Instructions for Getting Started with Iteration 2

Step 1: Prepare Eclipse for JavaFX GUI programming

1. Look at the procedure outlined here: <http://www.cs.umd.edu/eclipse/javafx.html>
2. Download JavaFX 13 from the link in the first step above. You will want the JavaFX 13.0.2 SDK for Mac OS X. This will download a zip file, and probably unpack it is your Downloads folder. It needs to be installed somewhere. I suggest to put it where you put the jar files that we downloaded before for JUnit testing.
3. In your home directory, there is a file called .bash\_profile that needs to be edited (with nano or something like that). Find the line in there that defines the $JUNIT\_HOME environment variable. Immediately after that line, add the following line:

export $PATH\_TO\_FX = $JUNIT\_HOME/javafx-sdk-13.0.2/lib

1. Save this file, and then open a NEW terminal window, to check that this new variable is now defined:

echo $PATH\_TO\_FX

1. Check out a new branch for Iteration 2

git checkout -b olivia\_iteration2

1. Stop and restart Eclipse, and make sure that your project is pointing to this new branch.

Step 2: Set up a new package directory structure for Iteration 2

1. First create two new packages within your polling project (there should already be one there called ‘application’ where your TextApplication codes from iteration 1 are located).

With your polling project highlighted in the project explorer window, right-click, and then choose New -> Package

Create a package called ‘model’, and then create another package called ‘view’.

You should see these new packages show up in parallel with the application package in the project explorer window.

1. Now, drag and drop ALL of the codes from your TextApplication from the application package into the model package … Eclipse will ask you some questions, and just choose the defaults … the point here is that Eclipse will edit these files on the fly in order to update them as necessary to be part of this new package.
2. At this point, you should have nothing in the application package, all of your files from Iteration 1 in the model package, and nothing in the view package.

Step 3: Add and edit the new GUI skeleton classes

1. Put the two new codes (PollTrackerApp.java and PollTrackerController.java) into the application package directory.
2. There are a couple of minor modifications that need to be made to PollTrackerApp.java in order to get things to work from the outset:
   1. At line 22, change this line so that it points to the correct location of the planned FXML file … i.e. change “src/view/” to “./view”
   2. At line 101, there are a couple of lines that can be used to define an initial set of polls. Uncomment the first of these (the one that creates a random poll list), and commend out the second one (that creates an empty set of polls).
   3. You will need to add a couple of lines just before this, to define a list of party names:

String[] nameList= {"Liberal","NDP","Green","CPC","BQ","Rhinoceros","PPC","Olivia"};

factory.setPartyNames(nameList);

1. Save this new version of the file.

Note: You should be seeing a lot of errors in PollTrackerApp.java at this point, because Eclipse still does not know where the JavaFX libraries are located!!!!

Step 4: Creating a Build Path

1. In the project explorer window, with the application package highlighted, right click and choose:

Build Path -> Configure Build Path

1. Click on the Libraries tab, and then highlight Classpath, and then click on ‘Add External JARs’
2. Navigate to where you installed the JavaFX 13.0.2 libraries in step 1 above, and then add ALL of the jar files in the lib directory. There should be eight of them in total. You may have to add them one at a time … each time going back to highlighting Classpath and then clicking Add External JARs.
3. Click on Apply and Close. Once you do this, you should see all of the errors in PollTrackerApp.java go away!!!

Step 5: Creating a Run Configuration for running the application

1. The last step is to tell Eclipse where it can find the JavaFX libraries that you downloaded and installed in Step 1
2. With the application package highlighted in the project explorer window, right-click and choose Run As -> Run Configurations
3. In the window that comes up, click on the Arguments tab, and add the following in BOTH the Program arguments and the VM arguments sections:

--module-path /Users/brash/Dropbox/java/javafx-sdk-13.0.2/lib --add-modules javafx.base,javafx.fxml,javafx.graphics,javafx.controls

where you clearly have to replace the directory structure shown here with wherever you installed the JavaFX libraries in step 1 above.

**IMPORTANT!!!! Uncheck the option "Use the -XstartOnFirstThread argument when launching with SWT"**

1. Click Apply and then close this window. }

});