CSCI 4176 & 5708 – Fall 2022

Lab Assignment 03 – Due 30/10, 11pm

Please start this assignment early; programming and logic take time - if you leave it to the last minute, you might not have enough time to finish or might make silly mistakes that you otherwise could avoid. Note that TAs and Instructors will not be able to answer last-minute questions!

All work is to be handed in on GitLab, main branch; other branches will be ignored.

Your repo is located at: https://git.cs.dal.ca/courses/2022-fall/csci4176_5708/assignment-3/<csid>

The repo contains only a README.md

The goal of this assignment is to get you familiar with the basics of the Recycler View. This Assignment depends on on A2! You will use Fragments, Navigation and a way to pass information between fragments.

An excellent resource for Kotlin is their docs: https://kotlinlang.org/docs/home.html

This assignment is not meant to be technically challenging but you will have to do some reading on the docs. **There is more than one way to solve this assignment**.

Your solution must be in Kotlin

Your code **must compile**. If it does not compile, you will receive a 0 (zero), and no partial marks will be given.

Coding Style: You must have proper names and comments. Please follow the guidelines on https://kotlinlang.org/docs/coding-conventions.html#naming-rules

Public methods and classes should be documented. Please note that different than Java, Kotling does not encourage the use of @params, @return tags: https://kotlinlang.org/docs/coding-conventions.html#documentation-comments

Git Rules for A3: You are always expected to have a README.md file with your name, banner, and any other information you might like to add (such as references); you can use the one from A1. You should also include a .gitignore. Your submission **must include the entire project minus the ignored files/folders**

The Assignment

A3 is about Single Activity applications with fragments, navigation **plus a RecyclerView**. Your goal is to create an app that can store and display notes using the **RecyclerView**. Notes have a *title* and a *body*. The body can be multi-line, but the title <u>cannot</u>.
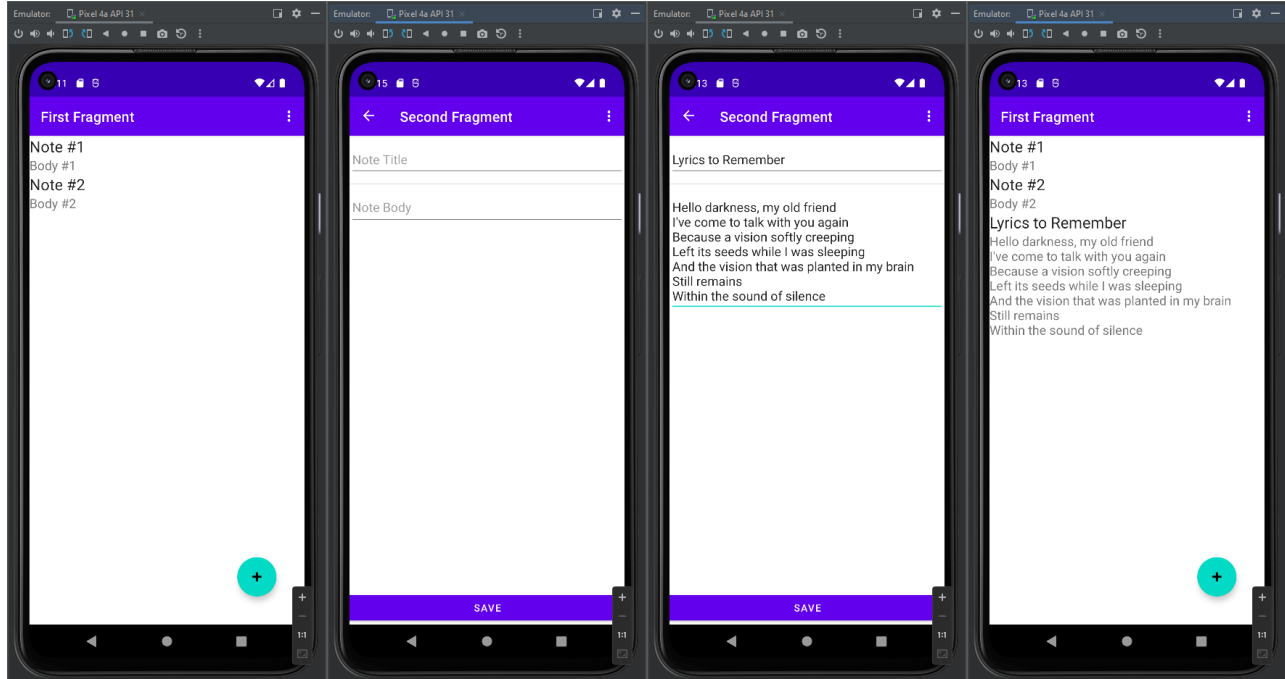
The first fragment (home/start) contains the recycler view (https://developer.android.com/develop/ui/views/layout/recyclerview) and a floating action button to add new notes. Once pressed, the application must switch to the second fragment, where the user can add new notes. See: https://developer.android.com/guide/navigation/

The second fragment will have a text field with *hints* indicating where to input the title and the body. You do not have to strictly follow the design above. The fragment also contains a button to save the notes. <u>If the note has an empty title or an empty body, the application should not save the new note</u>**;** <u>instead, it should return to the first fragment</u>. The user should be able to add as many notes as they want. The FAB button should only be visible on the first fragment.

Hint: The CardView class helps a lot in defining the Note View: https://developer.android.com/develop/ui/views/layout/cardview

From A2: There are multiple ways to pass information between fragments: Bundles, Companion Objects, ViewModels, SafeArgs, etc. Please take some time to investigate the different methods and their advantages/disadvantages. Some examples: https://developer.android.com/guide/navigation/navigation-pass-data

I've completed A3, what's next? You will be using recycler views a lot because they are versatile (through its adapters). How about you take some time to implement a *onClick* that shows the note details in a third fragment and allows the user to edit or delete the note?