AnimoSpace Support

Non-graded Exercise: Practicing with SQLite

0 Points Possible

Attempt 1

7/10/2025

NEXT UP: Review Feedback

Attempt 1 Score:
N/A

Add Comment

Unlimited Attempts Allowed

Details

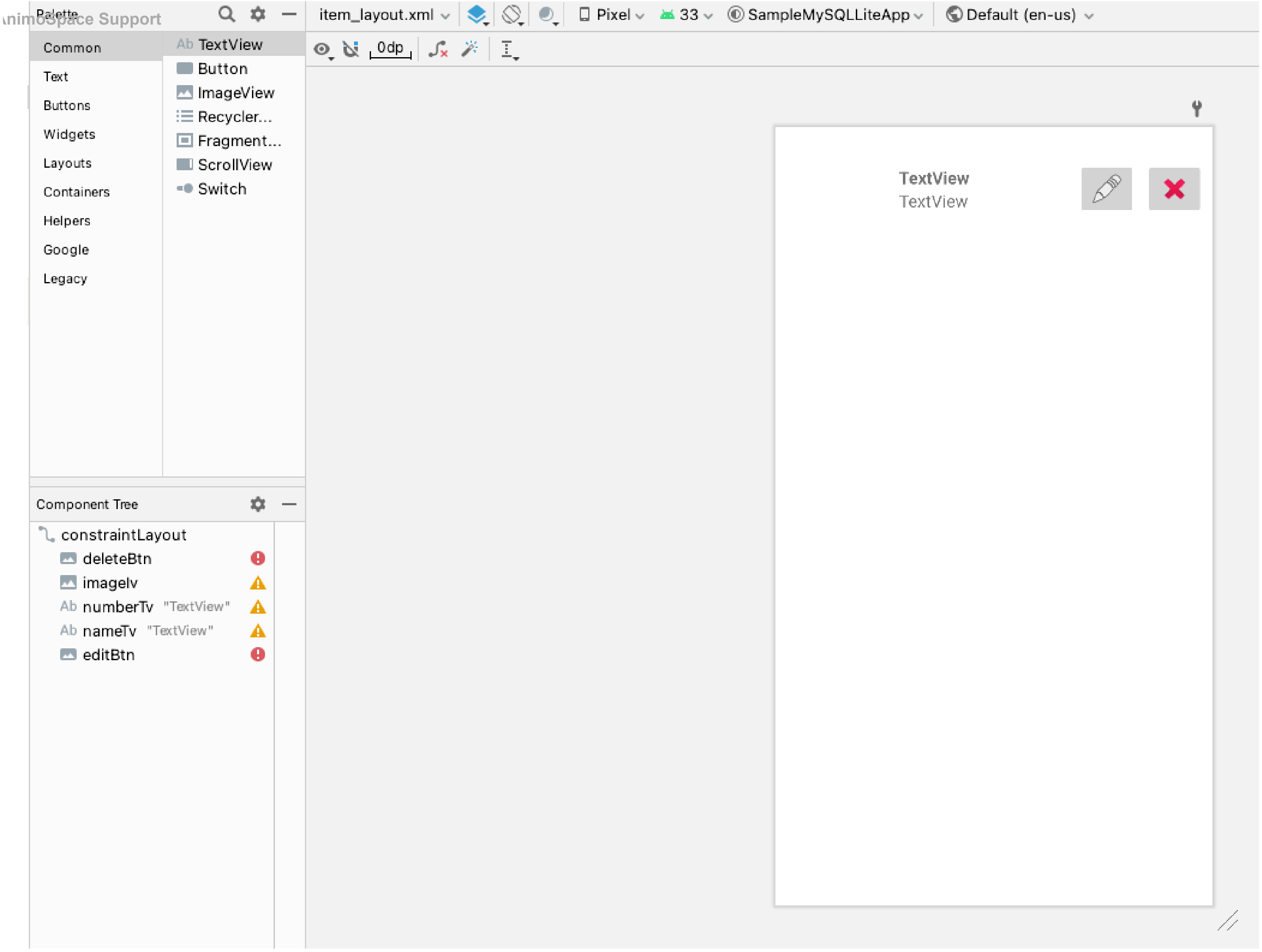
Description

This is a non-required exercise meant for practicing knowledge and getting more familiar with topics in **Module 4.a: Storage (Local)** (https://dlsu.instructure.com/courses/214805/modules/973790) .

Problem

The previous implementation of the Contacts app we discussed in class is limited to the add operation. For this problem, we'll want to extend the current implementation to accommodate **deleting** and **editing** of contact information. In order to do so, we'll want to modify the **add** operation as well.

Before anything else, a few changes have been made to the application to accommodate the exercise. This mainly revolves around the new Views found in the item_layout.xml. Each contact item now looks like the following:



Notice the buttons for adding and editing. This isn't the cleanest approach to adding these operations, but it is a simple approach for our exercise. There are also other changes that have been integrated into the original project, so in order for us to be on the same page, please refer to the following templates:

- Java: Practice-MySQLiteContacsApp (Java).zip (https://dlsu.instructure.com/courses/214805/files/26095786?wrap=1) (https://dlsu.instructure.com/courses/214805/files/26095786/download?download_frd=1)
- Kotlin: Practice-MySQLiteContacsApp (Kotlin).zip (https://dlsu.instructure.com/courses/214805/files/26095703?wrap=1) (https://dlsu.instructure.com/courses/214805/files/26095703/download?download_frd=1)

While you don't have to perform the tasks in the order specified below, it would be best to start with modifying the add operations since we'll need to ensure that all our contacts have IDs as we can't edit and delete without certainty without the ID. Here are the tasks:

- For the **add**, kindly modify the code so that the ID of the inserted Contact is added to the Contact found in the ArrayList
  - In the current implementation, the info is sent back from the AddContactActivity, but the ID is not included.
  - Modifications most likely would be in the AddContactActivity and in the DBHelperClass.
  - As a hint, make use of the return value of `database.insert()` as this returns the inserted value's primary key.
- For the **delete**, to make things simple, we've added a delete button for each of the ViewHolders. Clicking the delete icon should delete both the icon in the RecyclerView and in the DB
  - Modifications most likely would be in the MainActivity and in the MyAdapter.
  - Don't forget to use an ExecutorService to properly offload the delete operation from the main thread.
- For the **edit**, we repurposed the AddContactActivity to also accommodate editing.

nimoSpace Support

The logic here is that we send the ViewHolder data to the AddContactActivity. If it recognizes that an edit is in place, then the activity adjusts accordingly.

- Your task here is to implement the edit operation within the AddContactActivity without messing with the add operation.

Instructions

1. If you want to submit something, please feel free to submit your project.
2. If you'd like to jump to a possible solution, you'll eventually find it in the module
   - Send me any questions as the solutions aren't exactly straight forward.

Preview Unavailable

Clemente_SampleMySQLiteApp.zip

Download

(https://dlsu.instructure.com/files/26887814/download?download_frd=1&verifier=8ycHYH4Wfvi3tBzWMoGZ9AkGuVoo0IULmqvF7BkR)

(https://dlsu.instructure.com/courses/214805/modules/items/5850539)

N Attempt

(https://dlsu.instructure.com/courses/214805/modules/items/5