

Modul 02

# Grundlagen der Programmierung

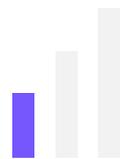
Nanodegree "Programmierer/in"

# Modulübersicht

Das Erlernen des Programmierens ist heutzutage von großer Bedeutung. Es ermöglicht uns, die digitale Welt zu verstehen und zu gestalten. Programmieren fördert logisches Denken, Problemlösungsfähigkeiten und Kreativität. Es eröffnet uns zahlreiche Karrieremöglichkeiten in der Technologiebranche. Programmieren kann uns dabei helfen, innovative Lösungen für Alltagsprobleme zu entwickeln und unsere kognitive Flexibilität zu verbessern. In einer digitalen Welt ist das Erlernen des Programmierens von unschätzbarem Wert.

## Zielgruppe

Anfänger



## Zeitaufwand

2 Monate bei 10h/Woche\*



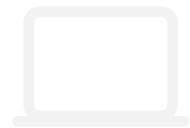
## Software

IntelliJ



## Hardware

PC



\* Die Dauer dieses Programms ist eine Schätzung der Gesamtstunden, die der durchschnittliche Student benötigt, um alle erforderlichen Kursarbeiten abzuschließen, einschließlich Vorlesungs- und Projektzeit. Wenn Sie etwa 10-15 Stunden pro Woche mit dem Programm arbeiten, sollten Sie es innerhalb der angegebenen Zeit abschließen. Die tatsächlichen Stunden können variieren.

# Grundlagen der Programmierung



Die Lernenden werden eine Reihe von Coding-Challenges absolvieren, um ihr Verständnis von Kotlin zu testen. Es wird Übungen für Variablen, Strings, if (else-if und else)-Anweisungen und Funktionen geben.

## Projekt

### Bundesliga Simulator

Bauen Sie eine einfache Konsolen-Anwendung mit Kotlin, um das Programmieren und die Programmiersprache besser kennen zu lernen. Diese Anwendung gibt Ihnen aus einer Zufälligen Anzahl an Teams eine Simulierte Saison aus, und bestimmt am Ende den Gewinner. Anhand dieses Beispiels werden zum einen wichtige Programmierkenntnisse erworben und Sie lernen die Besonderheiten der Programmiersprache Kotlin kennen, die zum Entwickeln von Android Anwendungen benötigt wird.

## Thema 01

### Variablen und Datentypen

- Deklarieren Sie Variablen und Konstanten mit grundlegenden Typen wie Bool, Int, Double und Float.
- Greifen Sie auf Werte von Variablen und Konstanten zu und ändern Sie diese.
- Beheben Sie Compiler-Probleme im Zusammenhang mit der falschen Verwendung von Variablen und Konstanten.
- Verwenden Sie Escape-Zeichen und String-Interpolation, um Variablen- und Konstantenwerte innerhalb von Strings zu formatieren.

## Thema 02

### Rechenoperatoren

- Berechnen Sie neue Werte unter Verwendung vorhandener Variablen und Konstanten.
- Verwenden Sie Vergleichsoperatoren, um die Gleichheit zwischen zwei Werten zu bestimmen.
- Verwenden Sie boolesche Operatoren, um Ausdrücke aufzubauen, die Wahrheitswerte verwenden.

## Thema 03

### Verzweigungen & Schleifen

- Schreiben Sie boolesche Ausdrücke, um Entscheidungslogik darzustellen.
- Kombinieren Sie boolesche Ausdrücke mit logischen Operatoren.
- Verwenden Sie boolesche Ausdrücke zusammen mit if, else-if und else-Anweisungen, um den Ablauf der Codeausführung zu steuern.
- Verwenden Sie switch-Anweisungen, um Code basierend auf mehreren Werten einer einzigen Variablen auszuführen.
- Verwenden Sie for-, while- und do-while-Schleifen, um den Ablauf der Codeausführung zu steuern.

## Thema 04

### Funktionen

- Bestehenden Code in wiederverwendbare Funktionen einpacken.
- Funktionen korrekt definieren und aufrufen.
- Funktionen mit Parametern und Rückgabetypen spezifizieren.
- Unterscheidung zwischen Werten, die im Geltungsbereich (in-scope) und außerhalb des Geltungsbereichs (out-of-scope) liegen.
- Lokale und externe Parameter korrekt verwenden.
- Parametertypen und Rückgabetypen identifizieren.

## Thema 07

### Strings manipulieren

- Strings mithilfe ihrer integrierten Eigenschaften und Methoden definieren und bearbeiten.
- Häufige String-Operationen wie Verkettung (Concatenation) und das Finden von Teilstrings durchführen.
- Häufige String-Manipulationen wie das Hinzufügen, Entfernen und Ersetzen von Teilstrings durchführen.

## Thema 08

### Listen und Maps

- Ungeordnete Daten des gleichen Typs mit Listen speichern.
- Auf Listen-Inhalte zugreifen und diese ändern.
- Schlüssel-Wert-Paare mit Maps speichern.
- Auf Dictionary-Inhalte zugreifen und diese ändern.
- Ungeordnete Daten des gleichen Typs mit Sets speichern.

## Thema 09

### Objektorientierte Programmierung

- Den Unterschied zwischen Wert- und Referenztypen verstehen und wie dies auf Strukturen und Klassen zutrifft.
- Eine Klasse erbt die Eigenschaften und Methoden einer anderen Klasse.
- Polymorphie verstehen: Wie ein Typ für einen anderen Typ eingesetzt werden kann und wie dies mit Vererbung zusammenhängt.
- Klassen schreiben, die demselben Protokoll entsprechen.
- Zusätzliche Funktionen zu Klassen mit Erweiterungen hinzufügen.

# Abschlussprojekt



Die Lernenden werden eine Reihe von Kodierungsübungen absolvieren, um ihr Verständnis von Swift zu testen. Es wird Übungen für Variablen, Strings, if (else-if und else)-Anweisungen und Funktionen geben.

## Projekt

### Deine eigene Konsolen-Anwendung

Suchen Sie sich ein Themengebiet heraus, was Sie besonders spannend finden und bauen in Kotlin Ihre erste eigene Konsolen-Anwendung die sämtliche Programmiergrundlagen abdeckt und Sie Ihre Programmierkenntnisse unter Beweis stellen können. Sie lernen wie man Projekte ortsunabhängig speichert und versioniert.

## Thema 01

### Git und Versionskontrolle

- Laden Sie ihr Projekt in das weit verbreitetste Versionskontrollsystem, GitHub hoch um Änderungen an Ihrem Projekt effizient zu verfolgen und zu verwalten.
- erstelle verschiedene Versionen (Branches) deines Projekts um neue Funktionen zu entwickeln oder Fehler zu beheben.
- machen Sie frühere Änderungen rückgängig bzw. stellen Sie frühere Versionen wieder her.
- Teilen Sie ihr Repository mit anderen Studenten, um die Zusammenarbeit im Team zu lernen.

# Unser bewährter Ansatz

Für den Aufbau berufsrelevanter digitaler Kompetenzen



Erfahrene Entwickler/innen

## Wir begleiten dich auf dem Weg zum App Entwickler

- Personalisiertes Projektfeedback mit Codeüberprüfung durch unser geschultes Personal.
- Jeden Monat 2x 60 Minuten Meeting mit unseren Tutor/innen.



**Marvin Gatermann**

iOS & Android Developer



**Till Hemmerich**

iOS & Android Developer



**Gordon Lucas**

iOS & Android Developer

# Vielen Dank.

[www.syntax-institut.de/selbstzahler/](http://www.syntax-institut.de/selbstzahler/)