# Practical 8.1 - External Data Storage 1

1. Build the Language Preference app from the lecture. Extend the app to allow the user to select both a language *and* a text colour for the greeting. Provide radio buttons (or other user controls) to allow the user to specify his or her preferences. Be sure to test that your app retains the preference settings between launches.
2. Following the technique described in lecture, build an app which loads a ListView or a Spinner from an external text file. If you use a ListView, make sure you have enough entries in your text file to be able to verify correct scrolling behaviour. Be sure to implement correct exception handling (i.e. try-catch for IOException). In your catch block, you can access the Message property of the exception object via its getMessage() method. If you need to review exception handling in Java, see docs.oracle.com/javase/tutorial/essential/exceptions/handling.html.
3. Build an app that uses an SQLite database to provide simple search functionality. The database should have a single table which has two data fields (e.g. in the lecture, we had city name and country name). The user inputs a value for one field, and you display *the other field* for all matching records. Continuing the city/country example, the user would enter a country name, and the app would display **all** cities in that country. You may use whatever data you like, with the following requirements:
   1. Your app should have correct modularity. Have appropriate methods (or classes) to manage the creation of your database, the creation and population of your database table, and the processing of queries against your database.
   2. The user should not have to type in the search key (typing is always bad, but on mobile it is really bad). Present an appropriate control for selecting the search key from the available values in the database.
   3. The search selection control described above should be populated dynamically from the database when the app launches. In my example, if I added more records to my database and some of the added cities were from a country not yet represented, that country would show up in the search selection control the next time the app was run.
   4. Populate your database with sufficient seed data that at least one search key value will return multiple results. Check that this query works and displays correctly.