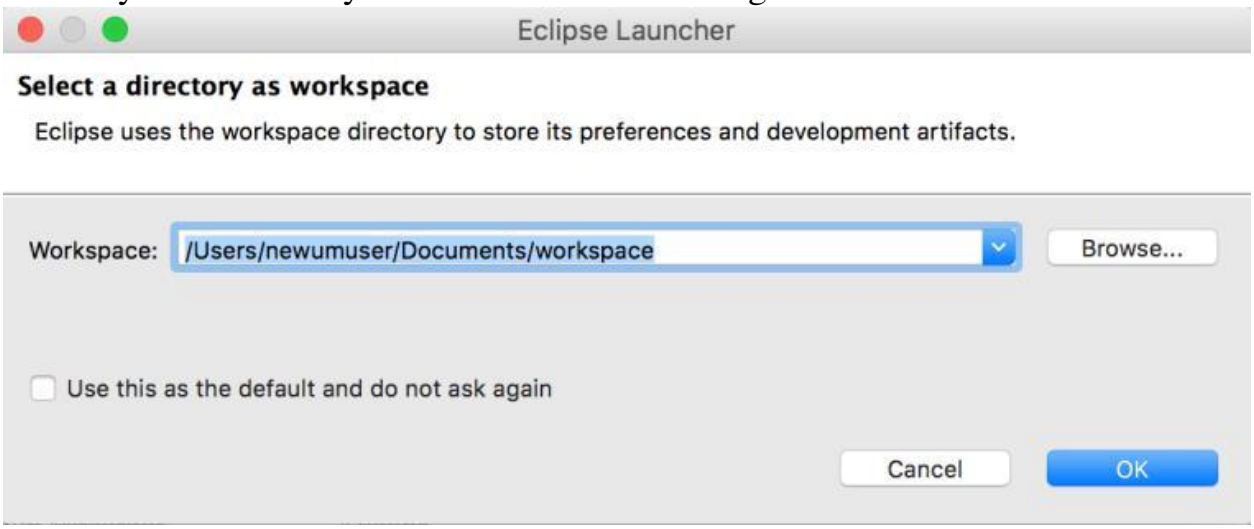
Part 1 - Create a Java project in Eclipse

We will now learn how to create a new Java project in Eclipse:

1. Launch Eclipse.

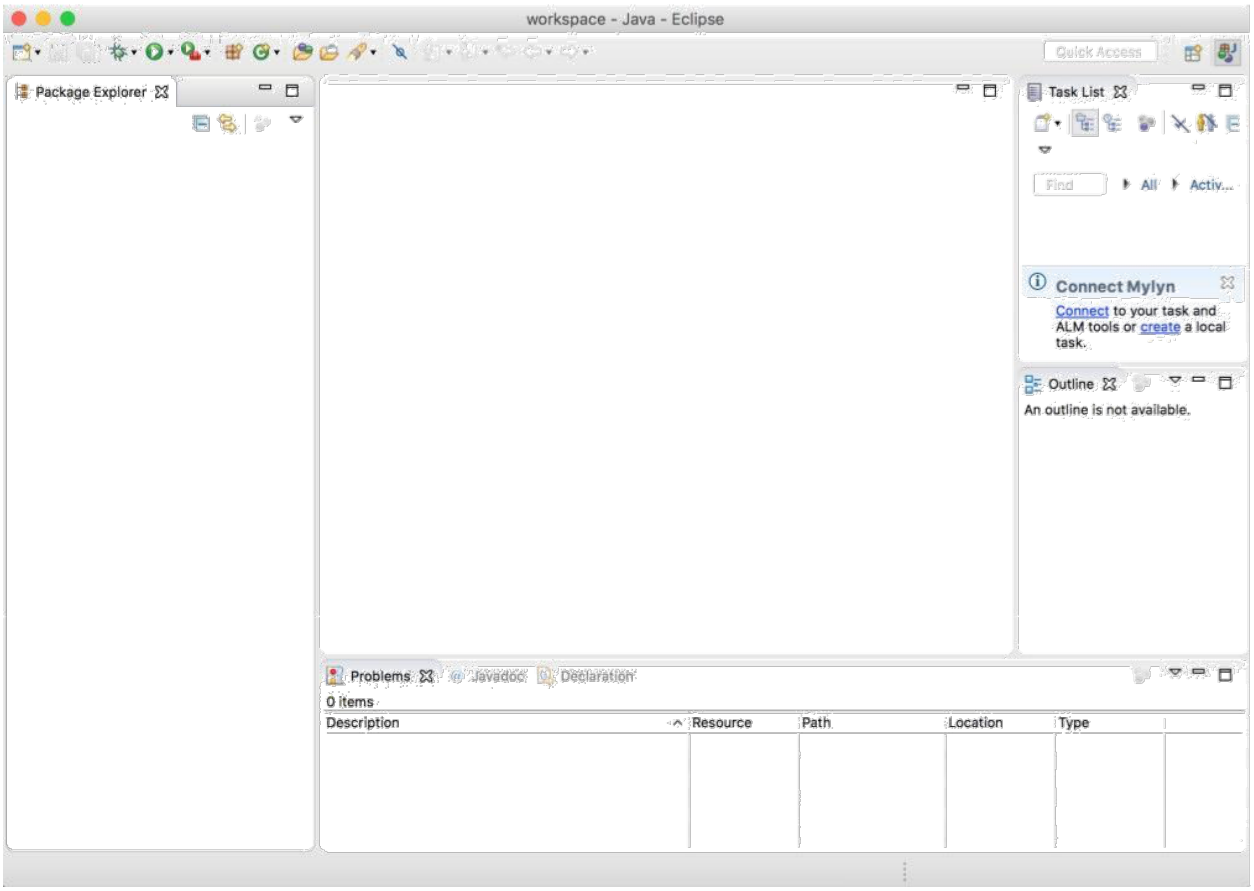
2. The very first window you will see is the following window:



3. This window is asking you where your "workspace" folder is. That means, where all your code and projects are going to be saved. Please make a note of this path. After hitting "OK" in this page, you will be directed to a "welcome page".

4. You are free to explore this page after class but none of the information in this page is required for our purposes. Simply close it by hitting the "X" sign on the top left (don't close the whole Eclipse).

5. Now you are in the main Eclipse page, it looks like the following figure.



6. The above window consists of several large sections for managing your Java projects.

    - **Package Explorer: This tab, on the left, is where you will eventually see a list of your java files.**

    - **Outline: This tab, on the right, is where you will see the organization of each Java file.**

    - **Center area: This area is where you will edit your code.**

7. Now we want to create a new Java project. We do so by selecting File->New->Java Project from the menu bar at the top. For the project name, enter project1 and hit finish to create your first Java project in Eclipse.

8. Your new project should show up in the Package explorer tab on the left. Open it by double clicking (or by clicking the small triangle next to the name). You will see a src folder and a JRE System Library folder. All of your work will be created in the src folder.

9. Right click on the src folder and select New->Package. Give the package the name project1 and click finish.

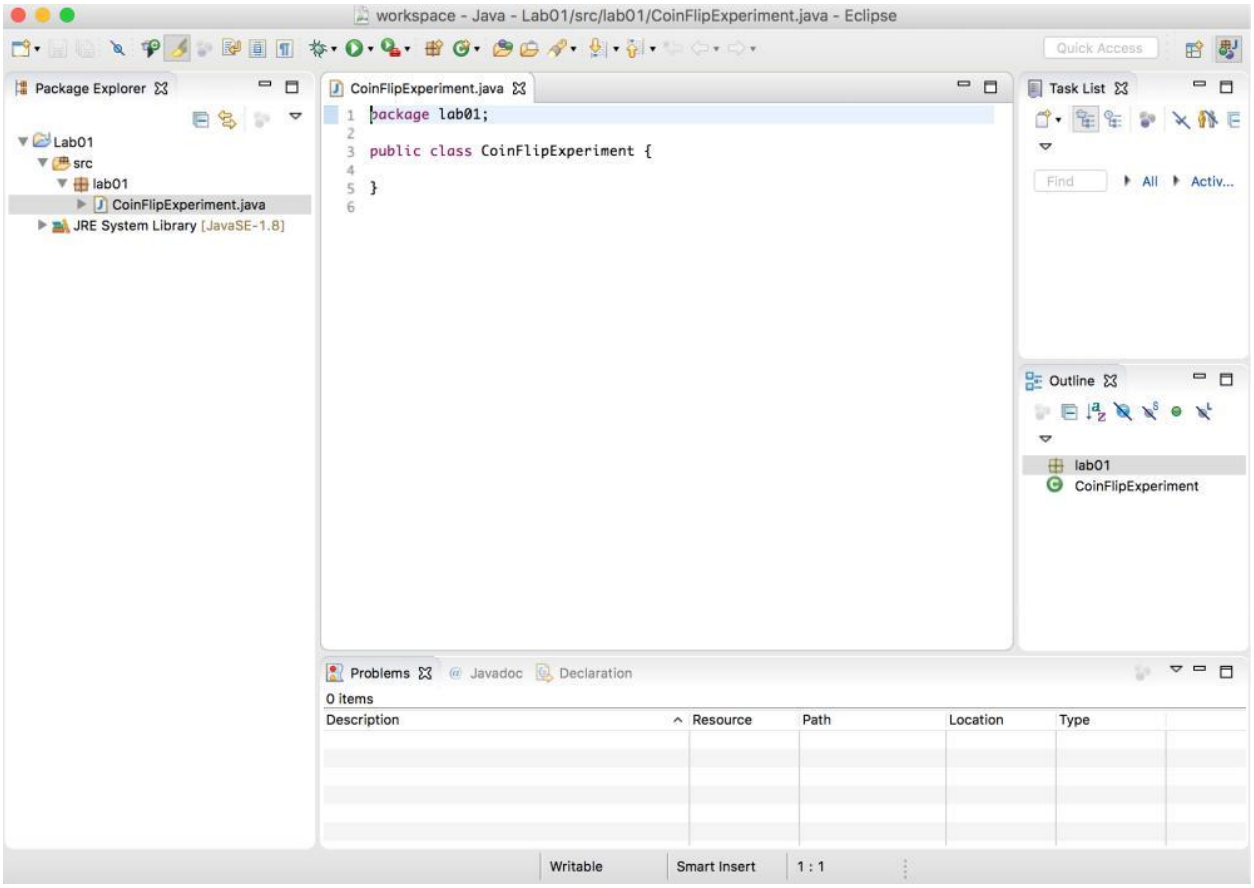
Now let's start programming :)

Part 2 – A Simple Experiment

Assume you have a coin that is slightly out of balance. When this coin is flipped, it comes up heads 50.5% of the time. You think to yourself that you can get rich using this coin, so you bet with friends on the outcome of a coin flip. The winner of each coin flip gets $1 from the loser and you always bet on "heads". Would you be very likely to win lots of money?! Let's code it up to see...

To start, create a new class named "CoinFlipExperiment" in the "project1" package as follows: 1. Right click on "project1" package that you just created. Then, select New->Class. In the

"name" tab, type "CoinFlipExperiment", and hit finish. Now you see a new file has been created under "project1" package called "CoinFlipExperiment.java" and you see the corresponding code in the center area of the window.



Your window should look like the figure above. Please double check the package Explorer view and the center area and if your window looks different consult with your instructor.

2. Now, let's test a simple "HelloWorld" program to learn how to compile and run Java programs in Eclipse. Copy and paste the following piece of code on line 4.

```
public static void main(String[] args) {
System.out.println("Hello, World"); // Prints to the terminal window.

}
```

3. Now you are ready to run your first program in Eclipse. To run your program, click on the little green button highlighted in the picture below

4. And then you should see your “hello world” appear on the bottom portion of Eclipse window (in the console section).

5. Now that you can successfully run a program in Eclipse, let us get back to our experiment. Replace the “System.out.println” line with the follow piece of code and rerun your code.

```
double flip = Math.random();

if (flip < 0.505) { System.out.println ("Heads");

} else {

System.out.println ("Tails"); }
```

6. Run your program multiple times and see the output.

7. Now, let us understand this piece of code. In the code above, the Math.random() method generates a random number between [0...1). If the random number is less than
50.5% (ie. 0.505), we treat it as “heads”, otherwise it’s “tails”.

8. Run your program a few times to make sure the coin flipping works. (Note, it is not unlikely for it to be the same five times in a row. It would be very unlikely to be
the same 10 times in a row. Make sure you run it until you see both results.)

Part 3 – Debugging

In this part, we are going to introduce you to Eclipse's debugger tool, and see how to use it to enhance your understanding of what your code is doing.

One of the most useful tools in Eclipse for debugging is a "breakpoint" so that we can stop the program execution on a specific line.

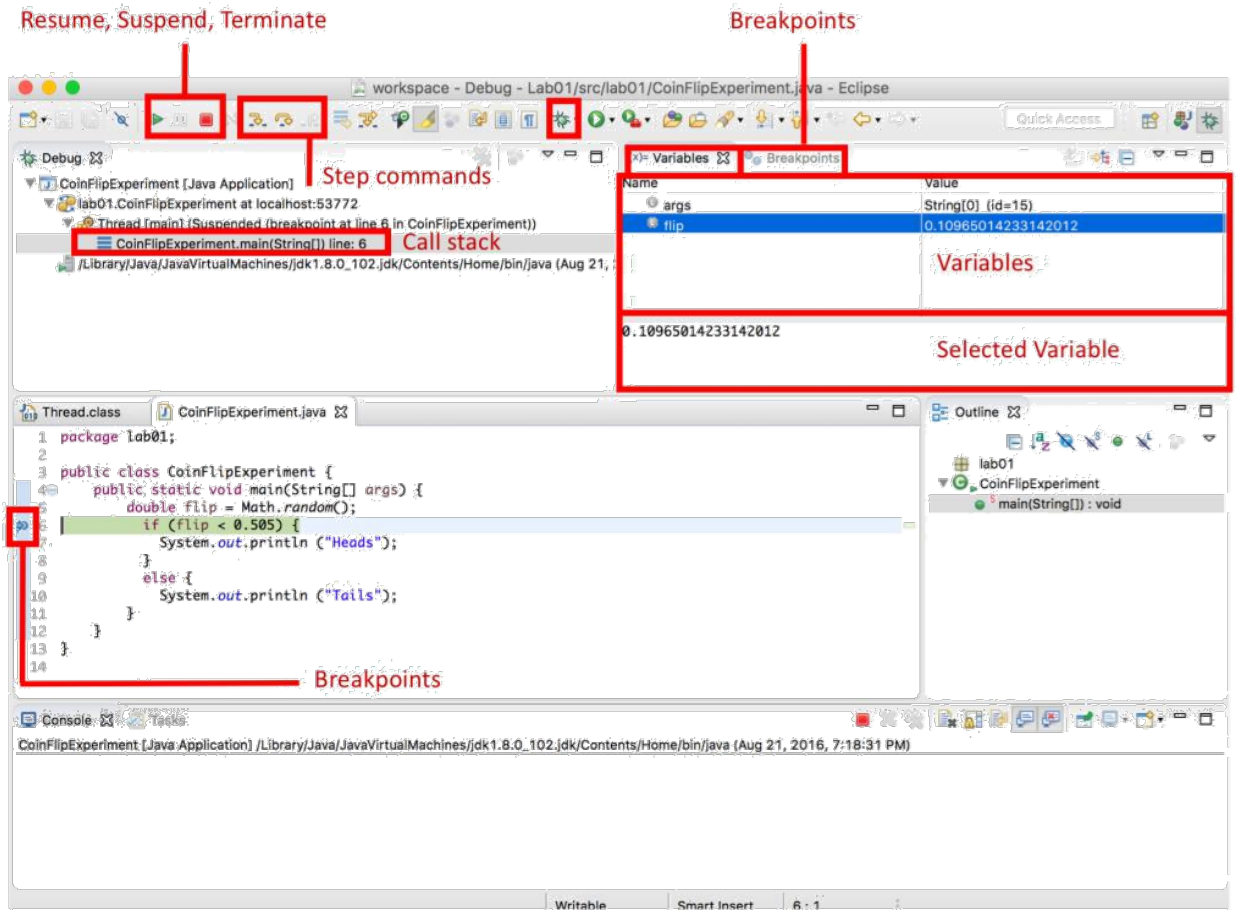
1. To create a breakpoint, double click on the line where you have the "if" statement. You will see a small circle appear on that line.

2. Now, if you run your program as before you won't see any difference, to stop the execution on this line you need to run the program in "debug mode".

3. Instead of running the program by clicking the "run" button, click on the button on its left (the little bug). A window opens up, asking you whether you want to switch to "debug perspective". Click "yes", and see what happens.

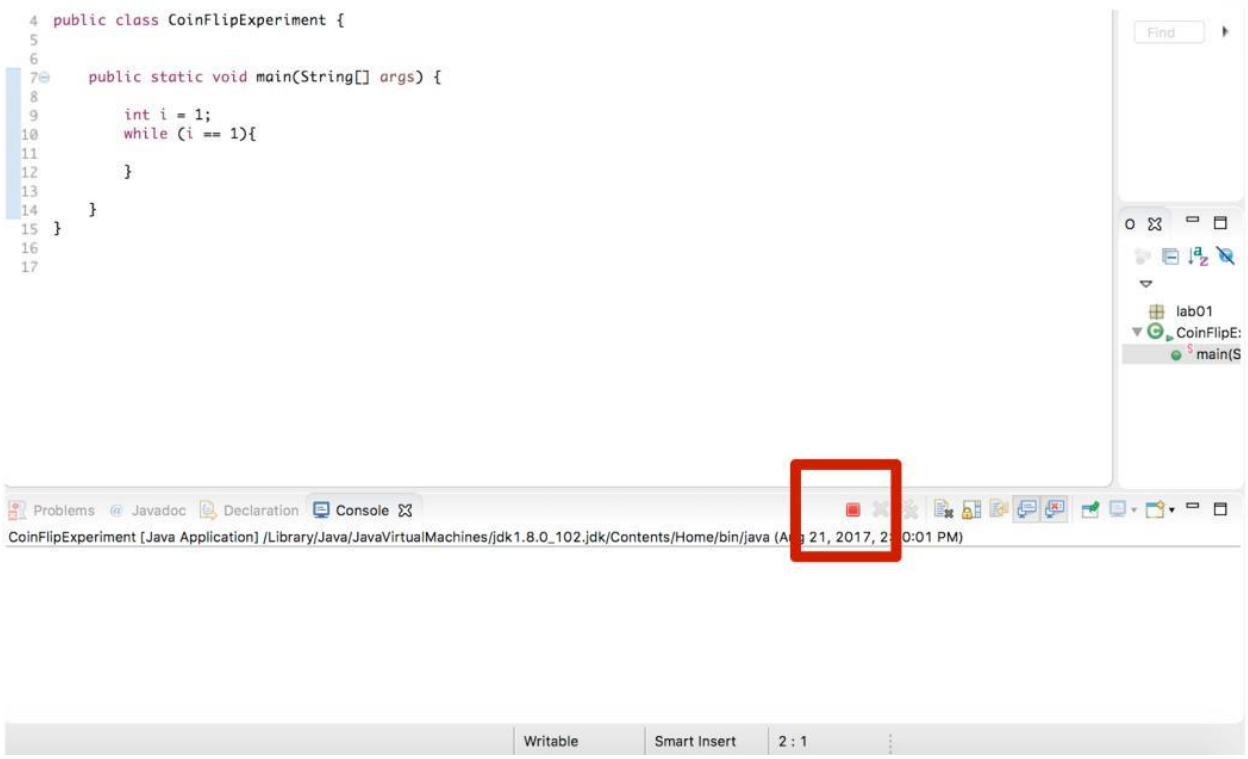
4. Now, you will see a window that looks like the following figure. Please spend a couple of minutes understanding various tabs you are seeing in this

figure before you go back to Eclipse.



5. Now you can go back to Eclipse and try different functionalities of the debug view.

6. Notice the red square on your console. When your program is done running this should turn into a grey square, if it stays red that means you have an infinite loop. The following shows you a snapshot of a simple infinite loop (you can also try it).



7. To get back to the original view of the eclipse window, you need to first "terminate" the program by clicking the little cross sign close to the red square (on the console window at the bottom) and then click on the little button on the top right - marked in the snapshot below.