COMP 210: Object-Oriented Programming

SETTING UP JAVAFX 11

Robert Utterback

Abstract

The perfect storm of Java version, JavaFX version, and how Ubuntu manages package installation makes setting up JavaFX a little difficult and tedious. Last year it was easier, and next year it will be easier, but for now it takes a little work.

1 Add a User Library (one-time)

You'll only need to do this one time, no matter how many JavaFX applications you create.

1. Open up `Eclipse -> Window -> Preferences -> Java -> Build Path -> User Libraries`.

2. At this screen click the `New` button.

3. Give it a name like 'JavaFX11'.

4. Click the 'Add External Jars' button.

5. Navigate to `/usr/lib/java/javafx11-manual`. You may need to start by clicking 'Other Locations' in the left columns.

6. Add all the `.jar` files in this directory. (Click the top one, then hold shift and click the bottom one. Then click 'OK'.)

2 Project Setup (for every project)

You'll unfortunately need to perform these steps for each project you setup, unless one of you can find an easieir way.

1. Start the 'New Java Project' dialog as usual.

2. After clicking the 'Next' button for the first time, click on the 'Libraries' tab. Highlight 'Modulepath' in the main window and then click the 'Add Library...' button. (This is where you normally add `JUnit`, which you should still do).

3. Highlight 'User Library' and click 'Next'.

4. Select the JavaFX11 user library you created above and click 'Finish'.

5. Continue with project creation as usual.

6. Right-click your project in the Package Explorer, and choose `Run As -> Run Configurations...`

7. Right-click 'Java Application' in the left column and choose 'New Configuation'.

8. Give it a name (e.g., same as your project name)

9. Click the 'Arguments' tab.

10. In the 'VM Arguments' text box, type the following line:

```
--module-path /usr/lib/java/javafx11-manual --add-modules=javafx.controls
-Dprism.order=sw
```

11. Back in the 'Main' tab, type in 'Main class' box the name of whatever your initial class will be, e.g., the one that will inherit from JavaFX's `Application` class.

12. Below that box, tick the next two checkmarks. One is about 'system libraries' and the other 'inherited mains'.

Whew! Now you should be setup. To be sure, add a new class that inherits from 'Application' and override the 'start' method to do something simple, like display a simple button. Let me know if you can't get this working.