Worksheet 4: Local Data

Updated: 10th September, 2018

In this worksheet, you'll take a simple list-based app and implement *persistence*, in the form of an SQLite database.

> **Note:** The games from both worksheets 2 and 3 also provide good opportunities to learn these skills.

1 Setting Up

Obtain a copy of "`alliesandenemies.zip`" from Blackboard. This contains the Java and XML files for an existing app (that lacks persistence). Start a new project in Android Studio, and copy the contents of the zip file into the equivalent locations in your project directory.

As in worksheet 3, you'll also probably need to add the following build dependencies:

- "com.android.support.constraint:**constraint-layout**".
- "com.android.support:**recyclerview**".

Once again, you can either edit `app/build.gradle` directly, or use the IDE's menu options. To do the latter, select "File" → "Project Structure", then select the "app" module (or whatever module name is shown), and then the "Dependencies" tab. You should see the lists of dependencies. If the required dependencies are not there, you can select "+", and then "Library dependency", and then choose from among the options shown.

2 Adding a Database

The existing app displays a list where entries can be added, edited and deleted. Currently all data is simply held in memory. Your task is to ensure it is also written to a database, so that it can be re-loaded on start-up. You can largely just apply and adapt the information from the lecture notes.

We only need a single database table for our purposes, and the contents of the `Faction` class will tell you what needs to be stored in it. However, to actually perform database interaction, you will need to add code to strategic points in the `FactionList` class (as hinted in the comments).

When testing your work, you will need to run the app *multiple* times, to ensure that each add, edit and delete operation is reflected the next time the list is re-loaded and displayed.

End of Worksheet