



Leseprobe

Mit Spaß am iPad programmieren lernen: Dank der Swift Playgrounds und Phillip Kiefers Anleitungen ist das gar kein Problem! In dieser Leseprobe stellt er die Programmiersprache Swift vor und führt gleich in die ersten Playgrounds ein, in denen das Programmieren zum spielerischen Vergnügen wird. Viel Spaß und guten Lernerfolg!

-  »Vorwort«
»Gestatten: Swift!«
»Ab in die Playgrounds!«
-  Inhaltsverzeichnis
-  Stichwortverzeichnis
-  Der Autor
-  Leseprobe weiterempfehlen

Philip Kiefer

Programmieren lernen mit Swift Playgrounds

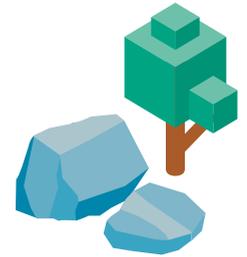
308 Seiten, broschiert, November 2017

24,90 Euro, ISBN 978-3-8421-0308-5

 www.rheinwerk-verlag.de/4429

Vorwort

Was dich in diesem Buch erwartet



Herzlich willkommen in den Swift Playgrounds, einer spannenden neuen Welt für junge Technikfuchse wie dich! Das englische Wort *swift* bedeutet übersetzt flink. Und das englische Wort *playgrounds* bedeutet Spielplätze. Swift ist aber auch der Name einer Programmiersprache, die vom amerikanischen Unternehmen Apple veröffentlicht wurde. Wie der Name andeutet und dieses Buch dir ausführlich zeigen wird, erlernst du die Programmiersprache Swift in den Swift Playgrounds spielerisch und mit jeder Menge Spaß.

Mithilfe einer Programmiersprache werden Programme geschrieben, die dann von verschiedenen Computern ausgeführt werden können. Es gibt viele ganz verschiedene Programmiersprachen. Die Programmiersprache Swift dient vor allem dazu, Programme für Apple-Geräte zu schreiben, beispielsweise für das iPhone, das iPad oder einen »normalen« Computer, den Mac. So ein Programm nennt man übrigens auch App – ein Wort, das du bestimmt schon gehört hast.

Was bringt es dir überhaupt, schon in jungen Jahren zu lernen, wie man programmiert? Nun ja, zum einen bringt es dir, wie schon gesagt, viel Spaß. Zum anderen eröffnet es dir aber auch eine ganz neue Welt – eine Welt des freien Gestaltens, des logischen Denkens, der Erfolgserlebnisse, aber auch des cleveren Umgangs mit Niederlagen, falls ein Programm mal nicht so funktionieren sollte wie gewünscht. Wer als Programmierer richtig gut ist, kann das später sogar mal als sehr gut bezahlten Beruf ausüben. Früh anfangen lohnt sich!

Was du brauchst, um mit diesem Buch Spaß zu haben und eine Menge zu lernen

Dieses Buch führt dich ganz sanft, verständlich und spielerisch an das Programmieren heran. Es basiert auf der App Swift Playgrounds, die von Apple kostenlos zur Verfügung gestellt wird und die du auf einem iPad ab der 4. Generation oder einem iPad mini ab der 2. Generation installieren kannst. Die genaue Vorgehensweise werde ich dir in den späteren Kapiteln Schritt für Schritt beschreiben.

Die App Swift Playgrounds ist keine Spiele-App wie jede andere, aber es ist doch eine Spiele-App. Beim Spielen gilt es, knifflige Aufgaben zu lösen. Und mit jeder gelösten Aufgabe lernst du etwas übers Programmieren, egal, ob du nun eine Spielfigur durch eine Fantasiewelt bewegst oder aber im Garten ganz wirklich eine Drohne in die Luft steigen lässt. In den Swift Playgrounds ist vieles möglich – und du kannst all die unterschiedlichen Möglichkeiten selbst ausprobieren.



Das Programmieren in den Swift Playgrounds ist faszinierend und kinderleicht. Grundvoraussetzung ist lediglich, dass dir ein iPad zum Programmieren zur Verfügung steht. (Quelle der Abbildung: Apple.com)

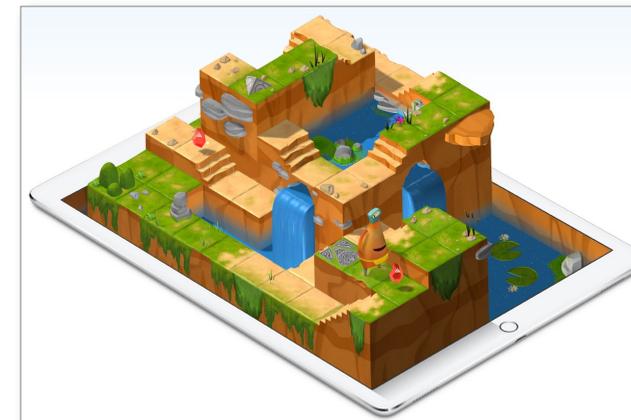
Du musst übrigens kein Computergenie sein, um die Programmiersprache Swift zu erlernen. Es gibt zwar richtige Computergenies wie den Pakistaner Shafay Thobani, der schon im Alter von acht Jahren zum Computer-Star wurde. Aber auch als Normalo kannst du beim Programmieren erfolgreich sein. Wichtig ist nur, dass

du dir die Neugier darauf bewahrst, etwas Neues zu erlernen, und dass du am Ball bleibst. Auch für das Programmieren gilt das alte Sprichwort »Übung macht den Meister«. Dieses Buch bietet dir den perfekten Einstieg dazu.

So ist das Buch aufgebaut

Wir gehen in kleinen Schritten voran, denn niemand soll überfordert werden. Das alles wird in den folgenden Kapiteln ausführlich beschrieben:

- ▶ In Kapitel 1 erhältst du zunächst mal eine Einführung darin, was überhaupt eine App ist und von wem Apps erstellt werden. Außerdem stelle ich dir die Programmiersprache Swift etwas näher vor.
- ▶ Kapitel 2 vermittelt dir allgemeine Grundlagen des Programmierens. Ich gehe genauer darauf ein, was das Wort Programmieren überhaupt bedeutet und welche verschiedenen Programmiersprachen es gibt.
- ▶ In Kapitel 3 geht es ab in die Swift Playgrounds. Ich zeige dir, wie du die App auf einem iPad in Betrieb nimmst und perfekt einrichtest. Du kannst beispielsweise verschiedene Playgrounds auswählen, aber auch die Spielfigur problemlos wechseln, falls dir eine andere mehr zusagt.



In den Swift Playgrounds löst du durch das Programmieren in Swift knifflige Aufgaben, hier wird zum Beispiel die Spielfigur Byte durch Befehle gesteuert. (Quelle der Abbildung: Apple.com)

- ▶ Kapitel 4 widmet sich der Lektion 1 in den Swift Playgrounds. Da geht es schon gleich am Anfang richtig zur Sache! Es geht um Befehle, Funktionen, for-Schleifen, bedingte Anweisungen und noch mehr komische Dinge. Was das alles bedeutet, werde ich dir natürlich genau erklären.
- ▶ In Kapitel 5 wartet die Lektion 2. In dieser Lektion werden weitere wichtige Begriffe des Programmierens eingeführt – und von mir gründlich erklärt: Variablen, Typen, Parameter ... Lass dich bloß nicht von den schwierigen Wörtern abschrecken!
- ▶ Kapitel 6 bietet die Lektion 3. Diese Lektion lehrt dich Möglichkeiten, die Spielwelt nach deinen eigenen Vorstellungen zu gestalten. Natürlich dürfen auch hier die schwierigen Wörter nicht fehlen: Koordinaten, Berührungseignisse, Ereignishandler usw.
- ▶ Nachdem du die drei Lektionen abgeschlossen hast, kannst du dich in den Swift Playgrounds anschließend so richtig austoben. In Kapitel 7 stelle ich dir weitere bereits verfügbare Playgrounds vor. Unter anderem zeige ich dir, wie du mit Swift Playgrounds eine Drohne steuern kannst.



Mit Swift Playgrounds kannst du sogar Drohnen steuern, zum Beispiel die Minidrohne Parrot Mambo. (Quelle der Abbildung: Parrot)

- ▶ Wenn du mit Swift Playgrounds das Programmieren erlernst, möchtest du bestimmt auch eigene Projekte umsetzen. Diesem Thema widmet sich das Kapitel 8. Ich beschreibe darin, wie du die verschiedenen Startpunkte in der App Swift Playgrounds als Basis für eigene Programme verwendest.

- ▶ Die iPad-App Swift Playgrounds ist in erster Linie zum Lernen gedacht. Aber du kannst das Geschriebene auch speichern und versenden. Und Spielzüge in der App kannst du, wenn du magst, fotografieren, filmen und sogar live übertragen. Wie es gemacht wird, zeige ich dir in Kapitel 9.
- ▶ Kapitel 10 widmet sich dem »richtigen« Programmieren. Jetzt wird aus dem Spiel Ernst! Ich erkläre dir, wie aus einem geschriebenen Code eine App entsteht, die auf der ganzen Welt heruntergeladen werden kann. Außerdem stelle ich dir den Mac vor: einen Computer, den du zum richtigen Programmieren unbedingt brauchst. Und wenn du dich für den Beruf eines App-Entwicklers interessierst, findest du hierzu ebenfalls Informationen.
- ▶ Im Anhang erhältst du schließlich noch eine Übersicht über die wichtigsten Befehle und schwierigsten Wörter zum schnellen Nachschlagen.

Wer ich bin und was ich so mache

Bevor wir gleich richtig in das Thema einsteigen, stelle ich mich noch kurz als Autor vor. Ich heiße Philip Kiefer und wurde 1973 in Friedrichshafen am Bodensee geboren. Mein erster Computer war eine vergleichsweise große Kiste namens C64. Darauf habe ich viel gespielt, aber auch schon ein wenig programmiert.

Nach einer längeren Pause, in der ich von Computern gar nichts mehr wissen wollte, bin ich in die Welt der Computer erst wieder eingestiegen, als ich in den 1990er-Jahren in Tübingen Literatur und Philosophie studiert habe. Ich habe in der Zeit auch eine Internetseite für Kinder programmiert. Und diese Seite war dann gewissermaßen der Einstieg in das, was ich ungefähr seit der Jahrtausendwende beruflich mache: schreiben, schreiben, schreiben.

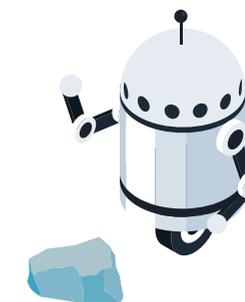
Ich habe schon eine Menge Computerbücher veröffentlicht, aber auch viele Bücher für Kinder und Jugendliche. Als Herr Lipperts von Vierfarben die Idee hatte, ein Programmierbuch für junge Leute herauszubringen, war ich sofort Feuer und Flamme, denn das passt doch perfekt! Das Ergebnis unserer Zusammenarbeit hältst du in deinen Händen und wird dir hoffentlich viel Freude bereiten.

Auf alle Fälle viel Spaß beim Herumtoben in den Swift Playgrounds wünschen dir der Autor Philip Kiefer, der Lektor Erik Lipperts und das gesamte Vierfarben-Team!

PS: An dieser Stelle geht auch noch ein ganz herzliches Extra-Dankeschön an Dominik Hauser. Er ist ein waschechter App-Programmierer und hat mich beim Schreiben dieses Buches die ganze Zeit über fachlich begleitet.

Kapitel 1

Gestatten, Swift!



Du hast doch bestimmt schon mal was von Apps gehört und vielleicht schon selbst die eine oder andere App benutzt. Darauf würde ich meinen Piepvogel aus Metall verwetten, der bei mir draußen im Garten sitzt. Aber nun möchtest du über Apps mehr erfahren als nur, wie du sie benutzen kannst. Du möchtest lernen, wie Apps funktionieren und – Schritt für Schritt – wie du eigene Apps erstellen kannst.

Auf den folgenden Seiten lernst du wichtige Grundlagen kennen. Du erfährst, was genau Apps sind und wer diese erstellt. Du bekommst wichtige Informationen zur Programmiersprache Swift mit auf den Weg und machst dich damit vertraut, wie diese aufgebaut ist. Außerdem stelle ich dir näher dar, was du in diesem Buch lernst – und was nicht.

Was sind Apps, und wer erstellt die?

Du weißt inzwischen schon, dass Apps Computerprogramme sind. Der Begriff App ist schlicht die Abkürzung für das englische Wort *application*, das Anwendung bedeutet – und eine Anwendung meint in diesem Fall nichts anderes als ein Computerprogramm.

»Aber Moment mal!«, wirst du nun bestimmt rufen, »Apps laufen doch auch auf einem iPhone oder einem iPad. Das sind doch keine Computer!« Doch, sind sie. Ein klassischer Computer steht zwar auf einem Schreibtisch und verfügt über einen Bildschirm und eine Tastatur. Aber die Technik hat in den letzten Jahren so große Fortschritte gemacht, dass es auch tragbare Computer im Kleinformat und mit einem berührungsempfindlichen Bildschirm (dem sogenannten *Touchscreen*)

gibt – iPhones und iPads eben sowie auch Smartphones und Tablet-PCs anderer Hersteller.

Bleiben wir beim Beispiel iPad. Wenn du dir ein iPad kaufst, so läuft darauf das Betriebssystem iOS. Ein Betriebssystem kannst du dir so vorstellen wie einen Sportplatz, auf dem du dann deine Sportler – die Apps – einlaufen lässt. Das Betriebssystem sorgt dafür, dass du das Gerät überhaupt einschalten kannst und dass nach dem Einschalten eine grafische Oberfläche auf dem Bildschirm angezeigt wird. Aber es sorgt eben auch dafür, dass du Apps herunterladen und auf dem iPad verfügbar machen – installieren – kannst. Eine ganze Reihe von Apps sind übrigens bereits Bestandteil von iOS, zum Beispiel eine App zum Aufrufen von Webseiten (Safari), eine Mail-App, eine Kalender-App usw.



Diese Abbildung zeigt dir ein iPad und ein iPhone mit unterschiedlichen Apps. (Quelle der Abbildung: Apple.com)

Wer ein iPad kauft (oder geschenkt bekommt), gibt sich aber meistens nicht mit den bereits vorhandenen Apps zufrieden. Deshalb gibt es den *App Store*. *Store* bedeutet zwar auf Englisch Kaufhaus, doch es gibt im App Store keineswegs nur Apps, die Geld kosten. Außer Apps, die du kaufen kannst, findest du im App Store:

- ▶ *völlig kostenlose Apps*, zum Beispiel Apps, die von großen Unternehmen zu Werbezwecken veröffentlicht werden, aber auch nützliche Apps, in denen die

eine oder andere Werbung läuft, damit sich der App-Entwickler den einen oder anderen Euro verdienen kann, obwohl er die App für lau zur Verfügung stellt

- ▶ *Lite-Apps*: Das sind Apps, die es in einer Basisversion kostenlos gibt; wer alle Funktionen der App nutzen möchte, muss für die Vollversion in seine Tasche greifen.
- ▶ *Apps mit In-App-Käufen*: Diese Apps sind zwar ebenfalls kostenlos, aber für bestimmte Funktionen muss man Geld berappen, etwa für besonders flotte Flitzer in einem Autorennspiel.
- ▶ *Apps mit Abo-Angeboten*: Bei diesen Apps wird für die Nutzung eine regelmäßige – meist monatliche oder jährliche – Gebühr verlangt.



Jeder Kauf und In-App-Kauf muss mit der Apple-ID bestätigt werden; hier siehst du als Beispiel einen In-App-Kauf in der App Gardenscapes – das Spiel macht aber auch ganz ohne In-App-Käufe Spaß!

Der App Store ist übrigens auch selbst eine App, die Bestandteil des iPad-Betriebssystems iOS ist. Um eine App aus dem App Store zu laden, brauchst du lediglich ein Benutzerkonto, die *Apple-ID*. Bei Apps, die Geld kosten, musst du zudem eine Zahlungsmethode auswählen. Wie das Installieren einer App genau funktioniert – nämlich speziell der App Swift Playgrounds – zeige ich dir Schritt für Schritt in Kapitel 3.



So werden Apps bezahlt

Für die Bezahlung einer App gibt es mehrere Zahlungsmethoden. Ein Erwachsener kann seine Kreditkarte einsetzen, den bekannten Zahlungsdienstleister PayPal zwischenschalten oder in manchen Fällen das Geld von seiner Handyrechnung abbuchen lassen. Eine weitere gute Zahlungsmethode, die für Erwachsene ebenso wie Kinder und Jugendliche gleichermaßen bestens geeignet ist, sind iTunes-Guthabekarten, die es in vielen Super- oder Elektronikfachmärkten an der Kasse zu kaufen gibt. Man kauft ein bestimmtes Guthaben, etwa 15 Euro, und kann dann nicht mehr ausgeben als dieses Guthaben. Eine sichere Sache!

Angenommen, eine App oder ein In-App-Kauf kostet 1,09 Euro, so heißt das noch lange nicht, dass der App-Entwickler alles bekommt. Einen Teil des Kaufpreises behält Apple ein, und ein weiterer Teil geht an das Finanzamt. Für eine in Deutschland verkaufte App bleiben dem App-Entwickler aber immerhin noch 64 Cent. So geht die Berechnung:

1. Nimm einen Taschenrechner zur Hand, und teile den Kaufpreis durch 1,19. Warum? Weil Apple für jede verkaufte App Umsatzsteuer an das Finanzamt abführen muss. Die Umsatzsteuer beträgt 19 Prozent. Teilst du 1,09 Euro durch 1,19, erhältst du einen Betrag von knapp 92 Cent (auf dem Taschenrechner steht 0,9159663...). Diesen Betrag nennt man auch Nettobetrag, weil du bei diesem Betrag die Steuer abgezogen hast.
2. Nun verlangt Apple für den Verkauf der App eine Provision in Höhe von 30 Prozent vom Nettobetrag. Um festzustellen, was dann noch für den App-Verkäufer übrig bleibt, nimmst du den Betrag auf dem Taschenrechner mit 7 mal und teilst das Ergebnis durch 10. In unserem Beispiel rechnest du also 0,9159663... mal 7 und erhältst 6,4117647... Teile das durch 10, so erhältst du den vorher genannten Betrag von 0,64 Euro, also 64 Cent.

Die App-Entwicklung scheint sich jedenfalls zu lohnen, denn mittlerweile wurden weit über zwei Millionen Apps im App Store veröffentlicht – eine schier unglau-

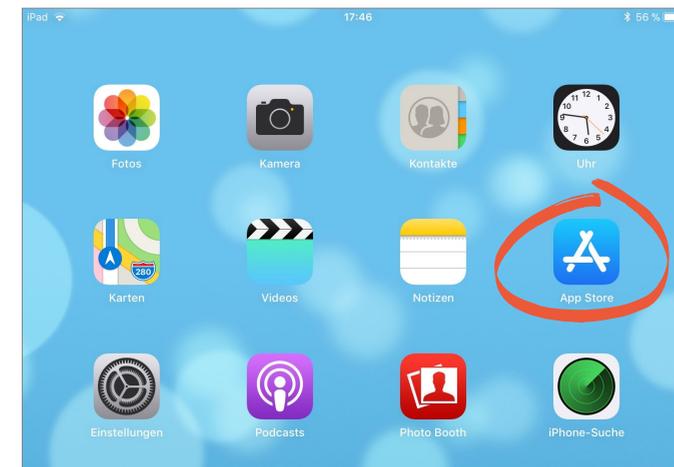
liche Zahl! Aber da sind natürlich nicht nur tolle Apps dabei, etwa geniale Spiele, sondern auch viele Schrott-Apps, die eigentlich kein Mensch braucht.

Und wer entwickelt diese ganzen Apps? Das ist ganz unterschiedlich. Es gibt Firmen, die Apps für andere Firmen entwickeln – sei es eine Werbe-App für ein großes Unternehmen oder die App-Version eines Gesellschaftsspiels im Auftrag eines Spieleverlags. Aber es gibt auch Firmen und Einzelpersonen, die eigene Apps entwickeln und im App Store vermarkten. Ein Paradebeispiel sind Entwickler von Spieleprogrammen, die natürlich schon längst auch den App Store für sich entdeckt haben. Der App Store ist zwar heiß umkämpft, aber mit einer tollen App-Idee kannst du auch heute noch erfolgreich sein.

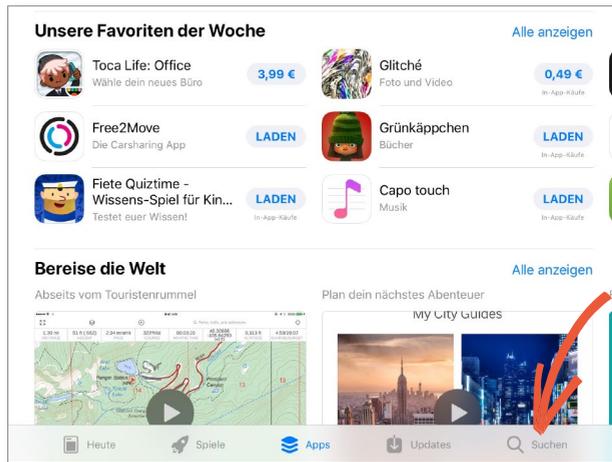
Welche Apps gibt es bereits?

Wenn du dir Gedanken darüber machst, eine App zu programmieren, ist es wichtig zu wissen, welche Apps es bereits gibt. Du möchtest schließlich nicht, dass deine App zwischen Hunderten ähnlichen Apps untergeht. Wenn du eine gute App-Idee hast, prüfst du deshalb als Erstes, ob es Apps dieser Art bereits gibt. Dazu nutzt du den App Store:

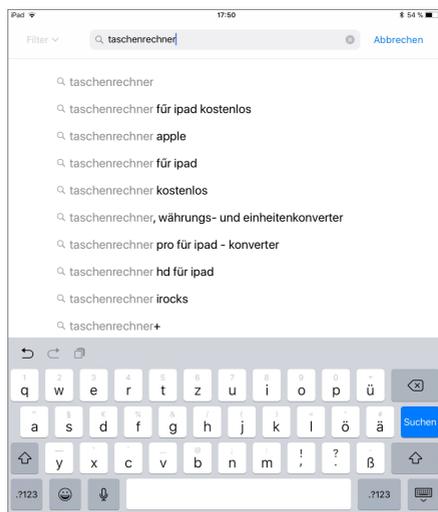
1. Tippe auf deinem iPad auf das blaue App-Symbol mit dem weißen Zirkel, um den App Store aufzurufen.



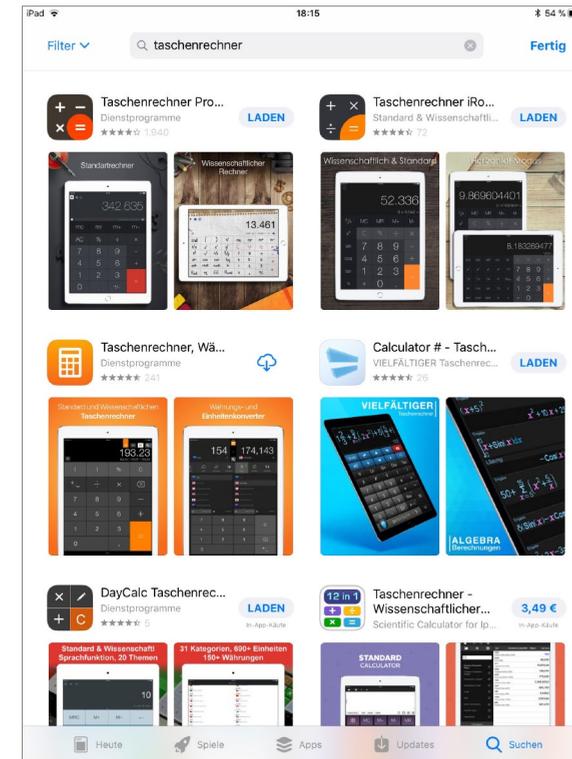
2. Du kannst nun, wenn du Lust hast, in den einzelnen Rubriken stöbern, um dir einen Überblick darüber zu verschaffen, welche Apps es zu den verschiedenen Themen bereits gibt. Um dir die Apps zu einem ganz bestimmten Thema anzeigen zu lassen, tippst rechts unten im App Store auf **Suchen**.



3. Gib deinen Suchbegriff ein, und bestätige mit **Suchen**. Es ist egal, ob du dein Suchwort groß- oder kleinschreibst. Wenn du magst, kannst du auch einen der Vorschläge antippen, der dir während der Eingabe des Suchbegriffs angezeigt wird. Hier wähle ich als Suchbegriff »taschenrechner«.



4. Schon werden dir die passenden Apps angezeigt. Um nähere Informationen zu einer App zu erhalten, tippe diese an.



Du möchtest dir nur die kostenlosen Apps ansehen?

Nachdem du eine App-Suche durchgeführt hast, findest du links oben im App Store ein Filtermenü. Mithilfe dieses Menüs kannst du die Anzeige auf kostenlose Angebote beschränken, aber dir beispielsweise auch nur die Apps einer bestimmten Kategorie ansehen.



Wenn du dir das riesige Angebot im App Store ansiehst, merkst du schnell: Es geht bei der App-Entwicklung keineswegs nur ums Programmieren, sondern auch darum, neue gute Ideen zu haben, um überhaupt die Chance zu bekommen, dass die App im App Store gefunden und heruntergeladen wird. Ohne Programmieren geht allerdings gar nichts!

Warum Swift eine flinke Programmiersprache ist

Als erster waschechter Programmierer der Geschichte gilt eine Frau namens Ada Lovelace. Sie war die Tochter des berühmten englischen Dichters Lord Byron und lebte von 1815 bis 1852. Sie schrieb in der Mitte des 19. Jahrhunderts ein Programm für einen mechanischen Computer, der allerdings nie fertiggestellt wurde.

Seither hat sich natürlich viel getan. In der Mitte des 20. Jahrhunderts baute Konrad Zuse den ersten richtigen Computer zusammen und erdachte für diesen die Programmiersprache *Plankalkül*. Im Lauf der Jahrzehnte entwickelten sich die Computer weiter, sie wurden immer leistungsfähiger. Und für die Computer wurden Hunderte unterschiedliche Programmiersprachen erdacht.

Swift ist also eine von sehr vielen Programmiersprachen, aber sie ist sehr modern und noch gar nicht so alt. Die Entwicklung dieser Programmiersprache begann erst im Jahr 2010. Die Veröffentlichung erfolgte dann im Jahr 2014. Swift wird seitdem regelmäßig aktualisiert: 2015 erschien die Version Swift 2, 2016 die Version Swift 3, 2017 die Version Swift 4 usw.

Die Programmiersprache Swift wurde aber keineswegs aus der Luft gegriffen, sondern sie hat sich andere Programmiersprachen zum Vorbild genommen, zum Beispiel Objective-C, Python und Ruby, um nur mal einige zu nennen. Wundere dich nicht: Programmiersprachen haben oft höchst merkwürdige Namen!

Warum gibt es Swift überhaupt? Nun, das dürfte daran liegen, dass das Unternehmen Apple eine moderne Programmiersprache für seine Produkte haben wollte. Bisher wurde die Programmiersprache Objective-C verwendet, die inzwischen jedoch ganz schön in die Jahre gekommen ist. Mit Swift möchte Apple außerdem den Einstieg in die Programmierung von Apps erleichtern. Für diese Apple-Produkte kannst du in Swift programmieren:

- ▶ *iOS*: iOS ist, wie schon an anderer Stelle erwähnt, das Betriebssystem für iPhones, iPads und den iPod touch.
- ▶ *watchOS*: Dieses Betriebssystem basiert auf iOS und wurde speziell für die Apple Watch – eine Smartwatch von Apple – veröffentlicht.

- ▶ *tvOS*: Auch tvOS basiert auf iOS. Es ist das Betriebssystem des Apple TV – einer kleinen Medienbox, die an den Fernseher angeschlossen werden kann.
- ▶ *macOS*: Auch für macOS, das Betriebssystem der Mac-Computer, kannst du in Swift programmieren, also für iMac, MacBook und Co.



In Swift kannst du Apps für unterschiedliche Apple-Geräte schreiben; die Abbildung zeigt als Beispiel ein MacBook. (Quelle der Abbildung: Apple.com)

Schließlich kann in Swift auch für Computer programmiert werden, auf denen das Betriebssystem *Linux* läuft. Linux stammt allerdings nicht aus dem Hause Apple, sondern wurde von dem Finnen Linus Torvalds erfunden. Das Maskottchen von Linux ist übrigens ein Pinguin namens Tux.

Mit Swift soll App-Entwicklern eine einfache, aber dennoch leistungsstarke Programmiersprache zur Verfügung stehen. Zusammen mit der Programmiersprache wird auch die passende Plattform für App-Entwickler zur Verfügung gestellt. Doch dazu später mehr.

Nun noch zur Frage, warum Swift eigentlich flink ist – denn das sagt ja der englische Name. Zum einen ist Swift durch seinen relativ einfachen Aufbau flink zu erlernen, auch wenn Leute, die noch nie mit dem Programmieren zu tun hatten, dafür natürlich eine gewisse Zeit brauchen. Zum anderen lässt sich in Swift durch die Vereinfachungen gegenüber den Vorgänger-Programmiersprachen flinker programmieren, und Programme lassen sich flinker ausführen.

So ist Swift aufgebaut

Wusstest du schon, dass ein Computer lediglich mit Nullen und Einsen rechnet? Wenn du in Swift oder einer anderen Programmiersprache ein Programm schreibst, so muss dieses für den Computer zum Schluss in Nullen und Einsen »übersetzt« werden, damit er es verstehen kann.

Ein umfangreiches Computerprogramm direkt in Nullen und Einsen zu schreiben – das schafft natürlich kein Mensch. Deshalb gibt es die Programmiersprachen, in denen die unterschiedlichsten Programmierbefehle erteilt werden können, die zusammen den *Quelltext* oder *Quellcode* ergeben, der dann zu einem Programm umgewandelt werden kann.

In Swift wirst du viele Befehle in englischer Sprache sehen. Manchmal verwendest du als Befehl aber auch nur die Abkürzung für ein englisches Wort. Ein Befehl teilt dem Computer mit, welche Aktion er durchführen soll.

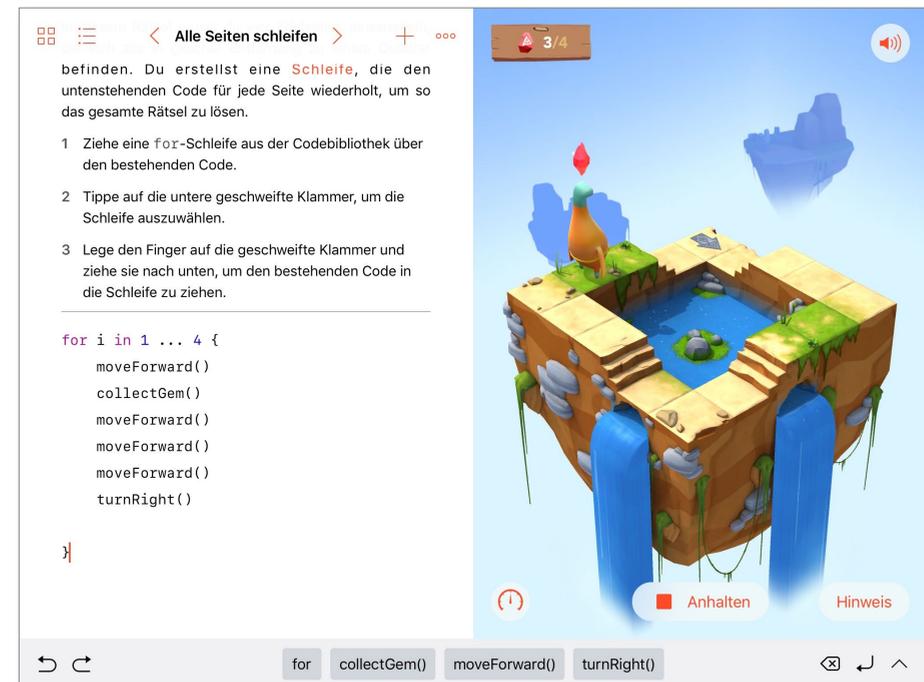
Dazu an dieser Stelle schon mal ein kleines Beispiel: Du möchtest, dass auf dem Bildschirm der Text »Hallo Welt« ausgegeben wird. Dazu gibst du zunächst den Befehl `print` (das englische Wort für drucken) und in Klammern und Anführungszeichen den Text ein, der ausgegeben werden soll. Das ist in diesem Fall der Befehl:

```
print("Hallo Welt")
```

Ins Deutsche übersetzt würde der Befehl in etwa lauten: »Gib ›Hallo Welt‹ auf dem Bildschirm aus!« Du könntest aber auch übersetzen: »Schreib ›Hallo Welt‹ auf den Bildschirm!« Oder: »Zeig ›Hallo Welt‹ auf dem Monitor an!« usw. Swift dient also zum einen dazu, Befehle möglichst einfach darzustellen, aber auch, diese zu vereinheitlichen.

Wenn 1.000 Programmierer auf der ganzen Welt diesen Befehl geben, lautet er genau gleich. Nur der Text, der ausgegeben werden soll, also das, was in Klammern und Anführungszeichen steht, kann sich verändern.

Wenn du Swift erlernst, reicht es aber nicht, die einzelnen Befehle zu kennen. Viel wichtiger ist es, die Logik dahinter zu begreifen. Die einzelnen Lektionen in den Swift Playgrounds werden dir genau hierbei helfen.



Diese Abbildung zeigt dir, wie in Swift Playgrounds Befehle eingeübt werden; in diesem Fall geht es um eine »for«-Schleife, die es ermöglicht, eine Befehlskette mehrmals hintereinander auszuführen.

Noch etwas, das du wissen solltest: Swift ist eine sogenannte *objektorientierte Programmiersprache*. Das bedeutet, grob gesagt, dass zusammengehörige Informationen in Swift zu Objekten zusammengefasst werden. Die Baupläne für solche Objekte bezeichnet man als *Objektypen* oder auch *Klassen*.

Das klingt ganz schön schwierig, nicht wahr? Aber wenn du dir nur mal als Objekt ein Fahrrad vorstellst, dann merkst du, dass ein Fahrrad die verschiedensten Eigenschaften haben kann, zum Beispiel im Hinblick auf die Farbe, das Gewicht oder die Größe. Statt dich beim Programmieren in Swift nur auf eine der Eigenschaften, etwa die Farbe, zu beziehen, möchtest du dich aber auf das ganze Objekt Fahrrad beziehen. Deshalb erstellst du aus den unterschiedlichen Eigenschaften des Fahrrads eine Klasse. Welche Klassen du benötigst, bestimmst du beim Programmieren selbst.

Was du in diesem Buch alles lernst – und was nicht

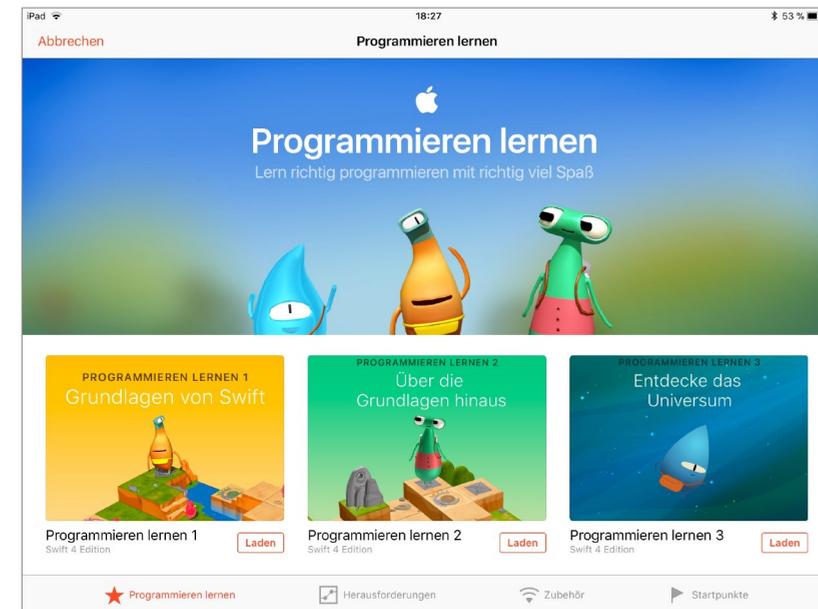
Wenn du eine neue Fremdsprache richtig lernen willst, beispielsweise Englisch, brauchst du dafür viel Zeit, und du musst dich eine ganze Weile dahinterklemmen. Ähnlich ist es auch, wenn du eine Programmiersprache erlernst. Ein einziges Buch macht dich noch nicht zum Sprachgenie, aber du erhältst alle Werkzeuge, um eines zu werden.

Deine Werkzeuge sind in diesem Fall ein iPad, die App Swift Playgrounds und dieses Buch. Die App Swift Playgrounds gibt jeweils die Lektionen vor. Das Buch ist die passende Ergänzung dazu und bietet dir Hilfe dabei, das in der App Gespielte auch wirklich zu verstehen.

Swift Playgrounds ist sozusagen der Spielplatz. Auf diesem Spielplatz kannst du unterschiedliche Spiele spielen. Am wichtigsten sind dabei erst mal die drei Lektionen, mit denen du das Programmieren Schritt für Schritt erlernst, nämlich:

- **Programmieren lernen 1 – Grundlagen von Swift:** In dieser Lektion erlernst du – die Erklärungen zu den einzelnen Themen folgen später – Befehle, Funktionen, `for`-Schleifen, bedingte Anweisungen, logische Operatoren, `while`-Schleifen, Algorithmen.
- **Programmieren lernen 2 – Über die Grundlagen hinaus:** In dieser Lektion machst du dich vertraut mit Variablen, Typen, Initialisierung, Parametern, Weltenbau und Arrays.
- **Programmieren lernen 3 – Entdecke das Universum:** In dieser Lektion geht es um Koordinaten, Berührungseignisse, Ereignishandler und noch einiges mehr.

In jeder Lektion erhältst du unterschiedliche aufeinander aufbauende Aufgaben, die es zu lösen gilt. Eine solche Aufgabe kann beispielsweise darin bestehen, die Spielfigur auf möglichst schnellstem Weg von einem Standort zum nächsten zu bewegen und dabei sämtliche Edelsteine einsammeln zu lassen. Mit jeder gelösten Aufgabe hast du wieder etwas Neues über das Programmieren in Swift gelernt.



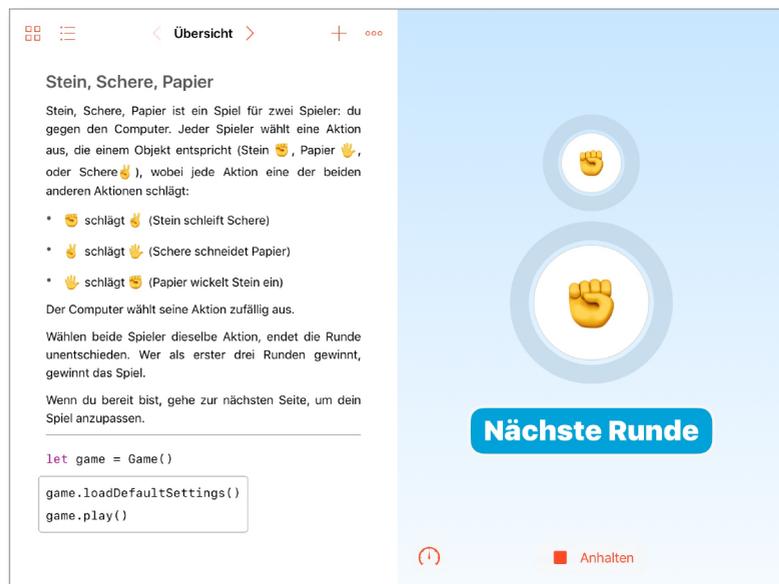
Drei Lektionen bringen dir das Programmieren in Swift auf spielerische und höchst unterhaltsame Weise bei.

Nachdem du dich mit mir zusammen durch die drei Lektionen durchgearbeitet hast, bist du zwar noch kein Programmierer, aber du hast schon eine ganze Menge Kenntnisse gewonnen, die du in verschiedenen Herausforderungen testen kannst. Diese Herausforderungen lassen sich ebenfalls in der App Swift Playgrounds herunterladen. Hier nur einige Beispiele:

- *Labyrinth:* Bei dieser Herausforderung gilt es, durch Programmieren aus kniffligen Labyrinth zu entkommen.
- *Stein, Schere, Papier:* Bei dieser Herausforderung passt du das bekannte Spiel »Stein, Schere, Papier« durch Programmieren ganz deinem eigenen Geschmack an.
- *Blockbrecher:* Auch dieses Spiel lässt sich durch Programmieren anpassen – sogar die Regeln kannst du selbst mitbestimmen. Eine echte Herausforderung!
- *Schiffe versenken:* Bei dieser Herausforderung versuchst du, die Schiffe deines Gegners durch Programmieren aufzuspüren und zu versenken.

- *Töne zeichnen*: Oder hast du Lust auf eine musikalische Herausforderung? Dann erstelle in diesem Playground eigene Musikinstrumente und Soundeffekte.

Wie gesagt, das sind nur einige Beispiele. In Swift Playgrounds findest du noch weitere Herausforderungen. Damit die Herausforderungen auch wirklich Herausforderungen sind, werde ich dir beim Lösen der Aufgaben nicht mehr großartig helfen. Natürlich gibt es – in Kapitel 7 – einige Tipps dazu, aber beim Programmieren selbst bist du hier schon auf dich allein gestellt.



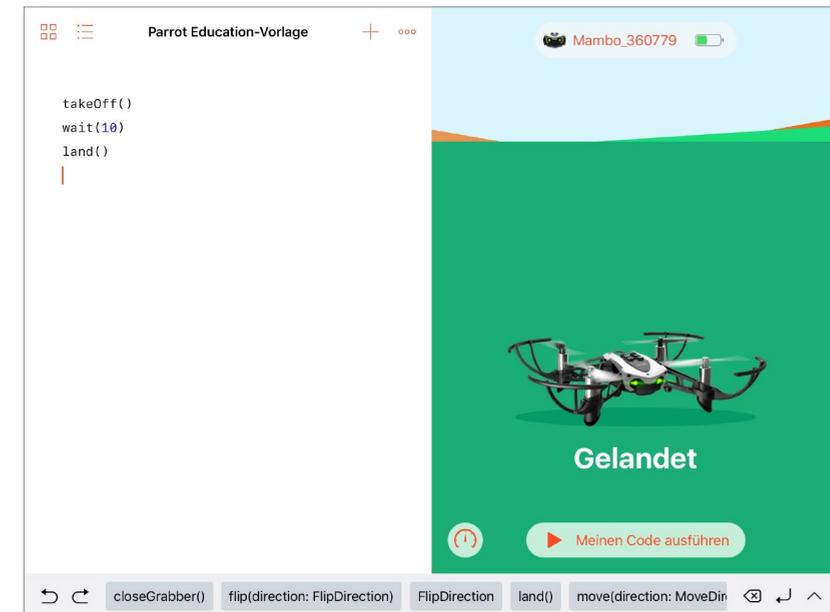
Eine der Herausforderungen in der App Swift Playgrounds: das Spiel »Stein, Schere, Papier«

Die App Swift Playgrounds unterstützt darüber hinaus eine ganze Reihe von Geräten, die sich durch Programme, die du selbst geschrieben hast, steuern lassen. Wie das gemacht wird, zeige ich dir – ebenfalls in Kapitel 7 – am Beispiel der Minidrohne *Parrot Mambo*. Auch in diesem Fall lernst du spielerisch, durch das Programmieren eigene Ideen umzusetzen.

Beim Programmieren der Geräte geht es darum, diese automatisch zu steuern. Eine Drohne kannst du beispielsweise automatisch starten lassen, in eine andere

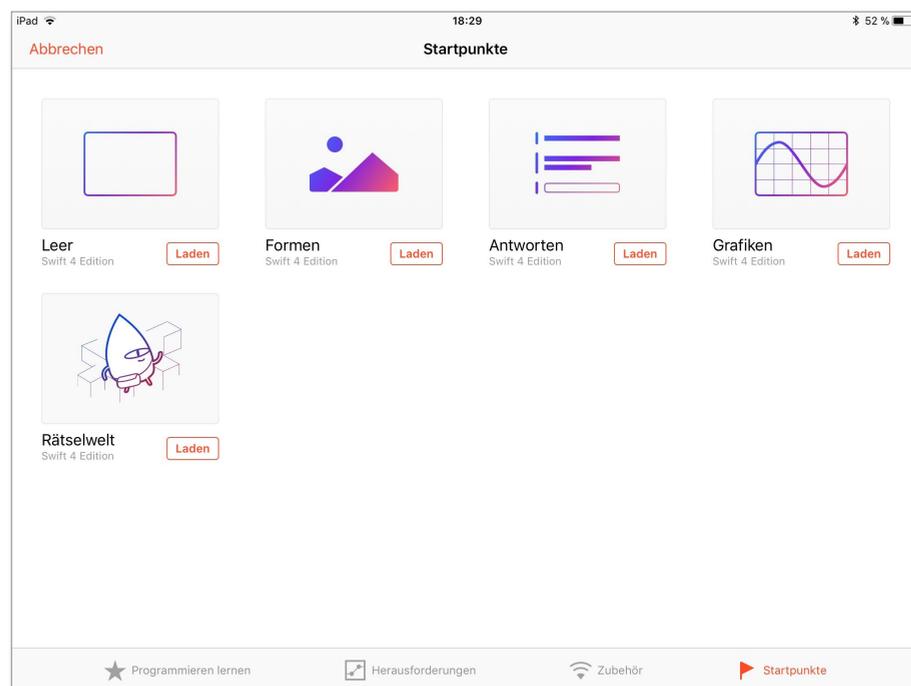
Richtung bewegen, und auch die Landung kann nach einer von dir festgelegten Zeit automatisch erfolgen.

Ich kann dir sehr empfehlen, dir ein entsprechendes Zubehör zuzulegen, denn du siehst deine Programmiererfolge in diesem Fall nicht nur auf dem iPad, sondern ganz echt in deinem Zimmer oder draußen im Garten. Wünsch dir doch eine mit Swift Playgrounds programmierbare Drohne oder einen Roboter zum Geburtstag oder einem anderen Anlass!



Hier programmiere ich eine Drohne so, dass sie vom Boden abhebt, zehn Sekunden in der Luft schwebt und dann wieder landet.

Schließlich kannst du mit Swift Playgrounds auch ganz eigene Programmierprojekte umsetzen. Dafür werden dir in der App verschiedene »Startpunkte« angeboten, je nachdem, um was für eine Art Projekt es sich handelt. Du kannst dein Projekt aber auch einfach in einer leeren Vorlage erstellen. Wichtig zu wissen: Die in der Swift Playgrounds erstellten Projekte lassen sich zwar aus der App heraus versenden, aber sind zunächst mal nur für die Verwendung in der App selbst gedacht – du programmierst mit Swift Playgrounds also noch keine Apps, die später mal im App Store gekauft werden können.



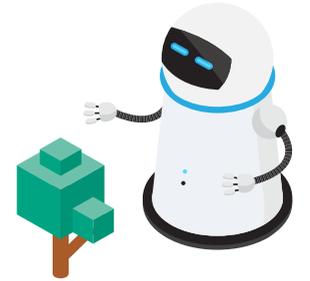
Die App Swift Playgrounds bietet mehrere Startpunkte für deine eigenen Programmierprojekte.

In Kapitel 8 werde ich dir zeigen, wie du auf der Basis verschiedener Startpunkte an eigene Programmierprojekte herangehst. Wenn du eine richtige App programmieren willst, reicht das iPad mit der App Swift Playgrounds aber, wie gesagt, nicht mehr aus, dann brauchst du einen Mac. Zu diesem Thema findest du alle wichtigen Informationen in Kapitel 10.

Kurz und knapp: Du kannst dir mithilfe der App Swift Playgrounds und diesem Buch das Wissen erarbeiten, das du zum Programmieren in der Programmiersprache Swift brauchst. Ob du dann später lediglich einfache Programme schreibst oder richtig tolle Apps programmierst, das liegt allein an dir!

Kapitel 3

Ab in die Playgrounds! – Swift spielerisch erlernen



In den vorigen Kapiteln hast du schon eine ganze Menge über das Programmieren gelernt. Aber jetzt kannst du endlich selbst etwas tun. Um die folgenden Anleitungen nachvollziehen zu können, brauchst du auf alle Fälle ein iPad sowie eine Internetverbindung. Ich gehe außerdem davon aus, dass du die wichtigsten Bedienfunktionen des iPads bereits kennst.

So erstellst du deine eigene Apple-ID

Um auf dem iPad eine App – in diesem Fall Swift Playgrounds – installieren zu können, benötigst du eine *Apple-ID*. Die Apple-ID ist ein Benutzerkonto, das du übers Internet beim Unternehmen Apple anlegst. Die Apple-ID wirst du später beispielsweise auch benötigen, wenn du bei Apple ein Entwickler-Konto eröffnen möchtest, um deine eigenen Apps veröffentlichen zu können.

Vielleicht verwendet das iPad die ganze Familie, und dein Vater oder deine Mutter hat sich bereits mit einer Apple-ID angemeldet? Dann kannst du die App Swift Playgrounds mit dieser Apple-ID herunterladen.

Genauso ist es auch möglich, eine eigene Apple-ID anzulegen und diese zum Herunterladen zu verwenden. Nun stellt sich das erste Problem: Du musst mindestens 13 Jahre alt sein, um die Apple-ID selbst zu erstellen, ansonsten muss ein Elternteil die Apple-ID für dich anlegen, oder du verwendest eben einfach die Apple-ID des Elternteils mit.

Es gibt verschiedene Möglichkeiten, eine neue Apple-ID anzulegen. Um eine Apple-ID in einem *Webbrowser* (einem Programm zum Aufrufen von Webseiten) zu erstellen, rufst du diese Webadresse auf: <https://appleid.apple.com>. Falls du dich mit dem Aufrufen von Webseiten noch nicht so gut auskennen solltest, lass dir dabei von einem Erwachsenen helfen. Klicke auf der so geöffneten Webseite auf den Link **Deine Apple-ID erstellen**.



Hier wird eine neue Apple-ID in einem Webbrowser angelegt; das Erstellen der Apple-ID kann aber beispielsweise auch im Programm iTunes oder in den iPad-Einstellungen erfolgen.

Zum Erstellen der Apple-ID musst du einige Angaben machen, unter anderem benötigst du eine E-Mail-Adresse. Wähle außerdem ein sicheres Passwort, das aus Groß- und Kleinbuchstaben, Ziffern und Sonderzeichen besteht, und beantworte drei Sicherheitsfragen. Ich empfehle dir, deine Zugangsdaten zu notieren und in einem Geheimversteck aufzubewahren für den Fall, dass du die Daten mal vergessen solltest.

Tipp!

Hilfe! Ich habe noch keine E-Mail-Adresse

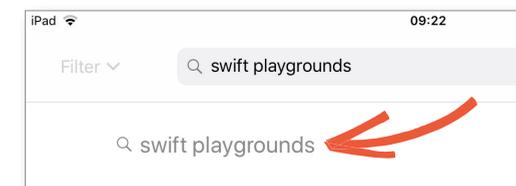
Falls du noch keine E-Mail-Adresse hast, kannst du beispielsweise unter <https://mail.google.com> kostenlos ein Google-Konto mit E-Mail-Adresse anlegen. Auch dazu musst du mindestens 13 Jahre alt sein.

Zur Anmeldung auf einem Apple-Gerät beziehungsweise bei einem Apple-Dienst benötigst du zukünftig die mit der Apple-ID verknüpfte E-Mail-Adresse sowie das Passwort. Die Sicherheitsfragen brauchst du nur im Notfall, etwa um ein vergessenes Passwort wiederherzustellen.

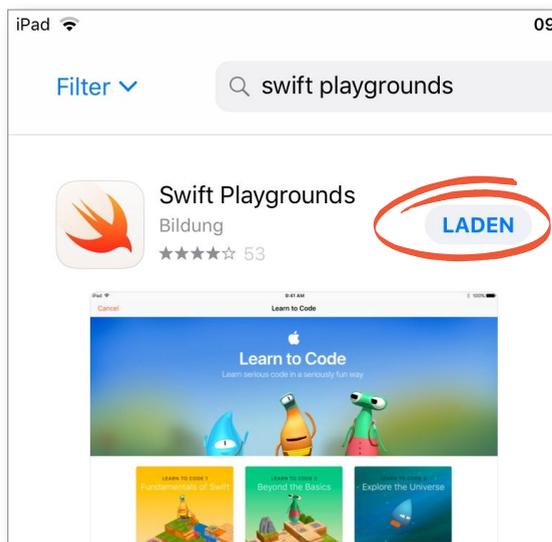
Swift Playgrounds auf dem iPad installieren

Nachdem das mit der Apple-ID geklärt ist, installierst du die App Swift Playgrounds auf deinem iPad. Die App wird dazu aus dem auf dem iPad bereits vorhandenen App Store übers Internet auf das iPad geladen. Ich zeige dir den ganzen Vorgang Schritt für Schritt:

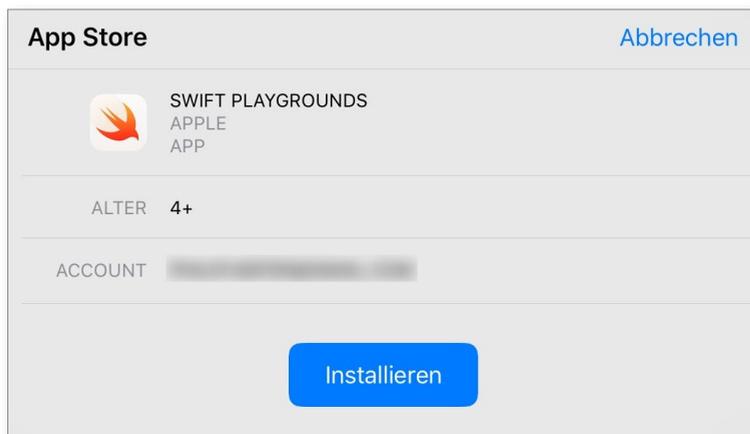
1. Tippe auf dem Home-Bildschirm auf das App-Symbol des App Stores. Du hast ja bereits in Kapitel 1 erste Bekanntschaft mit dem App Store geschlossen.
2. Tippe nun rechts unten auf *Suchen*.
3. Gib als Suchbegriff »swift playgrounds« ein.
4. Tippe den gefundenen Eintrag an.



5. Nun werden die passenden Suchergebnisse aufgelistet. Tippe bei der App Swift Playgrounds auf **Laden**.

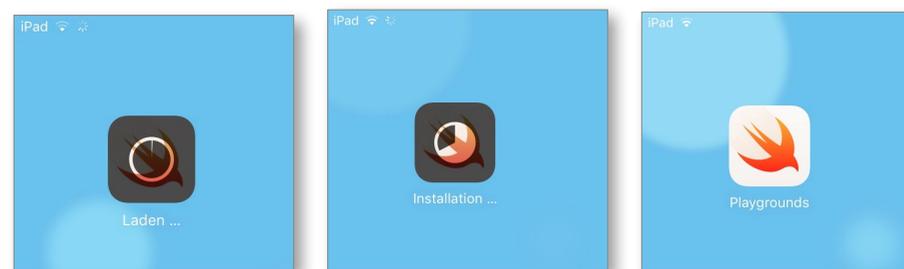


6. Tippe als Nächstes auf **Installieren**.



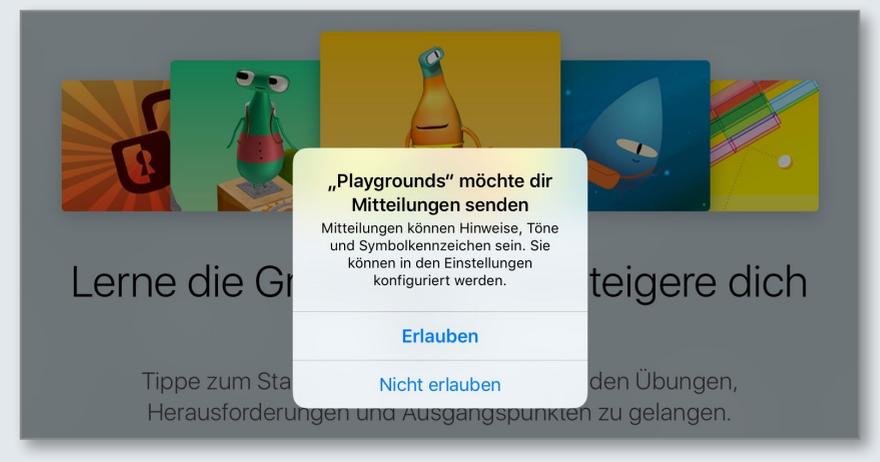
7. Nun wirst du gegebenenfalls zur Anmeldung mit deiner Apple-ID aufgefordert. Falls du noch nicht angemeldet bist, meldest du dich mit deiner E-Mail-Adresse und dem Passwort an.

8. Die App wird nun auf das iPad heruntergeladen und automatisch installiert. Anschließend kannst du das App-Symbol antippen, um die App aufzurufen.



Mitteilungen der App empfangen

Beim ersten Start der App wirst du gefragt, ob du erlauben möchtest, dass die App Mitteilungen auf dem Home-Bildschirm sowie auf dem Sperrbildschirm ausgibt. Da spricht nichts dagegen, deshalb tippe ruhig auf **Erlauben**. Das Mitteilungsverhalten der einzelnen Apps kannst du jederzeit nachträglich in den iPad-Einstellungen unter **Mitteilungen** ändern.

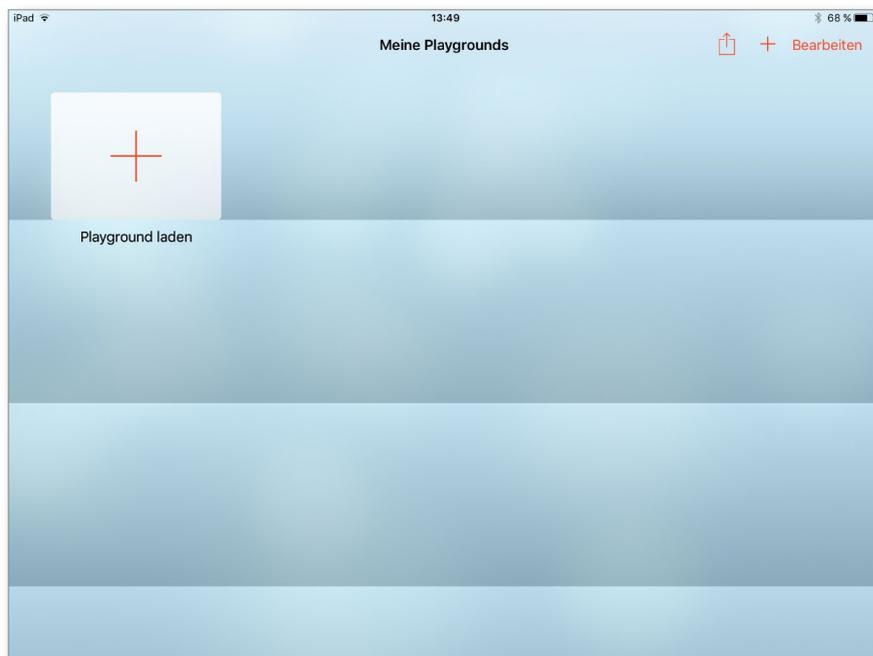


Du kannst die App Swift Playgrounds – so wie viele andere Apps auch – entweder im Hoch- oder im Querformat verwenden, indem du das iPad entsprechend ausrichtest. In den meisten Fällen ist das Querformat praktischer, aber probiere ruhig selbst aus, welche Ausrichtung des iPads dir besser liegt.

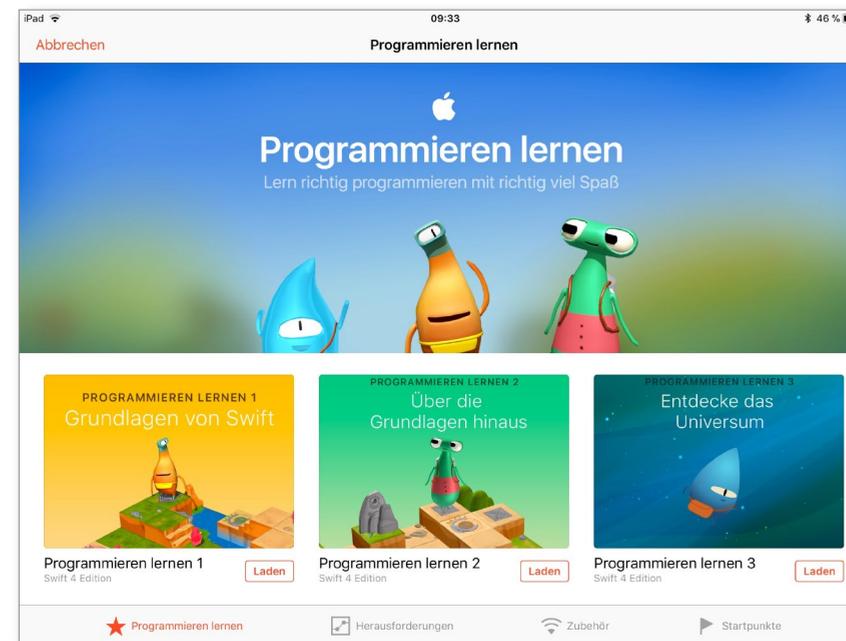
Playgrounds herunterladen

Mit der App Swift Playgrounds steht dir nun der Spielplatz zur Verfügung. Diesen füllst du mit den Spielen, die dich interessieren beziehungsweise die in diesem Buch besprochen werden. Die App besteht aus zwei verschiedenen Ebenen:

- ▶ einer Bibliothek mit deinen Playgrounds. Die Playgrounds werden darin in einer Art Bücherregal sortiert und lassen sich dort – wie ich es dir noch zeigen werde – auch wieder entfernen. Die Abbildung zeigt dir eine noch leere Bibliothek.



- ▶ einer Plattform, von der du die gewünschten Playgrounds in die App herunterladen kannst. Um von der Bibliothek zu dieser Plattform zu gelangen, tippst du in der Bibliothek auf das große Pluszeichen; wenn du von der Plattform zur Bibliothek zurückgelangen möchtest, ohne einen Playground zu laden, tippst du links oben auf **Abbrechen**.



Bei genauerem Hinsehen stellst du fest, dass die Plattform zum Herunterladen der Playgrounds noch mal in vier Rubriken unterteilt ist, die ich in Kapitel 1 schon mal angesprochen habe. Tippe unten in der App auf eine Rubrik, um die entsprechenden Playgrounds laden zu können. Die Rubriken heißen:

- ▶ **Programmieren lernen:** Hier findest du die drei Lektionen zum Erlernen der Programmiersprache Swift. Diese Lektionen werden wir in Kapitel 4 bis Kapitel 6 sehr ausführlich besprechen.
- ▶ **Herausforderungen:** Diese Rubrik bietet die unterschiedlichsten Spiele, mit denen du das Gelernte vertiefen kannst, indem du die Spiele durch Programmieren deinen Vorstellungen anpasst.
- ▶ **Zubehör:** Mit Swift Playgrounds lassen sich, wie bereits an anderer Stelle erwähnt, verschiedene Roboter und Drohnen steuern. Die entsprechenden Playgrounds findest du in dieser Rubrik.
- ▶ **Startpunkte:** Diese Rubrik schließlich bietet dir Startpunkte für deine eigenen Projekte, zum Beispiel wenn du ein Quiz programmieren, bestimmte Formen darstellen oder auch ein leeres Projekt anlegen möchtest.



Das sieht so anders aus!

An dieser Stelle noch der wichtige Hinweis, dass die App Swift Playgrounds ständig weiterentwickelt wird. Es kann sein, dass sich an der einen oder anderen Stelle etwas geändert hat, wenn du dieses Buch liest. Aber lass dich davon bitte nicht irritieren!

Das Herunterladen eines Playgrounds erfolgt ganz ähnlich wie das Herunterladen einer App aus dem App Store. Dazu eine kleine Anleitung:

1. Öffne die App **Swift Playgrounds**.
2. Wird die Bibliothek angezeigt, tippe auf das große Pluszeichen.



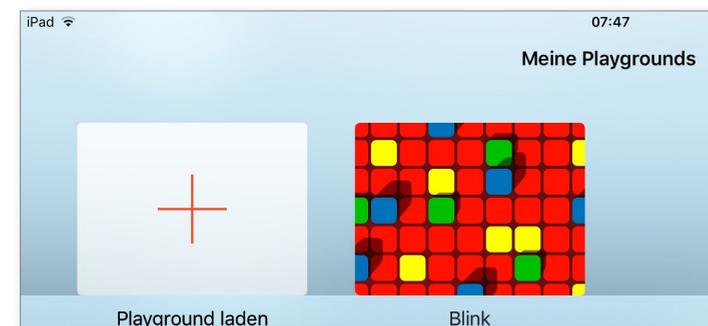
3. Dir werden nun die Playgrounds in der Rubrik **Programmieren lernen** angezeigt. Falls du einen Playground aus einer anderen Rubrik laden möchtest, wähle diese unten in der App aus (also **Herausforderungen**, **Zubehör** oder **Startpunkte**).



4. Bei einem Playground, den du in die App herunterladen möchtest, tippst du auf **Laden**.



5. Dir wird anschließend automatisch die Bibliothek angezeigt – der Playground steht dir dort nach dem Herunterladen zur Verfügung.



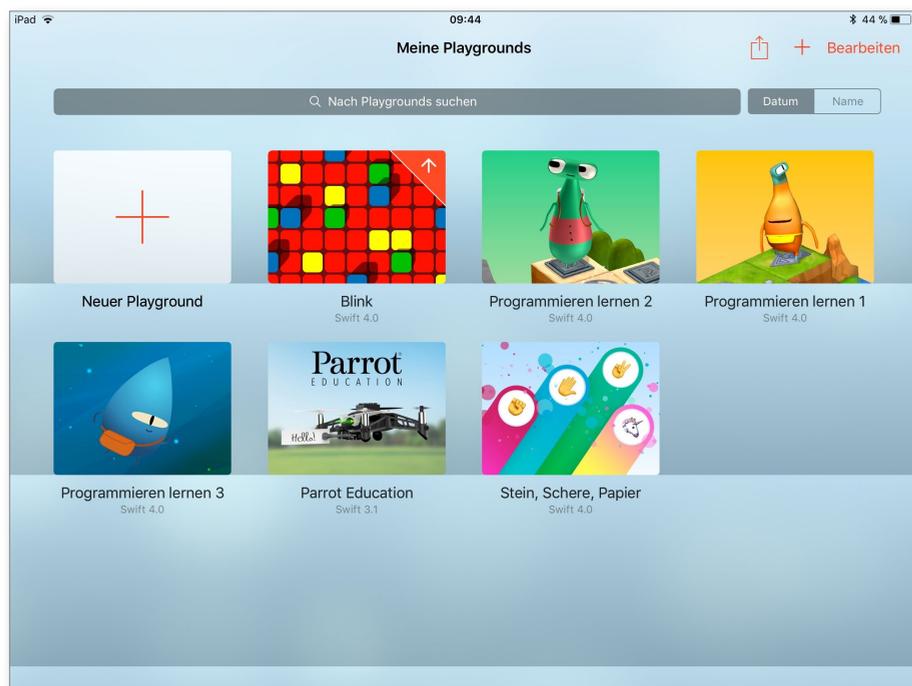
Die Playgrounds laufen dir nicht davon! Lade deshalb immer nur die Playgrounds in die App, die du verwenden oder ausprobieren möchtest. Du würdest damit nicht nur unnötigerweise Speicherplatz auf dem iPad vergeuden, sondern die Bibliothek könnte bei zu vielen heruntergeladenen Playgrounds auch ganz schön unübersichtlich werden.

Playgrounds verwalten

Die Playgrounds werden in der Bibliothek zunächst mal in der Reihenfolge angezeigt, in der du sie heruntergeladen hast. Der zuletzt heruntergeladene Playground steht am Anfang, der als Erstes heruntergeladene Playground am Ende. Wenn du später einen Playground bearbeitest, wirst du feststellen, dass der zuletzt bearbeitete Playground an den Anfang rückt.

Hast du viele Playgrounds in der Bibliothek und findest einen bestimmten Playground nicht auf Anhieb, kannst du die eingebaute Suchfunktion nutzen: Tippe

dazu mit dem Finger auf die Bibliothek, und ziehe den Finger nach unten – oberhalb der Playgrounds wird auf diese Weise ein Suchfeld eingeblendet. Tippe in das Suchfeld, und gib den Namen des Playgrounds ein, den du suchst.



Mithilfe des eingebauten Suchfelds lässt sich ein bestimmter Playground in der Bibliothek schnell aufspüren.

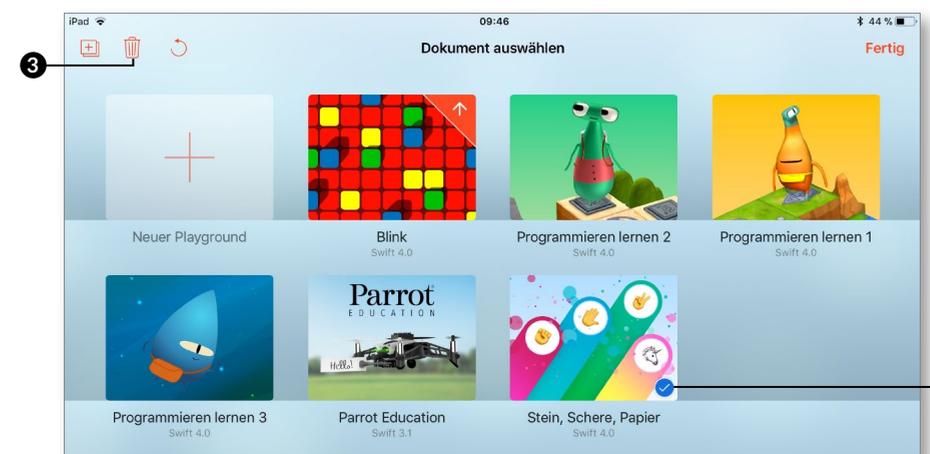
Rechts neben dem Suchfeld findest du außerdem zwei Schalter, um die Sortierung der Playgrounds in der Bibliothek anzupassen. Normalerweise ist hier **Datum** eingestellt, damit immer die aktuellsten Playgrounds am Anfang angezeigt werden. Du kannst aber auch den Schalter **Name** wählen, dann werden die Playgrounds in der Bibliothek nach dem ABC sortiert.

Natürlich lassen sich nicht mehr benötigte Playgrounds auch wieder aus der App löschen. Die folgende Anleitung zeigt dir, wie du dazu vorgehst:

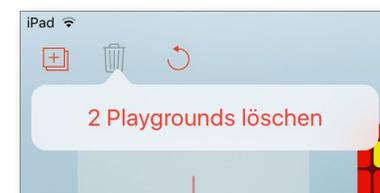
1. Tippe rechts oben in der Bibliothek auf **Bearbeiten** ❶.



2. Tippe einen oder mehrere Playgrounds an, die du löschen möchtest ❷. Die ausgewählten Playgrounds werden mit Häkchen versehen. Willst du die Auswahl eines Playgrounds rückgängig machen, tippst du diesen einfach erneut an.
3. Zum Löschen der ausgewählten Playgrounds tippst du nun links oben in der App auf das Papierkorbsymbol ❸.

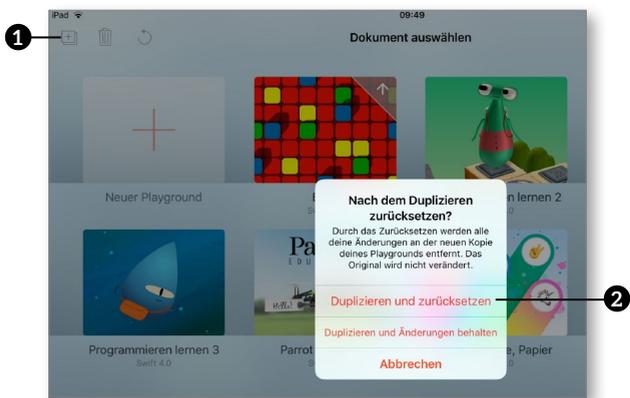


4. Bestätige das Löschen der Playgrounds in der Sprechblase, die aufpoppt.

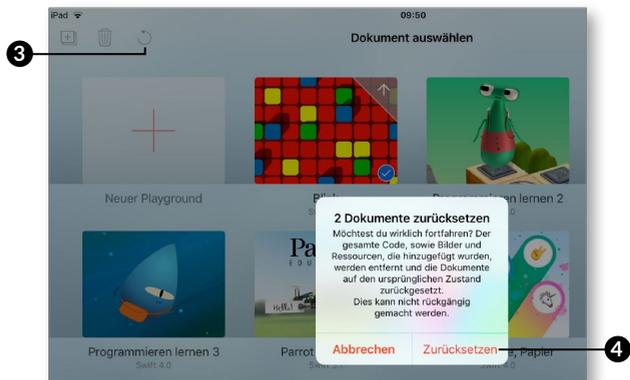


Für die ausgewählten Playgrounds bieten sich noch zwei weitere Funktionen, die du kennen solltest:

- ▶ **Duplizieren:** Tippe auf das Plus-Symbol links neben dem Papierkorbsymbol **1**, um die ausgewählten Playgrounds zu duplizieren, also diese zu verdoppeln. Wähle anschließend aus, ob die Duplikate die Änderungen des Original-Playgrounds beinhalten sollen (**Duplizieren und Änderungen behalten**) oder diese im ursprünglichen Zustand dupliziert werden sollen (**Duplizieren und zurücksetzen** **2**) – also so, wie du sie aus dem Internet heruntergeladen hast. Der Original-Playground wird in beiden Fällen nicht verändert.



- ▶ **Zurücksetzen:** Um die ausgewählten Playgrounds zurückzusetzen, sie also wieder in den Zustand zu bringen, in dem du sie aus dem Internet geladen hast, tippst du auf das runde Pfeilsymbol rechts neben dem Papierkorbsymbol **3** und bestätigst mit **Zurücksetzen** **4**.



Playgrounds teilen

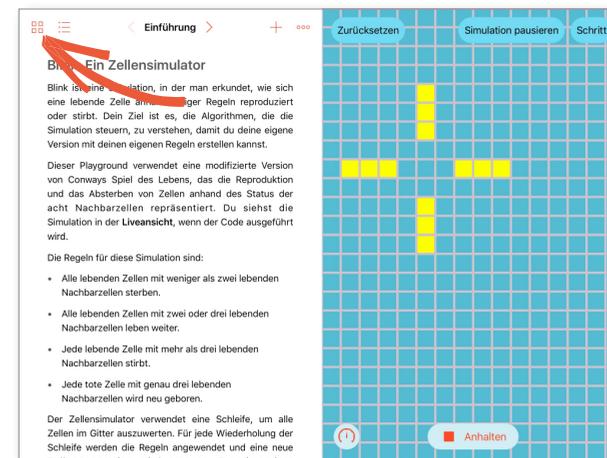
Möchtest du einen Playground mit deinen Freunden teilen? Auch das ist kein Problem: Tippe dazu rechts oben in der Bibliothek auf das Symbol, das ein Quadrat mit einem nach oben weisenden Pfeil zeigt. Anschließend tippst du den Playground an, den du teilen möchtest, und triffst im Menü, das sich öffnet, deine Auswahl. Übrigens kannst du Playgrounds auch auf deinem eigenen iCloud Drive speichern und dann über das Plus-Symbol rechts oben in der App darauf zugreifen.



Playground öffnen und Bedienoberfläche kennenlernen

Nun kennst du die Funktionen zum Herunterladen und Verwalten deiner Playgrounds. Es wird Zeit, einen Playground zu öffnen! Dazu tippst du einfach den Playground, den du öffnen möchtest, in der Bibliothek an.

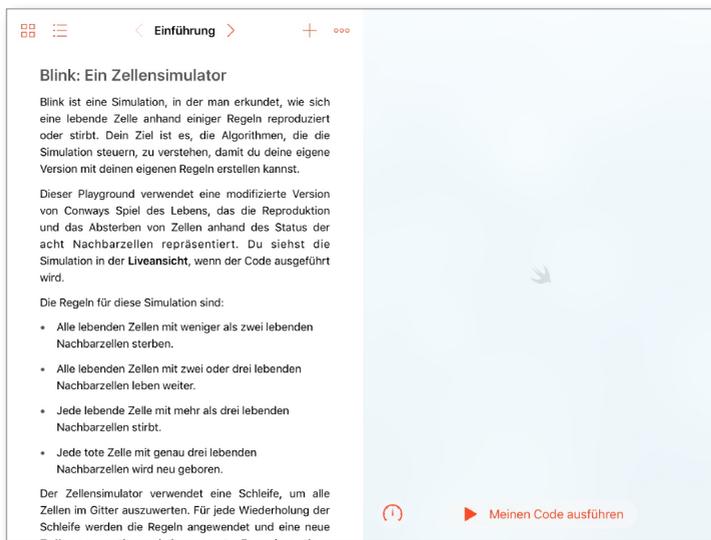
Um einen Playground wieder zu schließen und zurück zur Bibliothek zu gelangen, tippst du in einem Playground links oben auf das Symbol, das vier kleine Quadrate zeigt – das Bibliothekssymbol.



Hier wurde der Playground »Blink« geöffnet; durch Antippen des Bibliothekssymbols links oben im Playground gelangst du zurück zur Bibliothek.

Bevor du den Playground wieder schließt, stelle ich dir die Bedienoberfläche kurz vor, die bei allen Playgrounds zumindest ähnlich ist. Wenn du die Darstellungen hierzu auf deinem eigenen iPad genau nachvollziehen möchtest, lade dir ebenfalls den Playground *Blink* in die App und öffne ihn.

Den größten Teil der Bedienoberfläche nehmen der *Textbereich* und der *Anzeigebereich* ein. Hältst du dein iPad im Hochformat, so wird unten der Textbereich und oben der Anzeigebereich dargestellt; hältst du dein iPad hingegen im Querformat, erscheint der Textbereich links, und der Anzeigebereich erscheint rechts.



Hier halte ich das iPad im Querformat, deshalb wird der Textbereich links und der Anzeigebereich rechts dargestellt; da noch nichts programmiert wurde, ist im Anzeigebereich allerdings noch nichts beziehungsweise nur wenig zu sehen.

Bleiben wir noch kurz beim Anzeigebereich. Dort siehst du unten einen Abspielknopf (wie bei einem MP3-Player) und den Text **Meinen Code ausführen**. Da tippst du später drauf, wenn du etwas programmiert hast und das Ergebnis im Anzeigebereich betrachten möchtest. Tippst du auf das Uhrensymbol links daneben, erhältst du weitere Möglichkeiten zum Ausführen des Codes, nämlich **Schrittweise ausführen** und **Langsame Schritte** – das Ausführen des Codes wird in diesem Fall schrittweise vollzogen, sodass du die einzelnen Schritte im Textbereich genau nachvollziehen kannst.

Je nach Playground können im Anzeigebereich noch weitere Knöpfe auftauchen, beispielsweise um Einstellungen zum Sound vorzunehmen oder um einen Hinweis zur Lösung eines Programmierproblems zu erhalten.

Doch nun gehen wir erst mal in den Textbereich. Dort wird einerseits Text zum Lesen angezeigt, etwa die Einführung in den Playground oder in eine Seite, aber auch das Programmieren erfolgt im Textbereich.



Diese Abbildung zeigt die Seite »Neue Variationen« in der App »Blink«; der Quelltext ist bereits vorhanden und kann von dir beliebig angepasst werden.

Um in einem Playground eine Seite aufzurufen, hast du zwei Möglichkeiten: Blättere entweder in einem Playground vor oder zurück, indem du neben der Seitenüberschrift oben im Playground auf eines der Pfeilsymbole tippst; oder aber tippe auf das Listensymbol, das du ebenfalls oben im Playground findest – es dient dazu, eine Seitenübersicht aufzurufen; tippe eine Seite an, um sie zu öffnen.



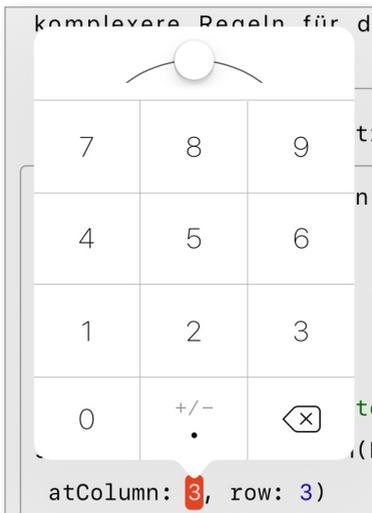
Tippe auf das Listensymbol, um die Seitenübersicht aufzurufen.

Mit dem Programmieren wirst du erst in Kapitel 4 starten, aber ich gebe dir jetzt schon mal einen kleinen Einblick, wie du zur Eingabe deiner Befehle vorgehst. Auch in diesem Zusammenhang gibt es wieder mehrere Möglichkeiten:

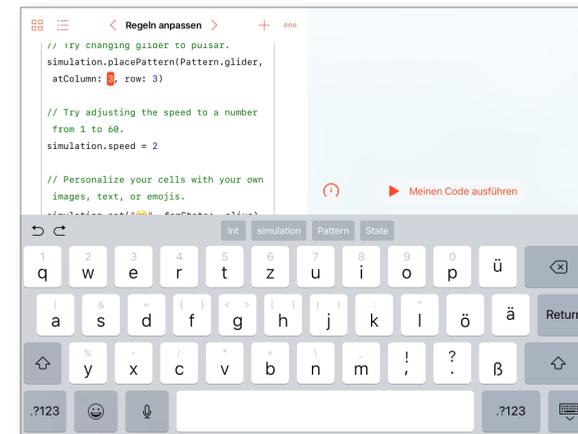
- ▶ **Befehlsleiste:** Tippst du im Textbereich in ein Eingabefeld, so wird unten in der App automatisch eine Befehlsleiste eingeblendet. Die Befehlsleiste enthält Befehle, die gerade passen könnten und die du durch Antippen in das Eingabefeld einfügen kannst – und zwar dort, wo die Eingabemarkierung blinkt. Die Eingabemarkierung kannst du übrigens ganz einfach an eine andere Stelle setzen, indem du diese antippst. Die Befehlsleiste enthält noch weitere Symbole: Links in der Befehlsleiste siehst du Symbole zum Rückgängigmachen und Wiederherstellen von Änderungen; rechts in der Leiste befindet sich ein Korrektursymbol zum Löschen der links von der Eingabemarkierung befindlichen Zeichen, ein Symbol zum Erzeugen eines neuen Absatzes sowie ein nach oben weisender Pfeil, mit dem du die Bildschirmtastatur einblendest.



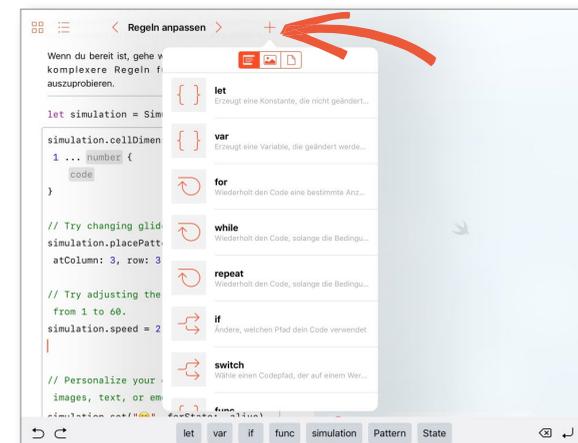
- ▶ **Popup-Fenster:** Ist im Programm die Eingabe einer Zahl erforderlich, markiere das entsprechende Element. Es öffnet sich ein Popup-Fenster, in dem du die gewünschte Zahl eingeben kannst.



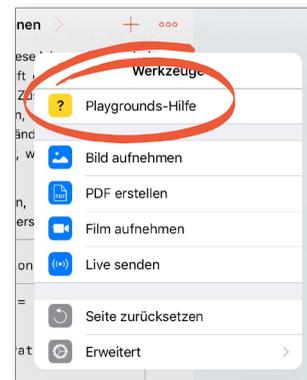
- ▶ **Tastatur:** Wie gesagt, kannst du über den nach oben weisenden Pfeil in der Befehlsleiste auch die Bildschirmtastatur eingeben und damit deine Befehle, sonstigen Text oder auch Zahlen eintippen. Möchtest du die Bildschirmtastatur wieder ausblenden, tippe auf das Tastatursymbol rechts unten in der Bildschirmtastatur.



- ▶ **Menü zum Einfügen:** Schließlich wird dir auch noch ein Menü zum Einfügen von Befehlen und weiteren Inhalten angeboten. Dieses Menü öffnest du, indem du oben im Playground auf das kleine Plusymbol tippst. Um im Menü einen Befehl auszuwählen – und an der Stelle der Eingabemarkierung einzufügen –, tippst du den Eintrag an. Anschließend machst du deine Angaben im Textbereich.



Wie dir sicher aufgefallen ist, gibt es oben im Playground noch ein weiteres Symbol, das drei Punkte zeigt. Unter diesem Symbol verbergen sich verschiedene Werkzeuge. Wieder tippst du das Symbol an und wählst dann im Menü, das sich öffnet, einen Eintrag aus.



Die App Swift Playgrounds bietet eine eingebaute Hilfefunktion zum schnellen Nachschlagen. Um die Hilfefunktion aufzurufen, tippst du oben im Playground auf das Drei-Punkte-Symbol und wählst im Menü den Eintrag **Playgrounds-Hilfe**. Im folgenden Fenster tippst du links oben auf das Listensymbol, um ein Thema auswählen zu können oder um ein Suchfeld zum Nachschlagen eines Begriffs zu verwenden.



Die App Swift Playgrounds bietet eine eingebaute Hilfefunktion, die dir auf die Schnelle wichtige Fragen beantworten kann.



Direkte Hilfe zu einem Befehl erhalten

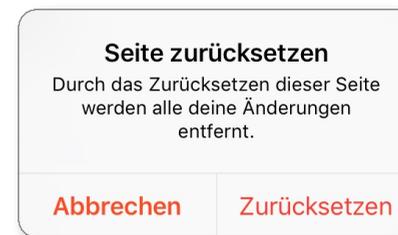


Beim Programmieren kannst du auch ganz einfach direkte Hilfe erhalten. Tippe dazu einen Befehl an, zu dem du Hilfe benötigst. Wähle im Menü, das dir angezeigt wird, **Hilfe**. Nun werden dir in einer Sprechblase Informationen zum Befehl angezeigt.



Deine Programmierfortschritte lassen sich auf unterschiedliche Weise aufzeichnen.

In dem Menü, das du öffnest, indem du auf das Drei-Punkte-Symbol tippst, findest du noch weitere nützliche Funktionen. Möchtest du von einem Playground ein Bild oder ein Video aufnehmen oder den Playground vielleicht übers Internet übertragen? Oder einfach eine PDF-Datei – also ein Dokument – daraus erstellen? Auch diese Funktionen findest du im Menü. Ich werde sie dir in Kapitel 9 noch ausführlich vorstellen.



Möchtest du in einem Playground von vorne anfangen? Dann setze mit wenigen Handgriffen die Seite zurück.

Eine weitere Funktion, die du manchmal benötigen wirst, ist das Zurücksetzen der Seite. Du weißt bereits, wie du den ganzen Playground zurücksetzen kannst. Möchtest du lediglich die gerade geöffnete Seite im Playground in den ursprünglichen Zustand zurücksetzen, tippst du auf das Drei-Punkte-Symbol und wählst im Menü den Eintrag **Seite zurücksetzen**. Bestätige anschließend noch mal mit **Zurücksetzen**.

Und falls du dich dafür interessierst, aus welchen Elementen ein Playground besteht, bringst du das in Erfahrung, indem du auf das Drei-Punkte-Symbol tippst und im Menü den Eintrag **Erweitert** wählst. Entscheide dich anschließend für **Hilfsquelldateien anzeigen** und tippe – im Querformat – links einen Eintrag an, um rechts die Infos dazu einzublenden.

Sound einstellen, Spielfigur auswählen und weitere Einstellungen

Gleich geht es mit Lektion 1 – also: *Programmieren lernen 1* – los. Rufe diese jetzt schon mal in der App Swift Playgrounds auf. Ich stelle dir noch ein paar Einstellungen vor, die du im Playground vornehmen kannst, bevor du mit dem Programmieren startest.

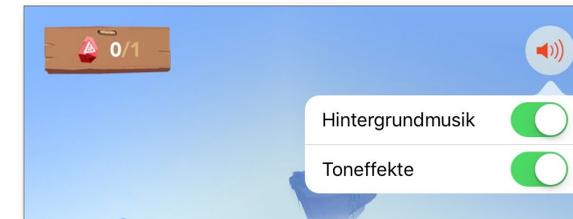
Wenn du den Playground öffnest, wird dir zunächst mal ein Einführungstext angezeigt. Lies dir diesen Text in Ruhe durch. Blättere die Seiten weiter, indem du unten auf die Pfeilsymbole tippst. Wenn du später einen Playground erneut lädst, kannst du die Einführung auch wegklicken, indem du rechts oben auf dem Bildschirm auf das Kreuzsymbol tippst.



Wenn du ein neues Kapitel öffnest, wird dir erst mal ein Einführungstext angezeigt, den du dir durchlesen solltest.

Nachdem du den Einführungstext gelesen hast, wird dir die bereits bekannte Playground-Ansicht angezeigt, die in den Textbereich links und den Anzeigebereich rechts unterteilt ist. Du stellst außerdem fest, dass eine Hintergrundmusik und Toneffekte abgespielt werden. Falls du das als störend empfindest, kannst du

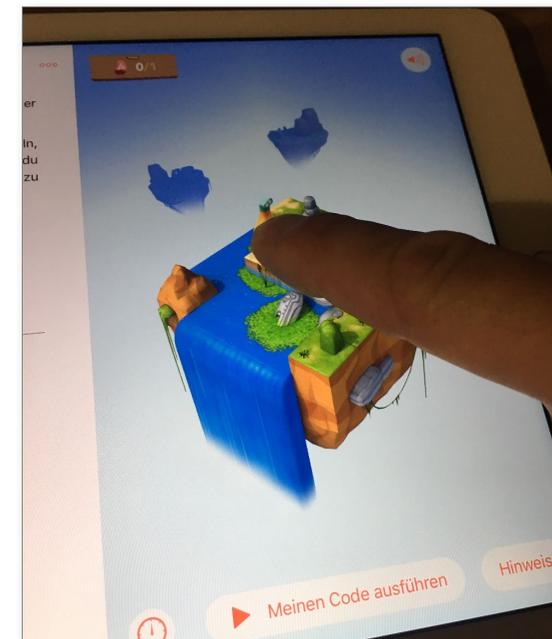
beides abstellen. Dazu tippst du rechts oben im Anzeigebereich auf das Lautsprechersymbol. Im Menü, das sich öffnet, kannst du nun per Schalter bestimmen, ob du dir die Hintergrundmusik beziehungsweise Toneffekte anhören möchtest oder nicht.



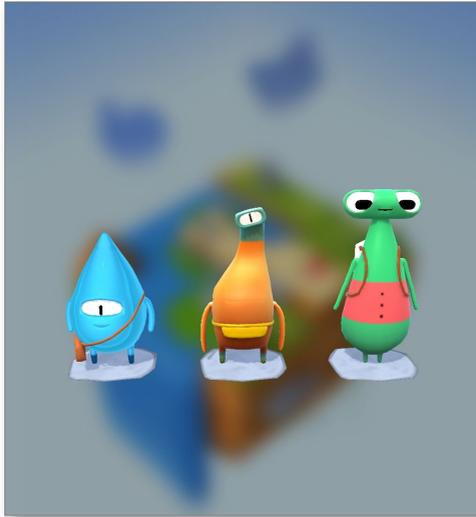
Tippe auf das Lautsprechersymbol, um Hintergrundmusik und Toneffekte einzustellen.

In der ersten Lektion steuerst du normalerweise die Spielfigur Byte durch eine Fantasiewelt. Wenn dir diese Spielfigur nicht zusagt, kannst du sie aber auch austauschen – als Spielfiguren stehen neben Byte auch Hopper und Blu zur Verfügung. So gehst du zum Austauschen der Spielfigur vor:

1. Tippe zweimal schnell hintereinander auf die Spielfigur – in diesem Fall Byte. Die Spielfigur fliegt daraufhin aus der Fantasiewelt.



- Nun werden dir Blu (die blaue Spielfigur), Byte (in der Mitte) und Hopper (der grüne Grashüpfer) als Spielfiguren zur Auswahl angeboten. Tippe die Spielfigur an, die du verwenden möchtest.



- Hier habe ich mich beispielsweise für die Spielfigur Hopper entschieden, die prompt in die Fantasiewelt hineinfliegt und sich dort durch Programmierbefehle steuern lässt.



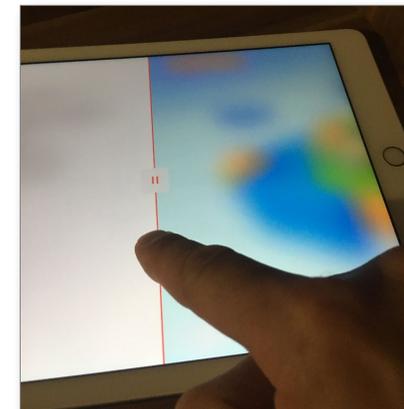
Dreh und vergrößere die Welt!

Die Fantasiewelt, in der du die Spielfigur bewegst, kannst du aus unterschiedlichen Richtungen und Größen betrachten. Um sie zu drehen, tippe mit dem Finger in den Anzeigebereich und ziehe ihn dann nach links oder rechts – je nachdem, in welche Richtung du die Fantasiewelt drehen möchtest. Zum Vergrößern tippst du mit Daumen und Zeigefinger in den Anzeigebereich und bewegst die beiden Finger auseinander (oder zueinander, um die Ansicht zu verkleinern). Um wieder die ursprüngliche Blickrichtung und Größe zu erhalten, tippe zweimal schnell hintereinander in den Anzeigebereich (aber nicht auf die Spielfigur) – die Fantasiewelt wird daraufhin wieder in den ursprünglichen Zustand zurückversetzt.

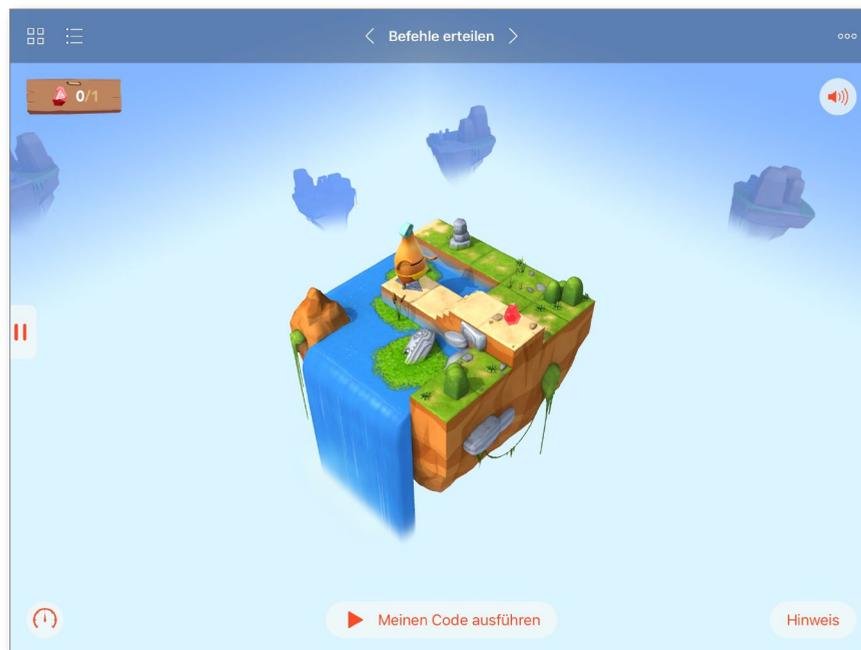


Ebenfalls gut zu wissen: Sowohl der Textbereich als auch der Anzeigebereich lassen sich auf den ganzen iPad-Bildschirm ausdehnen. So hast du mehr Platz – entweder zum Programmieren oder zum Spielen. Und so geht's:

- Tippe im Playground auf die Grenze zwischen Textbereich und Anzeigebereich, und halte diese gedrückt. Die beiden Bereiche werden ausgeblendet, und es wird dir stattdessen eine Linie angezeigt.



2. Halte die Linie weiter gedrückt, und ziehe sie entweder an den rechten Bildschirmrand (im Hochformat: an den oberen Bildschirmrand), um den Textbereich auszudehnen, oder aber an den linken Bildschirmrand (im Hochformat: an den unteren Bildschirmrand), um den Anzeigebereich auszudehnen.
3. In diesem Fall habe ich den Anzeigebereich auf den ganzen iPad-Bildschirm ausgedehnt. Um wieder die ursprüngliche Bildschirmteilung zu erhalten, tippst du auf den Bildschirmrand, an dem sich die zwei Striche befinden, hältst ihn gedrückt und ziehst die Linie zurück in die Mitte.



Du kennst jetzt alle wichtigen Funktionen der App Swift Playgrounds und weißt, wie du einen Playground nach deinen eigenen Vorstellungen einrichten kannst. Lege gleich mit deinen ersten Programmierschritten los – das nächste Kapitel wird dich dabei begleiten!

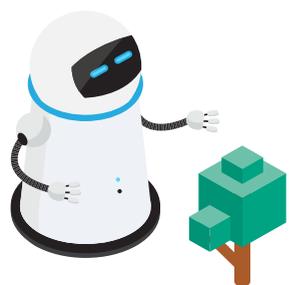
Auf einen Blick

| | | |
|-----------|---------------------------------------------------------------------------------------------|-----|
| 1 | Gestatten, Swift! | 17 |
| 2 | Programmieren – ein paar Grundlagen | 33 |
| 3 | Ab in die Playgrounds! – Swift spielerisch erlernen | 39 |
| 4 | Lektion 1 oder: Wie man seltsame Wesen Edelsteine sammeln lässt | 63 |
| 5 | Lektion 2 oder: Wie man in fremden Welten zählt und ordnet | 115 |
| 6 | Lektion 3 oder: Wie du die Game-Welt ganz nach deinen Vorstellungen gestaltest | 163 |
| 7 | Weitere tolle Playgrounds | 197 |
| 8 | Mit den passenden Startpunkten eigene Projekte umsetzen | 229 |
| 9 | Deine Fortschritte fotografieren, filmen und verbreiten | 265 |
| 10 | Apps veröffentlichen – so machen es die Profis | 275 |

Inhalt

| | |
|-----------------------------------------------------------------------------|----|
| Geleitwort: Programmieren im Unterricht mit Swift Playgrounds | 9 |
| Vorwort: Was dich in diesem Buch erwartet | 11 |
| Kapitel 1: Gestatten, Swift! | 17 |
| Was sind Apps, und wer erstellt die? | 17 |
| Welche Apps gibt es bereits? | 21 |
| Warum Swift eine flinke Programmiersprache ist | 24 |
| So ist Swift aufgebaut | 26 |
| Was du in diesem Buch alles lernst – und was nicht | 28 |
| Kapitel 2: Programmieren – ein paar Grundlagen | 33 |
| Was ist ein Computer? | 33 |
| Was bedeutet Programmieren, und welche Programmiersprachen gibt es? | 35 |
| Kapitel 3: Ab in die Playgrounds! – Swift spielerisch erlernen | 39 |
| So erstellst du deine eigene Apple-ID | 39 |
| Swift Playgrounds auf dem iPad installieren | 41 |
| Playgrounds herunterladen | 44 |
| Playgrounds verwalten | 47 |
| Playground öffnen und Bedienoberfläche kennenlernen | 51 |
| Sound einstellen, Spielfigur auswählen und weitere Einstellungen | 58 |





Kapitel 4: Lektion 1 oder: Wie man seltsame Wesen Edelsteine sammeln lässt 63

Was du in dieser Lektion lernst und welche Wörter du kennen solltest 63

Schritt für Schritt: Die Inhalte der ersten Lektion erarbeiten 70

Die Kommandos der ersten Lektion im Überblick 111

Kapitel 5: Lektion 2 oder: Wie man in fremden Welten zählt und ordnet 115

Was du in dieser Lektion lernst und welche Wörter du kennen solltest 115

Schritt für Schritt: Die Inhalte der Lektion erarbeiten 119

Die Kommandos der zweiten Lektion im Überblick 160

Kapitel 6: Lektion 3 oder: Wie du die Game-Welt ganz nach deinen Vorstellungen gestaltest 163

Was du in dieser Lektion lernst und welche Wörter du kennen solltest 163

Schritt für Schritt: Die Inhalte der Lektion erarbeiten 168

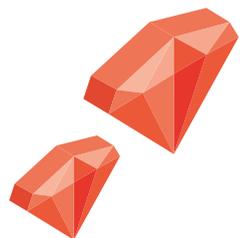
Die Kommandos der dritten Lektion im Überblick 195

Kapitel 7: Weitere tolle Playgrounds 197

Herausforderungen:

Knifflige Programmieraufgaben lösen 197

Zubehör: Geräte mit Swift Playgrounds steuern 206



Kapitel 8: Mit den passenden Startpunkten eigene Projekte umsetzen 229

Startpunkt herunterladen 229

Weitere Seiten hinzufügen 231

Fragen und Antworten 233

Formen 243

Grafische Darstellung 258

Rätselwelt 261

Leer 262

Von der Idee bis zur Veröffentlichung:

So gehst du an ein App-Projekt heran 262

Kapitel 9: Deine Fortschritte fotografieren, filmen und verbreiten 265

Code in einer PDF-Datei speichern und versenden 265

Code als Text kopieren und in ein Dokument einfügen 268

Den Anzeigebereich abfotografieren 269

Die Aktionen auf dem Bildschirm filmen 271

Live-Übertragung über Drittanbieter-Apps 272

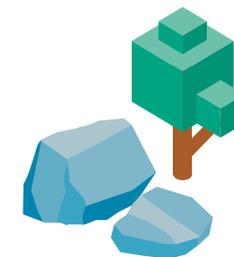
Kapitel 10: Apps veröffentlichen – so machen es die Profis 275

Ein Mac ist verfügbar? So holst du dir Xcode 275

Erste Schritte mit Xcode 279

Der lange Weg vom Code zur eigenen App 286

Berufswunsch App-Entwickler? 292





Anhang

| | |
|----------------------------|-----|
| Kommandos | 295 |
| Glossar | 300 |
| Stichwortverzeichnis | 305 |

Stichwortverzeichnis

A

Abo-Angebote 19
 Aktualisierungen 69
 Algorithmen 69, 107, 303
 AND 67, 113, 298
 Antwortmöglichkeiten 237
 Anwendung 17
 Anzeigebereich 52
 App 11, 17
 append 152
 App-Entwickler 292
 Apple Developer Program 289
 Apple-ID 19
 erstellen 39
 App Store 18
 Rubriken 22
 Suchen 22
 Argument 137
 Array 119, 151, 300
 Astrodance 164
 Autor 15

B

Bedienoberfläche 52
 Bedingte Anweisungen 66, 91, 298
 Bedingung 103
 Befehle 64, 70
 Befehlsleiste 54
 Benutzerkonto 19
 Berührungseignis 167, 179, 299

Beschriftung 174
 Bildschirmastatur 54
 Bildschirmteilung 62
 Blickrichtung 61
 Block 145
 Blu 59
 Bool 296
 Boolesche Werte 94, 160
 Bug 64, 75, 302
 Bugfix 302
 Byte 59

C

C# 38
 C++ 38
 Captions 174
 Character 135
 Code
 als Bild speichern 269
 ausführen 52
 drucken 266
 kopieren 268
 speichern und teilen 265
 Codekommentar 152, 168, 264, 295
 Compiler 36, 302
 Computer 33
 Computerprogramm 17
 Computersprache 36
 condition 103
 Container 120
 CSS 38

D

Datei 301
 Datentypen 295
 Datenverarbeitung 301
 Dauerschleife 106
 Debuggen 75
 Debugger 302
 Dekomposition 84
 Developer Program 289
 Dezimalzahlen 296
 Double 296
 Download-Bereich 198
 Draw 165
 Drohne 30, 206
 abheben 210
 bedingte Bewegung 222
 Beschleunigungsmesser 226
 Fernbedienung 227
 Flips 218
 Gas geben 212
 Geschwindigkeit 217
 gieren 215
 Grabber 222
 Kamera 219, 224
 komplexe Bewegungen 220
 landen 210
 nicken 214
 rollen 213
 Spirale 221
 Duplizieren 232
 Durchmesser 248

E

E-Bass 192
 Edelsteine 64
 EDV 301
 E-Gitarre 192
 Eigene App
 programmieren 262
 Richtlinien 287
 veröffentlichen 275
 Eigenschaften 117
 Einfügen 268
 else 66, 91, 112, 298
 E-Mail-Adresse 40
 Emojis 169
 Entwickler-Konto 39
 Entwicklerprogramm 289
 Entwicklungsumgebung 36,
 277, 301
 Ereignishandler 167, 186, 299
 EV3-Roboter 206
 Expert 132

F

false 94, 160, 296
 Farbsättigung 251
 Farbton 252
 Fehler 75
 Float 296
 for 66, 112
 for-Schleife 66, 86, 112, 297
 Fragebogen 234
 FreeFlight Mini 207
 func 65, 112
 Funktionen 65, 76, 112,
 117, 297

G

Ganzzahl 120
 Gleichheitszeichen 160, 299
 Glossar 300
 Google 41
 Grafiken 174

H

Hardware 33
 Herausforderungen 29, 197
 Hilfsfunktion 56
 Hilfsquelldateien 57
 Hintergrundmusik 58
 Hinweis 71
 Hochformat 43
 Hopper 59
 hue 252

I

iCloud Drive 51
 IDE 301
 if 66, 91, 112, 298
 iMac 275
 iMac Pro 276
 In-App-Käufe 19
 Index 151, 300
 Informatik 300
 Informationstechnologie 301
 Initialisierung 117, 132
 insert 152
 Instanz 128
 Instanzen 117
 Int 296
 Integer 120, 296
 Int-Wert 120

iOS 18, 24
 iPad 18
 iPhone-Simulator 285
 IT 301
 Iteration 152
 iTunes-Guthabekarten 20

J

Java 37
 JavaScript 37

K

Kaleidoskop 165
 Kaufpreis 20
 Klammern 161
 Klassen 27, 117
 Klavier 192
 Kommando 111
 Bool 296
 Datentypen 295
 Double 296
 Float 296
 Int 296
 Kommentar 295
 String 296
 Überblick 295
 Kommazahlen 183
 Kommentar 152
 Komposition 77
 Konstanten 116, 120, 297
 Koordinaten 146, 166, 170,
 196, 299
 Koordinatensystem 170
 Kreditkarte 20
 Kreiszahl 248

L

Label 284
 Laden 41
 Lektion 45
 Programmieren lernen 1 63
 Programmieren lernen 2 115
 Programmieren lernen 3 163
 let 117, 160
 Linux 25
 Lite-Apps 19
 Logische Operatoren 67, 97,
 113, 298

M

MacBook 276
 Mac mini 277
 macOS 25
 Mac Pro 277
 Malnehmen 120
 Menü 55
 Methoden 117
 Mikroprozessor 34
 Minidrohne 30
 Minus-Rechnen 120
 Mitteilungen 43
 Multiple-Choice-Quiz 238
 Musikinstrument 199
 Muster 81

N

NOT 67, 113, 298
 Notizen 268
 number 86

O

Objective-C 24, 38
 Objekte 27
 Objekttypen 117
 Operatoren 67, 298
 options 237
 OR 67, 113, 298

P

Parameter 118, 136, 161, 299,
 303
 Parrot Mambo 30, 206
 Passwort 40
 PayPal 20
 Personal Computer 34
 PHP 37
 Pi 248
 Play 166
 Playground
 duplizieren 50
 filmen 271
 herunterladen 44
 löschen 49
 streamen 272
 zurücksetzen 50
 Plus-Rechnen 120
 Portal 74
 Position 299
 Programm 33
 Programmfehler 302
 Programmieren lernen 1 63
 Programmieren lernen 2 115
 Programmieren lernen 3 163
 Programmierer 35
 Programmiersprache 24, 35
 objektorientiert 27
 Provision 20

Prozessor 34
 Pseudocode 108
 Punktnotation 161, 299
 Python 37

Q

Quelltext 26, 36
 Querformat 43
 Quiz 238

R

Radius 244
 remove 151
 Ruby 38

S

saturation 251
 Schalter 64
 Schleifen 297
 Schrägstriche 295
 Schriftart 176
 Schriftfarbe 176
 Schriftgröße 176
 Screenshot erstellen 270
 Seiten 231
 Seitenübersicht 53, 64
 Senkrechte 166
 Sicherheitsfragen 40
 Software 33
 Sound 190
 Spiele-App 12
 Spielen 166
 Spielfigur 59
 Sprachausgabe 199
 Startpunkt 45, 229

Antworten 233
Formen 243
Grafische Darstellung 258
Leer 262
Store 18
String-Wert 120
Swift 11
Swift Playgrounds
 Bibliothek 44
 installieren 41
 Lektion 1 63
 Playgrounds herunterladen 44, 46
 Playgrounds verwalten 47
Syntax 303

T

Tastatur 55
Teilen 120
Text 174
Textbereich 52
Töne zeichnen 198
Tool 187
touch 196, 299
Treppe 149

Trommeln 192
true 94, 160, 296
tvOS 25
Typen 117, 128, 235

U

Umfang 248
Umsatzsteuer 20

V

var 117, 160
Variablen 116, 117, 120,
 297, 303
Vergleichsoperator 117,
 131, 160, 299
Vergrößern 61
Verschachteln 105
Vorschau 164

W

Waagerechte 166
watchOS 24
Weltenbau 118, 145
while 68, 113

while-Schleife 68, 102, 113, 297
World Wide Web 35

X

x-Achse 166, 170
Xcode 264, 302
 eigenes Projekt 280
 erste Schritte 279
 installieren 277
 iPhone-Simulator 285
x-Wert 299

Y

y-Achse 166, 170
y-Wert 299

Z

Zahlungsmethode 19
Zeichnen 165
Zubehör 45
Zugangsdaten 40
Zuweisungsoperator 117, 131,
 160, 299
Zuweisungsparameter 117

Sehen wie's geht!



Philip Kiefer

Programmieren lernen mit Swift Playgrounds

308 Seiten, broschiert, November 2017
24,90 Euro, ISBN 978-3-8421-0308-5

Direkt bestellen!

 www.rheinwerk-verlag.de/4429



Philip Kiefer zählt zu den bekanntesten Fachbuchautoren in Deutschland. Bis heute hat er mehr als 150 Bücher veröffentlicht, neben vielen Publikationen für Kinder und Jugendliche auch über 70 Bücher zu Windows, Office und weiteren Computerthemen. Sein immenses Computerwissen hat er sich in über drei Jahrzehnten selbst beigebracht. Dabei versteht er es perfekt, selbst komplexe Sachverhalte so zu vermitteln, dass sie ein jeder leicht begreift.

Empfehlen Sie uns weiter!

Wir hoffen sehr, dass Ihnen diese Leseprobe gefallen hat. Sie dürfen sie gerne empfehlen und weitergeben, allerdings nur vollständig mit allen Seiten. Diese Leseprobe ist in all ihren Teilen urheberrechtlich geschützt. Alle Nutzungs- und Verwertungsrechte liegen beim Autor und beim Verlag.



www.facebook.com/rheinwerkverlag