

Damit man eigene Android-Apps programmieren kann, benötigt man neben dem SDK für Java (das bei einer Installation von Eclipse ja mit dabei ist) noch das SDK für Android von Google. [Google](#) stellt Entwicklern aber nicht nur das SDK zur Verfügung, sondern auch Tutorials sowie Codebeispiele und gibt Tipps und Tricks. Eine Dokumentation finden Sie unter [developer.android.com](#).

Übrigens: Wie bei jeder Smartphone-Plattform können Sie auch plattformübergreifend mit Javascript, HTML und CSS programmieren.

Wie entwickelt man eine App mit Android Studio?

Nach einem ähnlichen Prinzip funktioniert auch Googles freie Entwicklungsumgebung [Android Studio](#). Allerdings solltest du besser ein paar Java-Kenntnisse mitbringen, wenn du damit eine App selbst programmieren möchtest.

Doch fangen wir am besten von vorne an: Bevor du deine eigenen [Android-Apps](#) mit **Android Studio** programmieren kannst, benötigst du die aktuelle Version des Java Development Kit. Installiere dieses und Android Studio auf deinem PC, und schon kann es losgehen.

1. Leg zunächst ein neues Projekt an („Empty Activity“), gib ihm einen Namen, entscheide dich für die Geräte, auf denen die App laufen soll, und wähle die Android-Version aus, für die du die App programmieren möchtest.
2. Nun öffnet sich die Oberfläche, auf der du deine App gestalten kannst. Links befinden sich die Dateien, aus der sich deine App am Ende zusammensetzt. Alle zusammen bilden ein Projekt.
3. Um ein Layout zu entwerfen, greifst du auf den **Werkzeugkasten** zurück. Er befindet sich zwischen der Dateiliste und dem simulierten Interface und bietet dir zahlreiche Elemente, um die Oberfläche deiner App zu gestalten.
4. Wenn du schon weißt, wie man eine App entwickelt bzw. bereits Java-Vorkenntnisse hast, kannst du an der oberen Leiste direkt in das Java-Menü gehen und auf den Code zugreifen.
5. Über einen **Emulator** kannst du testen, wie deine App auf verschiedenen Smartphone-Modellen aussieht.

Wenn du das erste Mal eine App selbst programmieren möchtest, ist Android Studio vielleicht nicht ganz so leicht zu durchschauen. Mithilfe von YouTube-Tutorials und ein wenig Internetrecherche kannst du dich aber garantiert durch das Programm wurschteln.

Wie entwickle ich eine App in Android-Studio?

Um dir Android-Studie downloaden zu können, musst du zuerst darauf achten, die aktuelle Version des Java Development Kits bereits auf deinem Rechner vorinstalliert zu haben. Erst dann kann das Google-Programm vollständig auf deinem Computer installiert werden.

Wenn du Unterstützung brauchst beim Programmieren deiner Android-App in Android-Studio, gibt es zahlreiche hilfreiche Tutorials, die dich dabei unterstützen. Damit du allerdings einen kleinen Vorgeschmack davon bekommst, was dich in etwa erwarten wird, haben wir einen kurzen Guide für dich erstellt:

1. Öffne nach der Installation das Programm und lege über "Start a new Android Studio project" ein neues Projekt an: Wähle einen Namen für deine App und mit welchen Geräten sie funktionieren soll, etc.
2. Nachdem du auf "Weiter" geklickt hast, wählst du als Projektart "Blank Activity".
3. Jetzt bist du auf der Projektoberfläche. In der linken Menüseite siehst du Dateien untereinanderaufgelistet, die das ganze Projekt bilden und am Ende gemeinsam für die fertige App exportiert werden. Lösche hier niemals die Datei xml.
4. In der Mitte gibt es die Darstellung einer Smartphone-Oberfläche und daneben befindet sich dein Werkzeugkasten. Daran kannst du dich bedienen, um deine Android-App zu programmieren.
5. Hast du schon Java-Vorkenntnisse, kannst du auch direkt auf den Code zugreifen, um deine App zu entwickeln.

Android-Apps mit den Sprachen des Webs entwickeln

Wer bereits Erfahrungen im Webdesign gesammelt hat, für den könnte [Apache Cordova](#) etwas sein.

- Dadurch, dass unter Cordova die App mit **HTML, CSS und Javascript** programmiert wird, lässt sich der Code auch auf anderen Plattformen wie zum Beispiel einer Webseite für stationäre Computer nutzen. Das bedeutet, dass es einen Code für mehrere Plattformen gibt. Zudem bietet Cordova die Möglichkeit, Apps offline zu nutzen.
- Cordova läuft unter Windows, Mac sowie Linux und benötigt neben [Node.js](#), [Gradle](#), und [Java SDK 8](#), zusätzlich [Android Studio](#).
- Hat man Node.js zum Laufen gebracht, kann man mit dem Tool **npm** Cordova auf der Komandozeile installieren. Für die Installation der anderen Programme gehen Sie wie in der [Dokumentation](#) beschrieben vor.
- Sobald die App kompiliert wurde, lädt **npm** einen lokalen Webserver, sodass Sie die App im Browser betrachten können.

Web Apps (oder HTML5-Apps)

Wer nur ungern Dinge doppelt und dreifach schreibt und sich direkt zum Anfang nicht mit den unterschiedlichen Programmiersprachen der verschiedenen Hersteller rumschlagen möchte, greift zu dieser Lösung. Mit Webtechnologien kann man auch ganze Anwendungen schreiben. Und überall wo ein moderner Browser ausgeführt werden kann, kann auch ein Webbasiertes Programm ausgeführt werden. Es gibt tolle Bibliotheken, [wie Apache Cordova, mit denen man eine webbasierte App programmieren kann](#) und quasi dieselbe Funktionalität nutzen kann, die man von nativen Apps kennt. Sogar in die AppStores lassen sich diese Apps bringen. Das coole: Man entwickelt quasi eine normale responsive Webanwendung und kann diese mit ein paar Anpassungen zur vollfunktionsfähigen App umbauen. Die App sieht auf

allen Plattformen gleich aus. Man lernt zunächst nur eine Programmiersprache ([Javascript](#)) und kann damit fast alles bauen, was du dir vorstellen kannst. Für jemanden der anfängt und noch nicht so richtig weiß, was er eigentlich bauen möchte ist das ideal.

In diesem Tutorial entwickeln wir eine Android™ App, in die wir eine SQLite Datenbank integrieren

In 8 großen Lektionen werden wir gemeinsam eine Android App entwickeln und in diese eine SQLite Datenbank integrieren. Dabei werden wir am Beispiel einer eigenen App vermitteln, wie eine SQLite Datenbank in Android eingebunden und verwendet wird.

Als SQLite Anwendung werden wir eine Einkaufslisten-App programmieren, mit der eine eigene Einkaufsliste erstellt werden kann. Die Einträge der Einkaufsliste werden mit Hilfe einer SQLite Datenbank verwaltet. Im Laufe des Kurses werden wir erfahren, wie Datensätze in die Datenbank geschrieben, aus ihr ausgelesen, verändert und auch gelöscht werden.

Die eigene Android App wird dabei schrittweise entstehen und am Ende des Kurses zu einer vollständigen Android Anwendung angewachsen sein.

Auf dieser Übersichtsseite des Android SQLite Tutorials stellen wir die einzelnen Lektionen kurz vor. Wir empfehlen die Beschreibungen der einzelnen Lektionen zu lesen und anschließend mit der Lektion zu beginnen. Ihr könnt aber auch per Schnelzugriff direkt zur jeweiligen Tutorial Lektion springen.

+ Lektion 1: Das *SQLite* Projekt in Android Studio anlegen

+ Lektion 2: Integrieren einer *SQLite* Datenbank in Android

+ Lektion 3: Erstellen einer Tabelle in die *SQLite* Datenbank

+ Lektion 4: Daten in die *SQLite* Datenbank schreiben und auslesen

+ Lektion 5: Mit der *SQLite* App auf Benutzereingaben reagieren

+ Lektion 6: Datensätze aus der *SQLite* Datenbank löschen

+ Lektion 7: Datensätze in der *SQLite* Datenbank ändern

+ Lektion 8: Durchführen eines *SQLite* Datenbank Upgrades in Android

<https://www.programmierenlernenhq.de/android-sqlite-datenbank-tutorial/>

Verbindung zu git von AS

Zusätzlich: Spring Boot Server

Wie ist es dann mit dem Testen?

Flutter in Android Studio verwenden?

Brauchen wir dann zusätzlich n Server wie z.B. Spring Boot oder Node.js?