JavaFx Instructions

To ensure compatibility, I would like everyone to be using Java 15 for this class. You can get the latest version of Java here:

https://www.oracle.com/java/technologies/javase-downloads.html

You can tell what version of java you are using by typing the following at the command line:

```
javac -version
```

Java 15 gives you a version that starts with 15.0

JavaFx is a framework for making graphical user interfaces in Java. For many years, JavaFx was bundled as a standard part of the Java Development Kit. However, in Java 11 and higher, JavaFx has been moved to be a separate stand-alone module. This means that if you download the standard edition of Java 15 (i.e., Java 15 SE) you will not have access to JavaFx.

Install Standalone JavaFx

Another option is to download a Standalone module for JavaFx. The challenge with this approach is that you need to have a way for Java to find this module and make use of it when you compile and run your program.

You can download JavaFx at:
https://gluonhq.com/products/javafx/

You probably want to download the SDK
[[I haven't been able to get the jmods to work … but I don't have much experience with jmods]]

You need to take the javafx folder that you downloaded and put it somewhere that is easy to find. In particular, you will need to give Javac and Java the location of this javafx folder in order to make use of the javafx module.

I took the javafx-sdk folder and moved it to C:\ (my root folder) and then I renamed it to javafx. Therefore, the location of the javafx lib folder for me is: C:\javafx\lib

You can use any location on your machine that you want, but you will need that location to compile or run Java programs.

In order to use a module to compile, I need to give some command-line arguments (options) to the Java compiler. The way that you give command line arguments (options) to the Java compiler will depend on what development environment you are using. (Chief Nelson the TA or myself will likely be able to help you.)

To compile:

```
javac --module-path  C:\javafx\lib --add-modules javafx.controls    TestApplication.java
```

To run:

```
java --module-path  C:\javafx\lib --add-modules javafx.controls    TestApplication
```

This tells Java that I can find new modules at C:\javafx\lib and that there is a particular module that I want to add when compiling/running this program which is called javafx.controls

Note that -p can be used in place of --module-path (they mean the same thing)

```
javac -p  C:\javafx\lib --add-modules javafx.controls    TestApplication.java
```