# Worksheet 6: Local Data

Total: 10 Marks, Due: 21 Sep, 2022 13.30 AWST

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

## 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.

You may need to add the following build dependencies:

- "**androidx.constraintlayout:constraintlayout:1.1.3**".
- "**androidx.recyclerview:recyclerview:1.2.0-alpha05**".

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).

## 3 Testing [4+3+3 = 10 Marks]

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.

**Warning:** Don't forget to regularly test your app's ability to create and initialise its database. To do this, you must delete the database file, which can be done by wiping the virtual device. Open "Tools" → "AVD Manager", and you'll find the "Wipe" option in a drop-down menu for each virtual device.

If you don't do this every now and then, you might end up creating a bug that prevents database initialisation, and not being able to detect it. Your app may still appear to work fine if it finds an existing database from a previous version of the app, but it will fail if run on any other device.

**End of Worksheet**