deutschen üblich – ein Komma. Deshalb muss hier **0.2** und nicht **0,2** stehen.

4 Beim Drücken der Taste **Pfeil nach oben** soll das Raumschiff beschleunigen. Die Geschwindigkeit, also eigentlich die Länge eines einzelnen Bewegungsschritts, wird in der Variablen **Flug** gespeichert. Bei jedem Tastendruck wird diese Geschwindigkeit um **0.2** erhöht.

5 Wenn das Raumschiff zu schnell fliegt, gerät es leicht außer Kontrolle. Deshalb begrenzen wir die Geschwindigkeit auf **4**. Nur solange die Variable **Flug** kleiner als **4** ist, soll die Geschwindigkeit erhöht werden. Um das zu prüfen, kommt in die **falls ..., dann**-Abfrage mit der Tastaturprüfung eine weitere **falls ..., dann**-Abfrage.

```
falls Taste Pfeil nach oben gedrückt? dann
  falls Flug < 4 dann
    ändere Flug um 0.2
```

6 Auf die gleiche Weise soll die Taste **Pfeil nach unten** das Raumschiff abbremsen. Es soll aber nicht rückwärts fliegen, wenn die Geschwindigkeit immer kleiner wird, sondern auch bei hektischem Drücken der Taste mit einer Geschwindigkeit von **0** zum Stehen kommen. Um eine Schubumkehr wirklich auszuschließen, verwenden wir eine **falls ... sonst**-Abfrage, die das Raumschiff abbremsen, solange die Geschwindigkeit größer als **0** ist, und im theoretisch gar nicht möglichen Fall eine negative Geschwindigkeit sicherheitshalber genau auf **0** setzt.

```
falls Taste Pfeil nach unten gedrückt? dann
  falls Flug > 0 dann
    ändere Flug um -0.2
  sonst
    setze Flug auf 0
```

7 Nachdem alle Tasten in der Endlosschleife abgefragt wurden, soll sich das Raumschiff in jedem Schleifendurchlauf auch noch einen Schritt bewegen. Das gilt ebenfalls, wenn keine der Tasten gedrückt wurde. Die Schrittlänge entspricht der Variablen **Flug** und ist umso größer, je schneller das Raumschiff fliegt.

```
gehe Flug er Schritt
```

8 Das Programm sieht jetzt so aus:

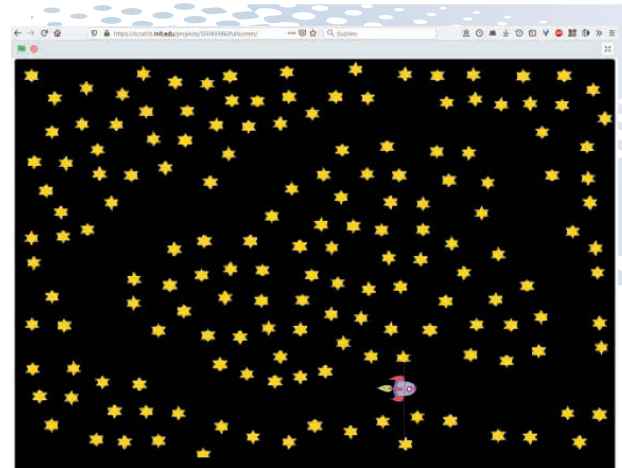


```
Wenn angeklickt wird
  gehe zu x: 68 y: -85
  setze Richtung auf 90 Grad
  setze Flug auf 0
  wiederhole fortlaufend
    falls Taste Pfeil nach links gedrückt? dann
      drehe dich um 5 Grad
    falls Taste Pfeil nach rechts gedrückt? dann
      drehe dich um 5 Grad
    falls Taste Pfeil nach oben gedrückt? dann
      falls Flug < 4 dann
        ändere Flug um 0.2
    falls Taste Pfeil nach unten gedrückt? dann
      falls Flug > 0 dann
        ändere Flug um -0.2
      sonst
        setze Flug auf 0
  gehe Flug er Schritt
```

Sterne fliegen. Das Raumschiff fliegt unbeirrt weiter.



Mit dem Symbol rechts oben kannst du die Scratch-Bühne auf das ganze Browserfenster vergrößern. Beim Spielen brauchst du das Skriptfenster und die Figurenliste schließlich nicht.



Starte das Programm mit einem Klick auf das grüne Fähnchen. Jetzt kannst du das Raumschiff mit den Pfeiltasten fliegen. Dabei kannst du einfach über die



KOLLISIONSERKENNUNG

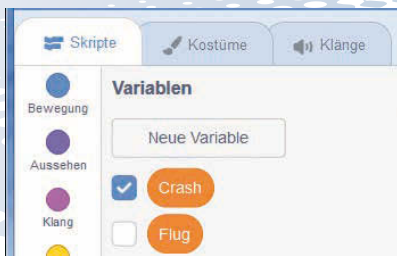
Die Spielregel besagt: Jedes Mal, wenn das Raumschiff an einen Stern stößt, kommt es zum Stillstand und bewegt sich ein kleines Stück zurück, damit du Gelegenheit hast, zu lenken und in die richtige Richtung wieder zu beschleunigen. Außerdem sollen die Crashes gezählt werden. Informatiker und Physiker sprechen von „Kollisionserkennung“, wenn es darum geht, festzustellen, ob sich

5 Space Race - oder auf Deutsch: Raumschiffrennen

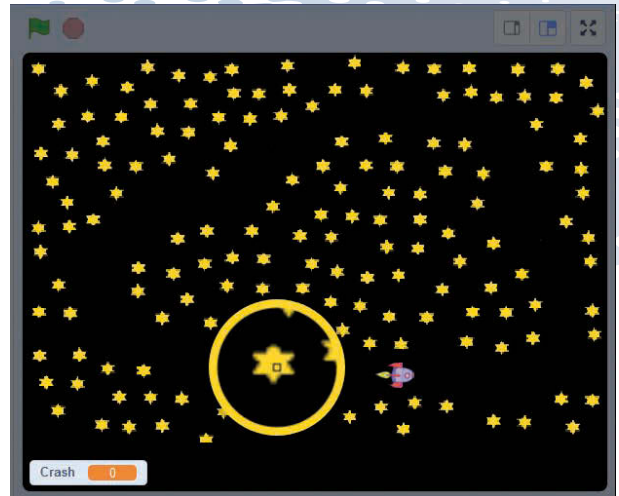
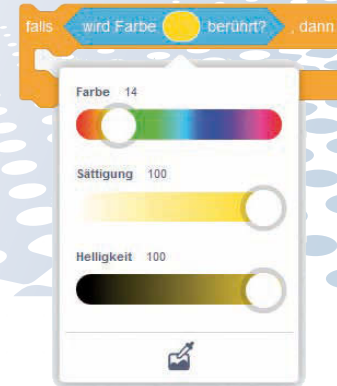


zwei Objekte in der Realität oder virtuell auf dem Bildschirm berühren. Scratch liefert dafür vorgefertigte Blöcke mit, die diese Kollisionserkennung ganz einfach machen.

1 Lege als Erstes auf der Blockpalette **Variablen** eine neue Variable **Crash** an, die die Zusammenstöße mit den Sternen zählt. Achte dabei darauf, dass das kleine Kästchen links neben dem Variablennamen angekreuzt ist. Damit wird diese Variable auf der Bühne angezeigt. Schiebe dieses Anzeigefeld in eine der Ecken, wo es die Flugbahn des Raumschiffs nicht behindert.



2 Baue in die Endlosschleife zusätzlich zu den Tastaturabfragen eine Abfrage ein, die prüft, ob das Raumschiff die Farbe der Sterne berührt. Klicke auf das farbige Feld im Block **wird Farbe ... berührt**. Ein Feld zur Auswahl der Farbe erscheint. Klicke auf die Farbpipette ganz unten in diesem Feld. Klicke dann auf einen der Sterne auf der Bühne, und das Farbfeld nimmt dessen gelbe Farbe an.



- 3 Bei jedem Crash sollen drei Aktionen ablaufen:
- Der **Crash**-Zähler wird um 1 hochgezählt.
- Das Raumschiff wird sofort auf 0 abgebremst
- Das Raumschiff bewegt sich um 10 Koordinateneinheiten rückwärts, um leichter wieder freizukommen.
- 4 Diese drei Aktionen werden durch drei Blöcke innerhalb der **falls ..., dann**-Abfrage erledigt.





5 Am Anfang des Programms muss der **Crash**-Zähler noch auf **0** gesetzt werden, damit du mit einem Klick auf das grüne Fähnchen wieder bei 0 anfangen kannst. Das Programm sieht jetzt so aus:

```
Wenn grünes Fähnchen angeklickt wird
  gehe zu x: 68 y: -85
  setze Richtung auf 90 Grad
  setze Flug auf 0
  setze Crash auf 0
  wiederhole fortlaufend
    falls Taste Pfeil nach links gedrückt? dann
      drehe dich um 5 Grad
    falls Taste Pfeil nach rechts gedrückt? dann
      drehe dich um 5 Grad
    falls Taste Pfeil nach oben gedrückt? dann
      falls Flug > 4 dann
        ändere Flug um 0.2
    falls Taste Pfeil nach unten gedrückt? dann
      falls Flug > 0 dann
        ändere Flug um -0.2
      sonst
        setze Flug auf 0
  gehe Flug er Schritt
  falls wird Farbe gelb berührt? dann
    ändere Crash um 1
    setze Flug auf 0
    gehe -10 er Schritt
```

6 Starte das Programm mit einem Klick auf das grüne Fähnchen. Jetzt kannst du das Raumschiff mit den Pfeiltasten fliegen. Bei jedem Zusammenstoß mit einem Stern wird der **Crash**-Zähler auf der Bühne um 1 hochgezählt.



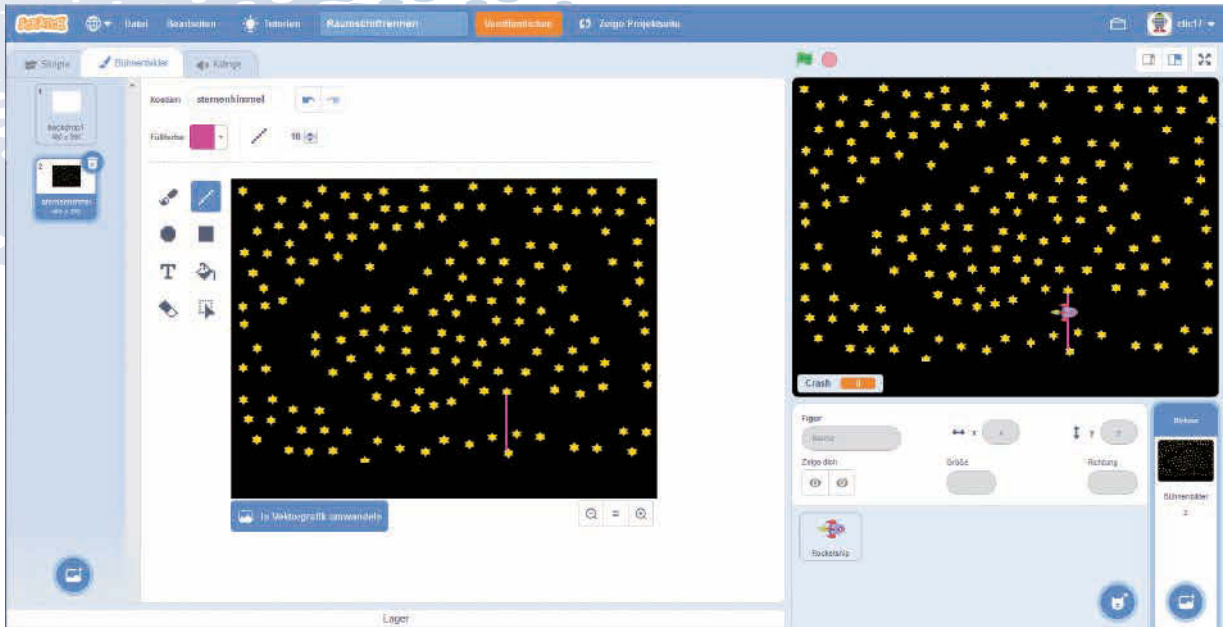
DER RUNDENZÄHLER

Ein Laserstrahl zwischen zwei Sternen dient als Start- und Ziellinie für das Raumschiffrennen. Jedes Mal, wenn das Raumschiff durch diesen Laserstrahl fliegt, wird ein Rundenzähler um 1 erhöht.

1 Klicke in der Figurenliste auf die Bühne und schalte oben auf die Seite **Bühnenbilder**. Male in einer kräftigen Laserfarbe eine gut erkennbare Linie zwischen zwei Sterne an die Stelle, an der das Raumschiff beim Start steht.

2 Lege nun auf der Blockpalette **Variablen** eine weitere Variable namens **Runde** an, die die geflogenen Runden zählt. Auch diese Variable soll auf der Bühne zu sehen sein.

5 Space Race - oder auf Deutsch: Raumschiffrennen



3 Für den Rundenzähler verwenden wir ein eigenes Skript, das bei Klick auf das grüne Fähnchen ebenfalls gestartet wird und als Erstes den Rundenzähler auf **0** setzt.

zählt dann eine Runde. Das Ganze wiederholt sich endlos und zählt jedes Mal eine Runde, wenn das Raumschiff erneut in den Laserstrahl fliegt, nachdem es ihn zwischendurch verlassen hat.

```
Wenn  angeklickt wird
setze Runde auf 0
```

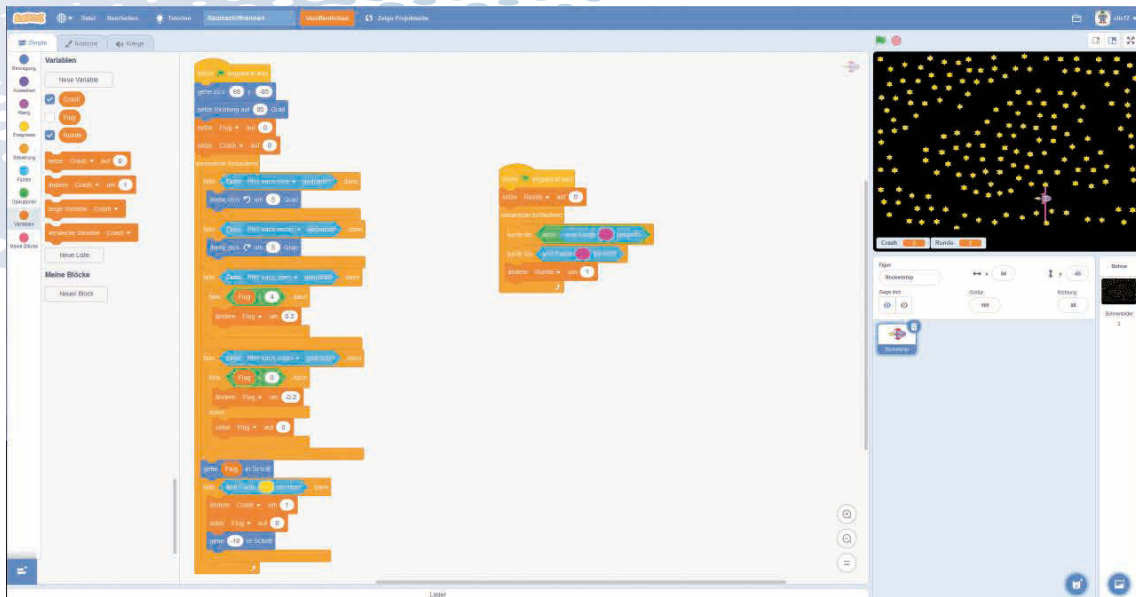
4 Um eine Runde zu zählen, kann nicht einfach nur überprüft werden, ob das Raumschiff den Laserstrahl berührt. Für den Flug durch den Strahl braucht es etwas Zeit. Es würden also bei jedem Durchflug mehrere Runden schnell hintereinander gezählt. Diese Aufgabe lässt sich mit **warte bis ...**-Blöcken elegant lösen. Das Programm wartet am Anfang, bis das Raumschiff über die Startlinie geflogen ist, den Laserstrahl also nicht mehr berührt. Danach wartet es wieder, bis das Raumschiff den Laserstrahl berührt, und

```
Wenn  angeklickt wird
setze Runde auf 0
wiederhole fortlaufend
  warte bis  wird Farbe  berührt?
  warte bis  wird Farbe  berührt?
  andere Runde um 1
```

5 Der Operator **nicht ...** kann zu jeder beliebigen Abfrage feststellen, ob sie **nicht wahr** ist. Er liefert also in dem Moment das Ergebnis **wahr**,



wenn das Raumschiff den Laserstrahl verlassen hat.



Starte das Programm mit einem Klick auf das grüne Fähnchen und fliege möglichst viele Runden mit möglichst wenigen Crashes. Mit der Pfeiltaste nach unten kannst du das Raumschiff jederzeit zum Stehen bringen, um dir eine kurze Verschnaufpause zu gönnen. Der Rundenzähler zählt danach weiter.



Erst wenn du das Programm mit dem Stoppsymbol anhältst und danach mit dem grünen Fähnchen neu startest, beginnen alle Zähler wieder bei **0**.

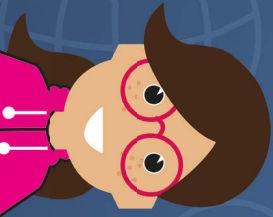


Programmieren

lernen

mit

Scratch



Richtig programmieren zu lernen mag für viele eine staubtrockene Angelegenheit sein, aber nicht mit Scratch! Was Scratch von anderen Programmiersprachen unterscheidet, ist der rein visuelle Ansatz. Das bedeutet, dass du keine ellenlangen Quelltexte mit komplizierter Syntax in deinen Computer tippen musst. Bei Scratch arbeitest du mit kleinen Bildbausteinen, die aneinandergereiht und ineinander verschachtelt werden. So lernst du schon nach kurzer Zeit, wie ein echter Informatiker zu denken.

Damit der Einstieg gelingt, ist von Anfang an alles da. So z. B. die bekannte Scratch-Katze, deren Farbe du verändern und die du unterschiedlichste Befehle ausführen lassen kannst.

Schnell geht es ans Eingemachte: Zeichne einfache Vektorobjekte, entwirf einen Spielwürfel, baue ein schon fast fertiges Raumschiff nach deinen Ideen weiter oder nutze Scratch für spannende Raspberry-Pi-Projekte.

Schon nach kurzer Zeit schreibst du Programme, mit denen du sogar echte Elektronik ansteuern kannst. Lass LEDs leuchten oder baue ein Keyboard, das wirklich Musik macht. Am Ende des Buches erfährst du, wie du Mitglied in der großen Scratch-Community wirst, deine Programme veröffentlichst und mit anderen Scratchern chattest. Und auch wenn es wirklich schwierig wird, steht dir der Autor Christian Immler mit Tipps und Ratschlägen zur Seite.

In diesem Buch geht es um:

- Scratch im Web-Browser nutzen
- Scratch auf dem Desktop offline einsetzen
- Scratch auf dem Raspberry Pi
- Ein neues Scratch-Projekt anlegen
- Das erste Pong-Spiel programmieren
- Figuren miteinander spielen lassen
- Programmtest in der Entwicklungsphase
- Einfache Vektorobjekte zeichnen
- Einen Spielwürfel programmieren
- Fertiges Raumschiff modifizieren
- Bewegungen im Koordinatensystem
- Steuerung für einen Käfer-Roboter
- Scratch-Erweiterungen installieren
- Micro:bit und SenseHAT einsetzen
- Eine Turtle-Grafik erstellen
- Musik machen mit Scratch
- Kultspiel Flappy Bird nachbauen
- Faszinierende Labyrinth und Irrgärten
- Analoguhr mit Sekundenzeiger
- Einfache Spiele mit Farben
- und vieles mehr!



25,00 EUR [D]
ISBN 978-3-645-60672-1

FRANZIS

Besuchen Sie unsere Webseite:
www.franzis.de