

CS 477 Programming Assignment #2: SQLite, ListViews

Due date on blackboard

The primary purpose of this assignment is to have you create and use a SQLite database and ListViews (or RecyclerViews). This document contains information about the functionality that your app needs to include. Your version can have a different look and feel but needs to serve the same purpose and demonstrate the basic skills. Because I get so many requests to do a workout app as a course project, everyone gets to write one.

The app should be fairly simple but usable. Since the purpose is to keep information about what exercises you need to do and information about these exercises (weight, reps, ...), it is that kind of information you will need to keep in a database. Getting this running is a little more complicated than it sounds but it is still mostly database work.

The user of your app needs to be able to do two primary things: 1) create a database of exercises (setup/edit button) and 2) use this database to create and run a workout (Do a workout button). The workout itself does not need to be saved (but it could be and then re-used).

I have included some screen shots from a version of my project 2.

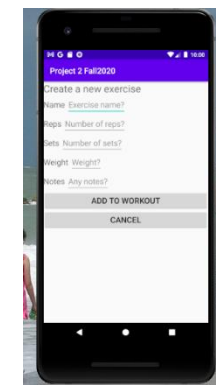
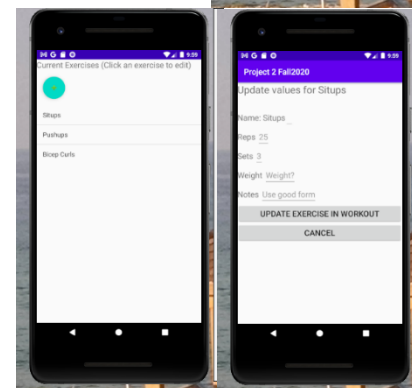
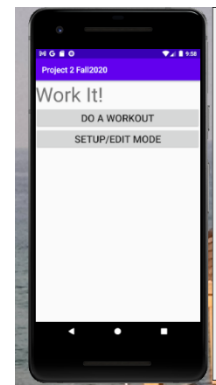
Part 1: Workout database (50%)

You will be using a database to save exercises created by the user. Use a listview

(or recyclerview) to display the database. Be sure that the first time a user (i.e. the GTA) opens the app, there are at least three starting exercises there. My app just lists the exercises on this screen. If the user touches an exercise, they can see the details (sets, reps, weight, ...) and they can also edit any of these features except for the exercise name here. I have a cancel button but be aware that the user might also use the back button and that must be implemented as well.

The user should be able to add and delete exercises. For deletion, the easiest way is a long click with an alert dialog. However, you do it, be sure it isn't easy to accidentally delete an exercise. When an exercise is added, be sure it has a name – you should not be able to create exercises with duplicate names.

Test this completely before you worry about workouts. Be sure that your app does not crash even when unexpected things happen.



Part 2: Do a workout (50%)

The user can now use this database to create a workout. There needs to be a way for a user to select a subset of the database as their workout. There are several ways you might consider doing this

- The easiest way would be to list all of the possible exercises and let the user delete from the list any exercises they don't want in their workout. Once they have done this, the remaining list is their workout and they can delete each exercise when they complete it. NOTE: deleting from a workout should NOT make any changes to the database!
- A better way might be to let them use checkboxes to choose and then have a button that goes to a mode which lists the remaining exercises and provides a way to delete or mark each exercise they finish.
- There are probably lots of other ways. Think about how you would want to create something that helped you keep track of things. However, be sure your app is easy to use for a first time user.

Final Notes

- Remember to put at least 3 different starting exercises in your database to make grading of the assignment easier. This should be done in the onCreate() in the database helper (not the main activity's onCreate()).
- Whenever possible, limit the user's input – for example, the number of sets will always be an integer. Limit the keyboard so that you don't have to error check this.
- It is often a good idea to use a 'hint' in an edit text rather than 'text' if you expect that the user will be changing that value.

You are free to implement the user interface in any way you want but if the TA or I have to spend time trying to figure out how to do something, you will most likely lose points. I will demo a version in class - yours should be at least as fancy ☺

As with project 1, submit your Java and XML code on Blackboard along with your .apk file. (BE SURE TO FOLLOW THE NAMING CONVENTION FOR THE APPLICATION AND THE APK – PROJECT1-userid – AND USE BUILD APK).