



Einführung in die Medieninformatik

3 PRAXIS-ÜBUNG: JAVA

Beachten Sie bei der Bearbeitung die folgenden Modalitäten:

- Arbeiten Sie in Gruppen zu je 2 Studierenden!
- Abzugeben ist das vollständige Projekt inklusive aller Quelldateien als gezipptes Android Projekt. Wählen Sie dazu in Android Studio „**File > Export to Zip File ...**“. *Nur so kann gewährleistet werden, dass ihr Projekt auf anderen Systemen lauffähig ist und bewertet werden kann!*
- Die Abgabe erfolgt im OPAL als (wie oben beschrieben) ZIP-gepacktes Verzeichnis mit dem Ergebnis Ihrer Arbeit (Konvention: `ueb1_nachname1_nachname2.zip`) bis zum **09.12.2018**.

Vorbereitung: Für die Bearbeitung dieser Übung benötigen Sie ein aktuelles Java-SDK (JDK¹) sowie Android Studio² (oder IntelliJ IDEA³ mit installiertem Android SDK⁴).

Performance-Tipp: Emulieren Sie ein Gerät auf x86-Basis und mit einer möglichst geringen emulierten Bildschirmauflösung.

3.1 AUFGABE: APP-LAYOUT (1 BONUSPUNKT)

Erstellen Sie ein neues Android Projekt mit einer leeren Activity. Bearbeiten Sie nun zunächst das Layout Ihrer Activity. Darin sollen ausschließlich die folgende Komponenten enthalten sein:

- 4 x TextView
- 1 x ConstraintLayout
- 1 x LinearLayout (horizontal)
- 2 x Button
- 1 x Eingabefeld (EditText) mit dem Eingabetyp numberDecimal

Nutzen Sie die Komponenten, um das abgebildete Layout nachzubilden. Dabei sollen die Überschriften „Quersumme“ und „Fibonacci“ jeweils 32 dp vom letzten Element in y Richtung und 24 dp vom linken bzw. rechten Displayrand entfernt sein. Zusätzlich sollen Sie als Material.Headline formatiert werden. Die Elemente unterhalb der Überschriften sollen 24 dp Abstand zum vorherigen Element in y Richtung und 32 dp Abstand zum

¹ JDK: <http://www.oracle.com/technetwork/java/javase/downloads/>

² Android Studio: <https://developer.android.com/studio/index.html>

³ IntelliJ IDEA: <https://www.jetbrains.com/idea/>

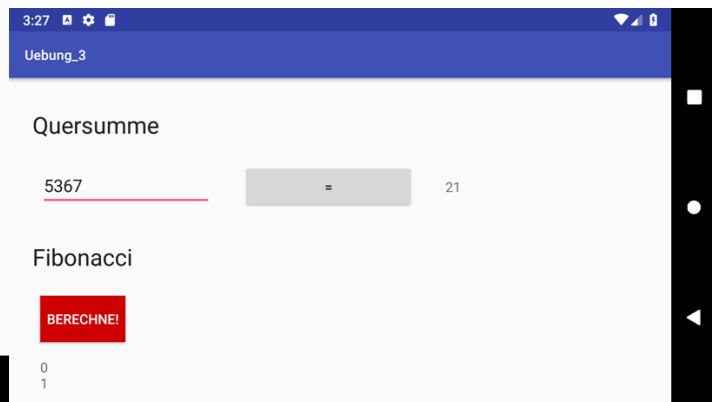
⁴ Android SDK: verfügbar über die Android Studio-Website

Rand besitzen. Die Elemente, welche zur Überschrift Quersumme gehören, sollen in einem Layout zusammengefasst werden.

Achten Sie auch darauf, dass sich die App beim Drehen des Geräts verhält, wie unten abgebildet. Dies können Sie bereits in der Design Ansicht testen.



App Design im Portrait Modus



App Desing im Landscape Modus

3.2 AUFGABE: APP FUNKTIONALITÄT (1 BONUSPUNKT)

Nun soll die App um ihre Funktionalität erweitert werden. Dabei sollen die Komponenten des in Aufgabe 3.1 erstellten Layouts verwendet werden. In der zur Activity zugehörigen Klasse sollen folgende Funktionen implementiert werden.

Teilaufgabe 1: Zunächst soll die Quersumme einer Zahl berechnet werden können. Schreiben Sie dafür eine Methode `calculateDigitSum()`, welche eine Zahl als Parameter entgegen nimmt und deren Quersumme zurück gibt (beachten Sie dazu auch die Hinweise auf den Übungsfolien). In der `onClick`-Methode des „=“ Buttons soll nun die Zahl des Eingabefeldes genutzt werden und deren Quersumme in das Ausgabefeld der App geschrieben werden. (Wie im Beispiel zu sehen ist die Quersumme von 5367 gleich 21).

Teilaufgabe 2: Im zweiten Teil sollen Sie nun eine Funktion schreiben, welche die ersten 30 Zahlen der Fibonacci-Reihe berechnet. Schreiben Sie also eine Methode `calculateFibonacciRow()`, welche eine Zeichenkette (String) zurück gibt, in der die ersten 30 Zahlen enthalten und durch Zeilenumbrüche voneinander getrennt sind. Diese Zeichenkette soll nun in der App ausgegeben werden, sobald man die Schaltfläche „Berechne!“ wählt.

Die Fibonacci-Reihe definiert sich als Zahlenreihe, wobei jede Zahl aus der Summe ihrer beiden Vorgänger gebildet wird. Einzige Ausnahme bzw. Vorgabe: Die Reihe beginnt mit den beiden Zahlen 0 und 1.

$$F_{i,i \geq 0} = \begin{cases} 0 & : i = 0 \\ 1 & : i = 1 \\ F_{i-1} + F_{i-2} & \text{sonst} \end{cases}$$

Sie können die Berechnung sowohl sequentiell, als auch rekursiv durchführen. Beachten Sie dazu die Hinweise in den Übungsfolien.