Matt Anderson U00628270

https://github.com/MattAndersonCEG3900

Laptop: Lenovo ThinkPad T400, 4 GB RAM, Intel Core 2 Duo

Phone: Samsung Galaxy S5 running Android 6.0.1

Task 1:

- Source: Pilot dropbox
- Status report: Page 2
- Summary: Page 2

Task 2:

- Screenshots: Page 7
- Source: Pilot dropbox
- Status report: Page 8
- Experience report: Page 8

Task 3:

- Source: Pilot dropbox
- Status report: Page 8
- Experience report: Page 8

Task 4:

- Screenshots: Blocked
- Source: Blocked
- Status report: Page 9
- Experience report: Page 9

Task 5:

- Source/Examples: Blocked
- Status report: Page 9
- Experience report: Page 9

Task 1: Java-8 Streams in Linux or Windows (2 hours)

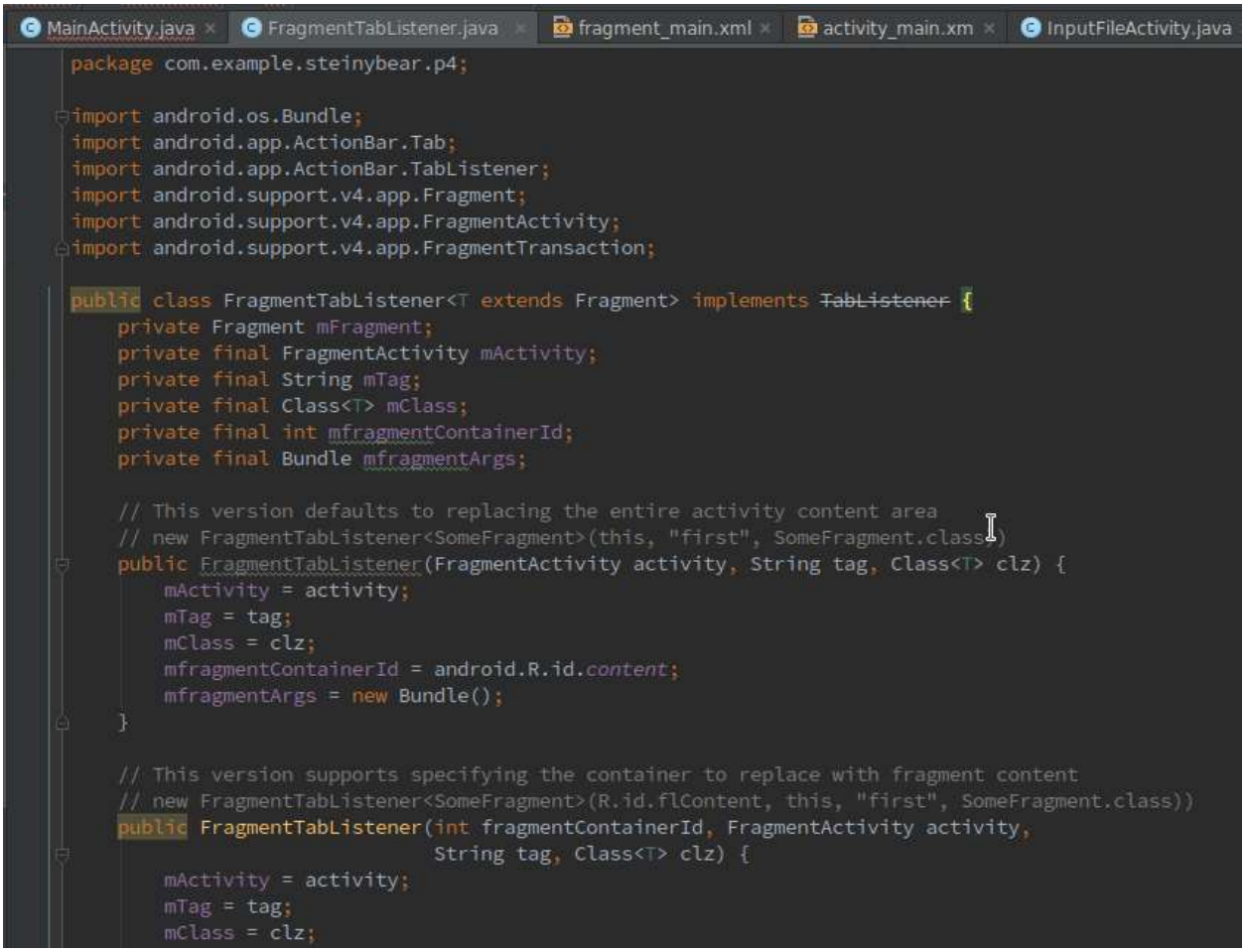
Status:

This program is working fine. I have not experienced any hangs, bugs, or crashes with it in its current state.

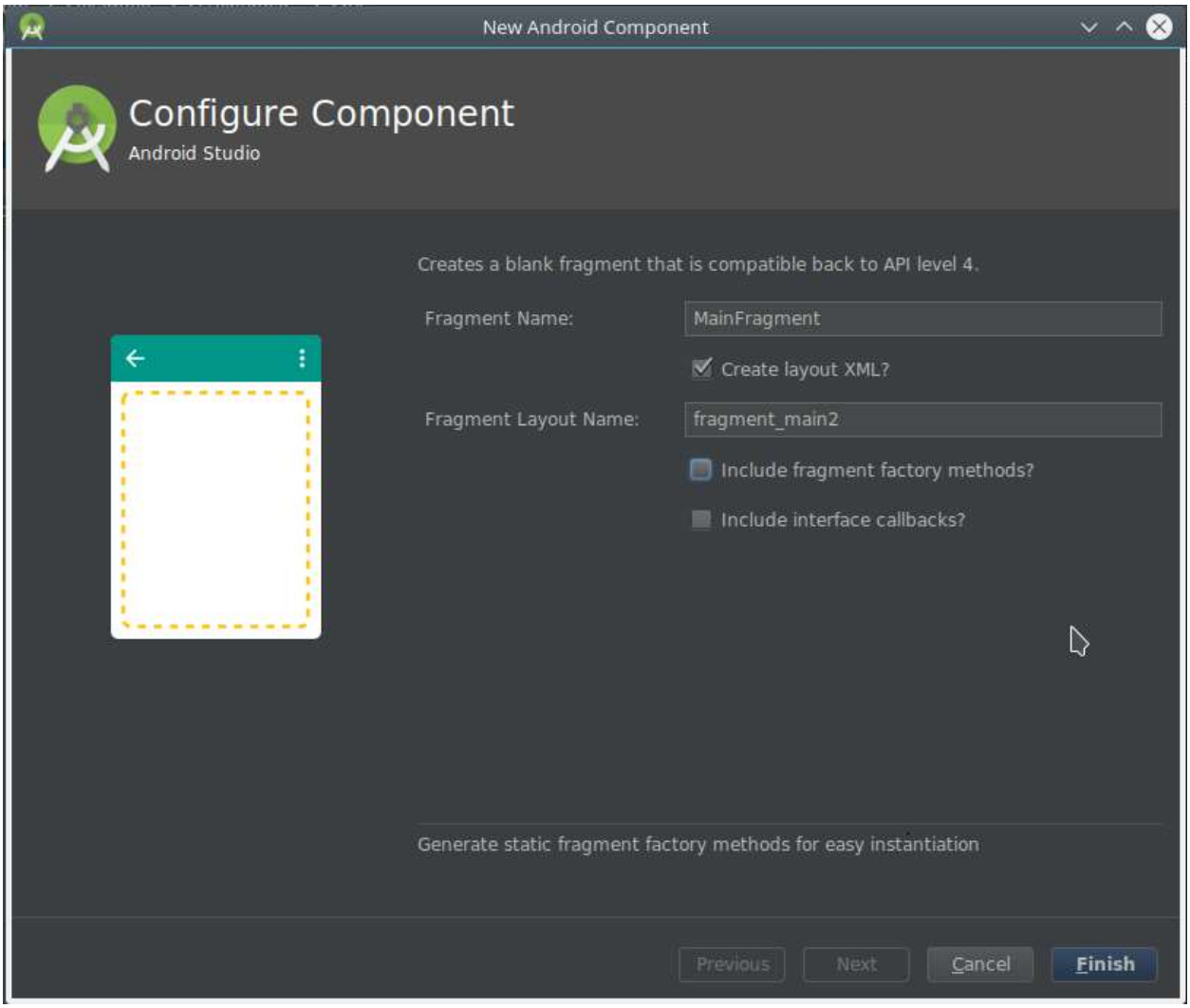
Summary:

I developed this program using NetBeans and Java 8. I figured out the syntax by doing a couple hours of research into mapping with Java 8 as I still do not feel fluent in using them. Once I got the basic syntax of maps down, this part of the lab was pretty straight forward. From my experience, this part of the lab did not require much clarification as it only takes a very fundamental understanding of maps to accomplish.

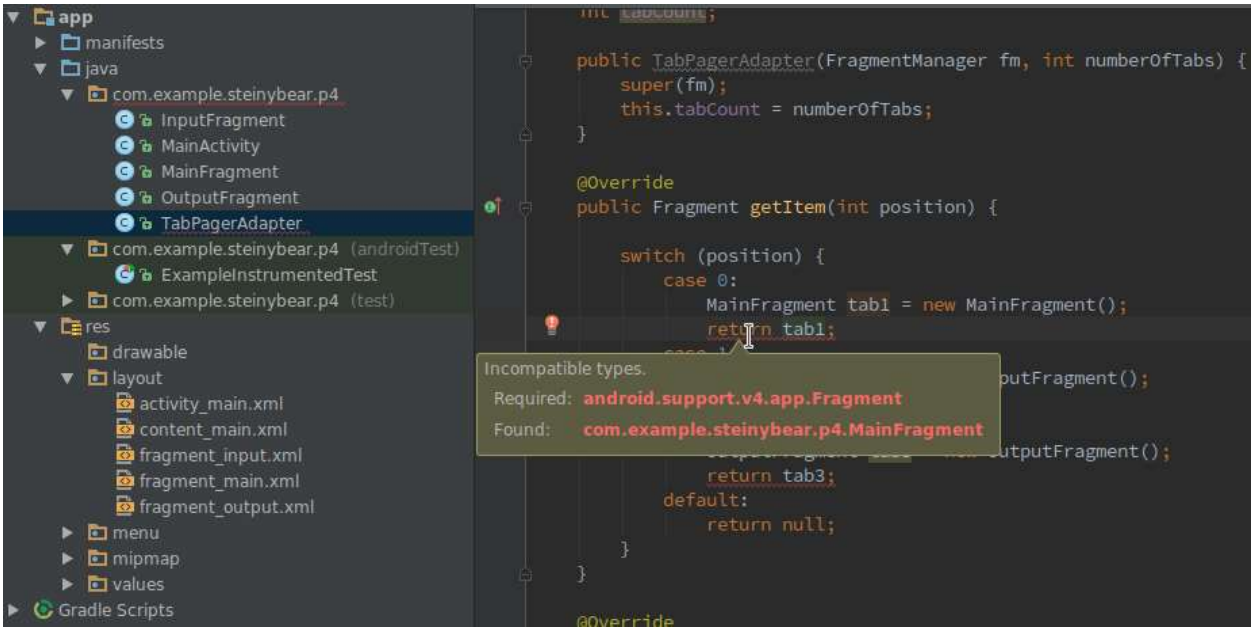
Task 2: Java-8 Streams in an Android APK (4 hours)



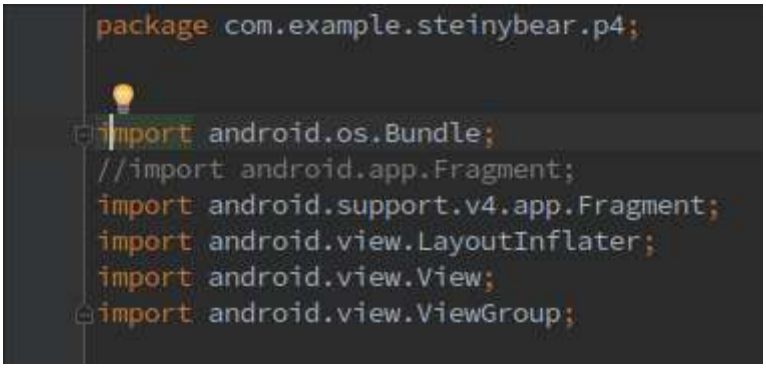
I first attempted to set up the described UI by using a fragment tab listener and adding fragment activities, but this led to a chain of errors that I was not able to find a solution to. I elected to try another solution.



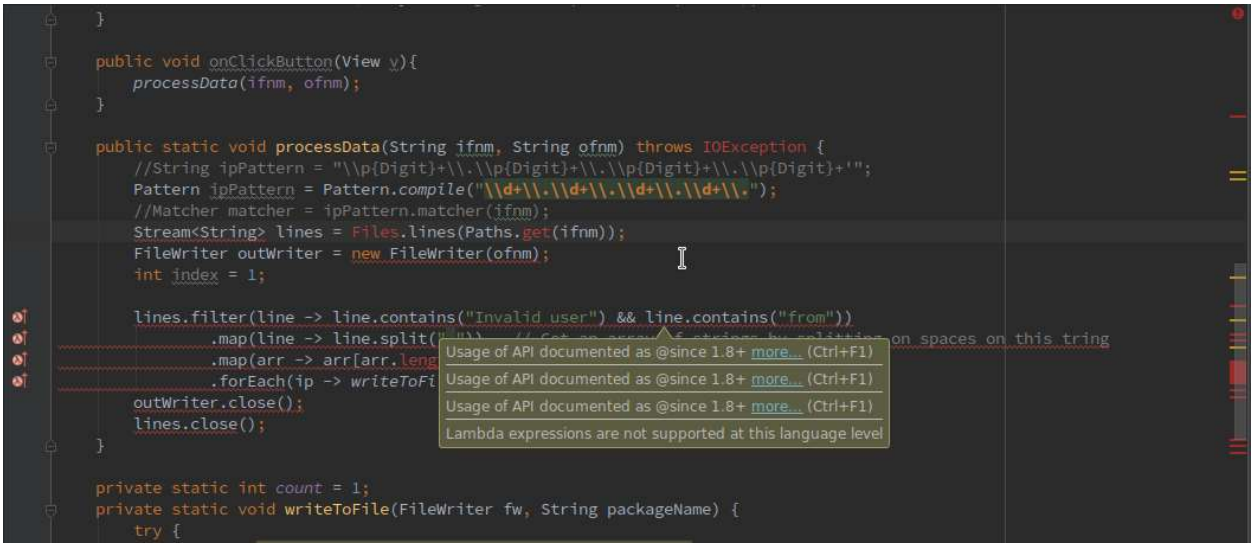
This time I simply used the main activity with a fragment activity for each tab I wanted on the screen.



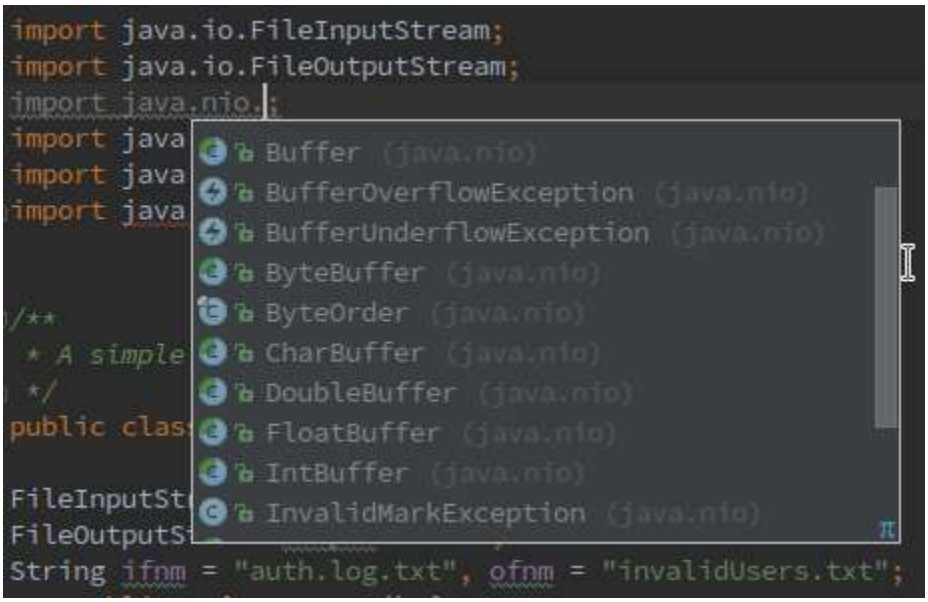
I was having some trouble with the TabPagerAdapter that I implemented in that each fragment (MainFragment, InputFragment, OutputFragment) extended the Fragment class, but the IDE still did not interpret these objects as Fragments. As shown in the screenshot above, it was expecting a v4.app.Fragment.



After a fair amount of research, I discovered that there were in fact multiple Fragment libraries and the one that was automatically imported by AndroidStudio was incorrect. I changed the import statement and the tabs began working fine.

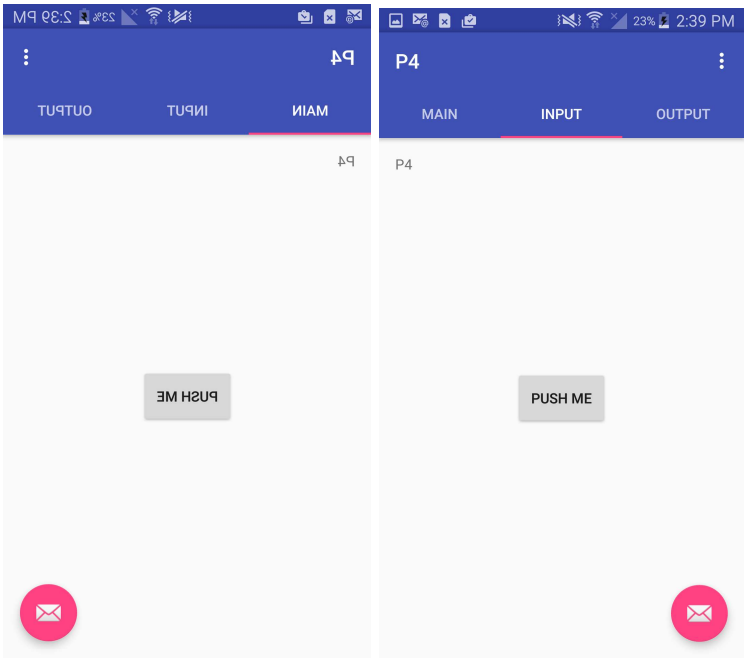


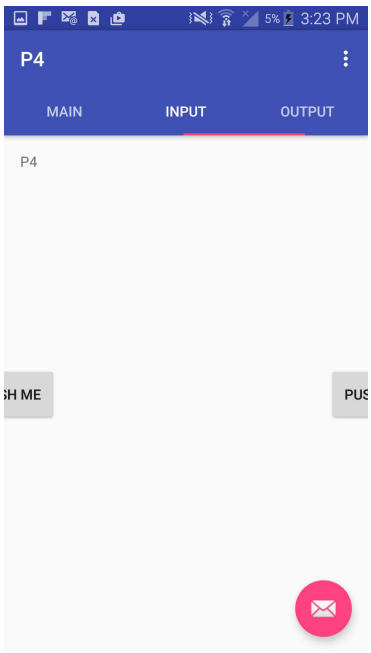
The code from part 1 didn't end up being compatible with Android. I spent the rest of the time on this section trying to find a way to use Streams, Map, and Files in a way that works with Android.



By hovering over various red code, I was able to determine what libraries it was attempting to use. When I went to import those libraries, they didn't exist. For example, java.nio.File.

APK Screenshots:





I was able to successfully create a GUI that could be scrolled through horizontally, but wasn't able to set up the input and output GUI fully since I was still unable to parse in the main activity.

Status report:

The UI of the APK is mostly set up but libraries for Android files, streams, and maps need to be discovered and understood before the program can supply any meaningful information to the UI tabs.

Experience report:

This was the beginning of the frustrations with this project. At the very least, I learned exactly how different Android can be from “standard” Java in that many libraries in Android vary or at least don’t exist as they do in Java.

Task 3: Map-Reduce in Linux/Windows (2 hours)

Status report:

I am able to map from multiple URLs and parse the appropriate lines of text from HTML, but I was not able to learn how to reduce them based on word count in time. The program currently contains functional mapping and an in-progress implementation of reduction.

Experience report:

All of the resources I could find (in lecture notes and across the web) did not help me understand what exactly it is I’m doing with this code. I feel that I need at least one dedicated class period with examples and explanations for what is happening if I’m going to be able to fully understand and utilize these functions.

Task 4: Map-Reduce in Android

Status report:

Blocked. This task is not completable without a functioning Task 2 and Task 3, and will also utilize some of the same libraries and code that I was not able to find/get working in Task 1.

Experience report:

Very difficult. I could not get the simpler component pieces of this task working in previous tasks.

Task 5: Java8 Concurrency

Status report:

Blocked. This task is not completable without a functioning Task 2, and will also utilize some of the same libraries and code that I was not able to find/get working in Task 1.

Experience report:

Very difficult. I could not get the simpler component pieces of this task working in previous tasks.

Total Time: About 8 hours (9 including making the report)

P4 Notes:

These projects could use a smaller scope or greater amount of time to complete. Since P1, I have not been able to find the time to fully complete one of these projects (despite spending up to 22 hours on them).