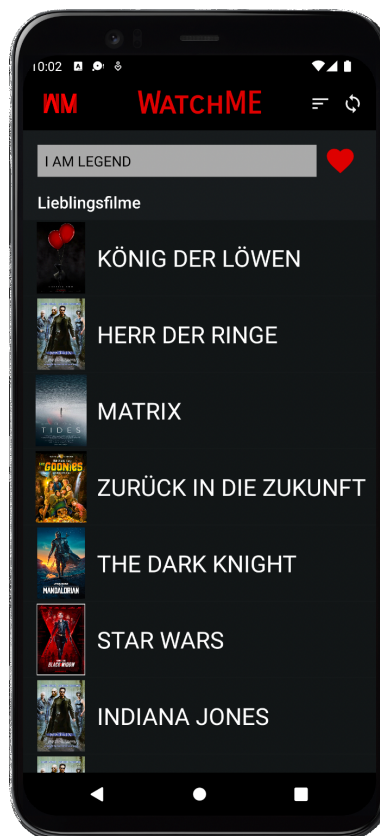


Hinweis: Zu bearbeiten ist Aufgabe 1. Aufgaben 2 und 3 sind Bonusaufgaben

1. FILME STREAMING SERVICE - Listen

In dieser Aufgabe wird die Mediathek und Favoritenliste (Sortierfunktion) einer Streaming-App programmiert.



- Öffne das Projekt "Filme Streaming Service" und die Quelltextdatei "Mediathek"

Hinweis: die Datei befindet sich unter:

`app/java/com.syntax_institut.filmstreamingservice/Mediathek.kt`

- In der Funktion `createDatabase` befindet sich bereits eine Liste aus verschiedenen Serien
 - Da die Mediathek von WatchME jedoch nur Filme enthalten soll, besteht die erste Aufgabe darin, alle Elemente aus der vorhandenen Liste zu löschen
- Die Mediathek von WatchME soll insgesamt 20 Filme beinhalten
 - Lege 20 Variablen an und weise Ihnen unterschiedliche Filmtitel zu
 - Füge alle Titel in die Liste ein

- Zum Schluss soll die Liste noch durchgemischt werden, bevor die Funktion sie wieder zurückliefert
- Im Folgenden wird die `Favoritenliste` mit verschiedenen Funktionen sortiert
 - Die Funktion `sortListAlphabetically` soll die übergebene `liste` alphabetisch von A-Z sortieren
 - Die Funktion `sortListTitleLength` soll die übergebene `liste` alphabetisch nach Wortlänge vom Kurz-Lang sortieren

Hinweis: die Listenfunktion `sortBy{}` akzeptiert die Funktion `it.length`

- Die Funktion `invertList` soll die Reihenfolge der übergebenen `liste` umkehren
- Die Funktion `deleteFromDatabase` soll den übergebenen `titel` aus der übergebenen `database` löschen
- Die Funktion `addToFavorites` soll den übergebenen `titel` in die übergebenen `favorites` hinzufügen
- Führe den Emulator aus, wähle deine Lieblingsfilme aus dem Dropdown Menü und teste die Sortierfunktionen

Viel Erfolg!



2. FINDE DEN FEHLER - Fehlersuche

Bei der Programmierung dieser APP ist etwas schiefgelaufen, begib dich auf die Fehlersuche und versuche das Problem zu lösen

- Öffne das Projekt "Finde den Fehler"
- Versuche die APP auszuführen und auf dem Emulator zu starten (Die APP wird nicht starten)
- Versuche den Fehler zu finden
- Wenn der Fehler behoben ist, führe die APP erneut auf dem Emulator aus und überprüfe, ob die APP funktioniert

Viel Erfolg!



3. DATENTYPEN SONNENSYSTEM - Datentypen (ADVANCED)

In dieser Aufgabe soll unser Sonnensystem dargestellt werden. Für jeden Planeten muss die Entfernung und der Datentyp stimmen, um ihn anzuzeigen.



- Öffne das Projekt "Datentypen Sonnensystem" und die Quelltextdatei "Entfernungen"
- Erstelle in der Funktion `distanceToEarth` 10 Variablen, in denen du die Entfernungen der Planeten in Kilometern speicherst. Wähle hierfür den kleinstmöglichen numerischen Datentyp!
 - Die Entfernungen zur Erde in richtiger Reihenfolge
 - Merkur: 77,3 Mio. km
 - Venus: 38,2 Mio. km
 - Mond: 360k km
 - Erde: 0 km
 - ISS: 408 km
 - Mars: 55,7 Mio. km
 - Jupiter: 588,5 Mio. km
 - Saturn: 1,1955 Mrd. km
 - Uranus: 2,5189 Mrd. km
 - Neptun: 4,3059 Mrd. km
- Erstelle eine veränderliche Liste und befülle sie (in der oben angegebenen Reihenfolge) mit den Entfernungen
- ersetze `mutableListOf()` in der `return` Anweisung mit deiner Liste
- Die Funktion `distanceToSunAE` erhält eine Liste `distances`, gefüllt mit den Entfernungen der 10 Objekte, nur dieses Mal die Entfernungen zur Sonne. Die Liste enthält verschiedene numerische Datentypen. Die Aufgabe der Funktion ist es, die Entfernungen in AE

(Astronomische Einheit) umzurechnen und als Float, gerundet auf zwei Nachkommastellen in einer Liste zurückzuliefern

- Eine Astronomische Einheit (1 AE) entspricht der Entfernung von der Sonne zur Erde und beträgt genau: 149.597.870,7 km
- Rechne jede Entfernung der übergebenen Liste von km in AE um, wandle die Entfernungen dafür in einen geeigneten Datentyp um
- Runde das Ergebnis auf zwei Nachkommastellen

Hinweis: Dafür kannst du folgenden Trick benutzen:

Multipliziere die Kommazahl mal 100 und wandle sie in eine Ganzzahl um (die restlichen Nachkommastellen werden weggerundet). Wandle die ganze Zahl wieder in eine Kommazahl um und teile wieder durch 100.

Du erhältst eine auf zwei Nachkommastellen gerundete Zahl

- Füge alle deine Ergebnisse in eine veränderliche Float Liste ein
 - ersetze `mutableListOf()` in der `return` Anweisung mit deiner Liste
- Führe den Emulator aus. Wenn alles stimmt, siehst du unsere 8 Planeten, über der Erde den Mond und unter der Erde die ISS

Viel Erfolg!

