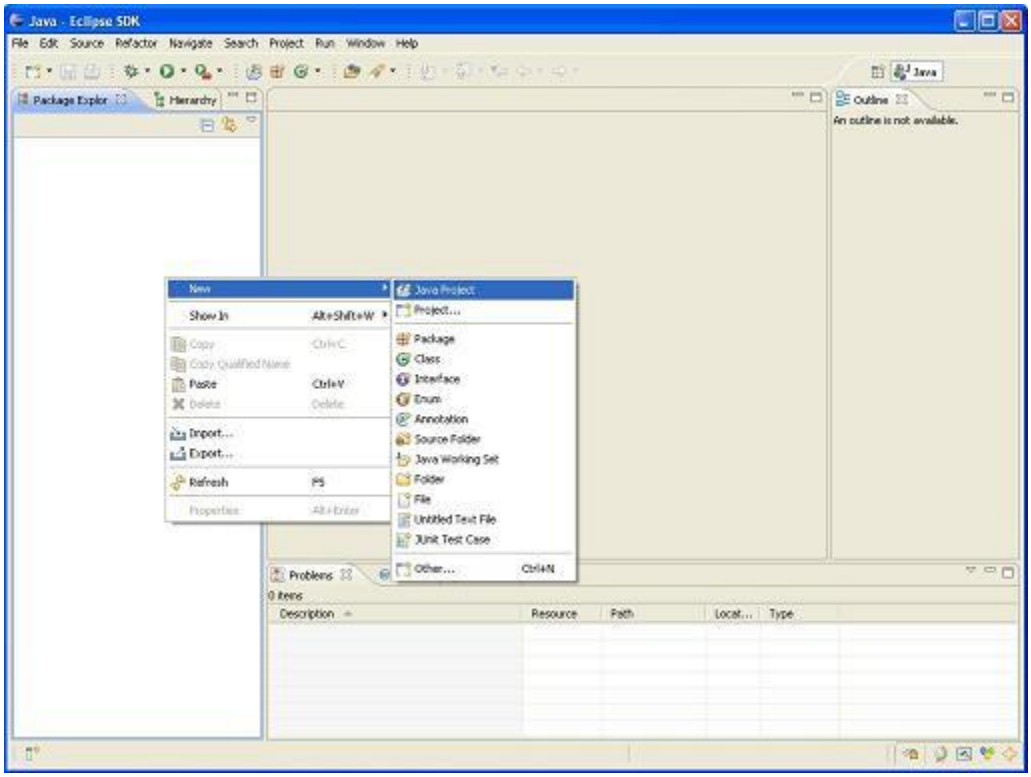
# Activity – Greeting

Create a Greeting application similar to the one shown in note 1b. Your program should say "Hello World!". The goal of this review is to write the source code for a simple application and then compile and run the code

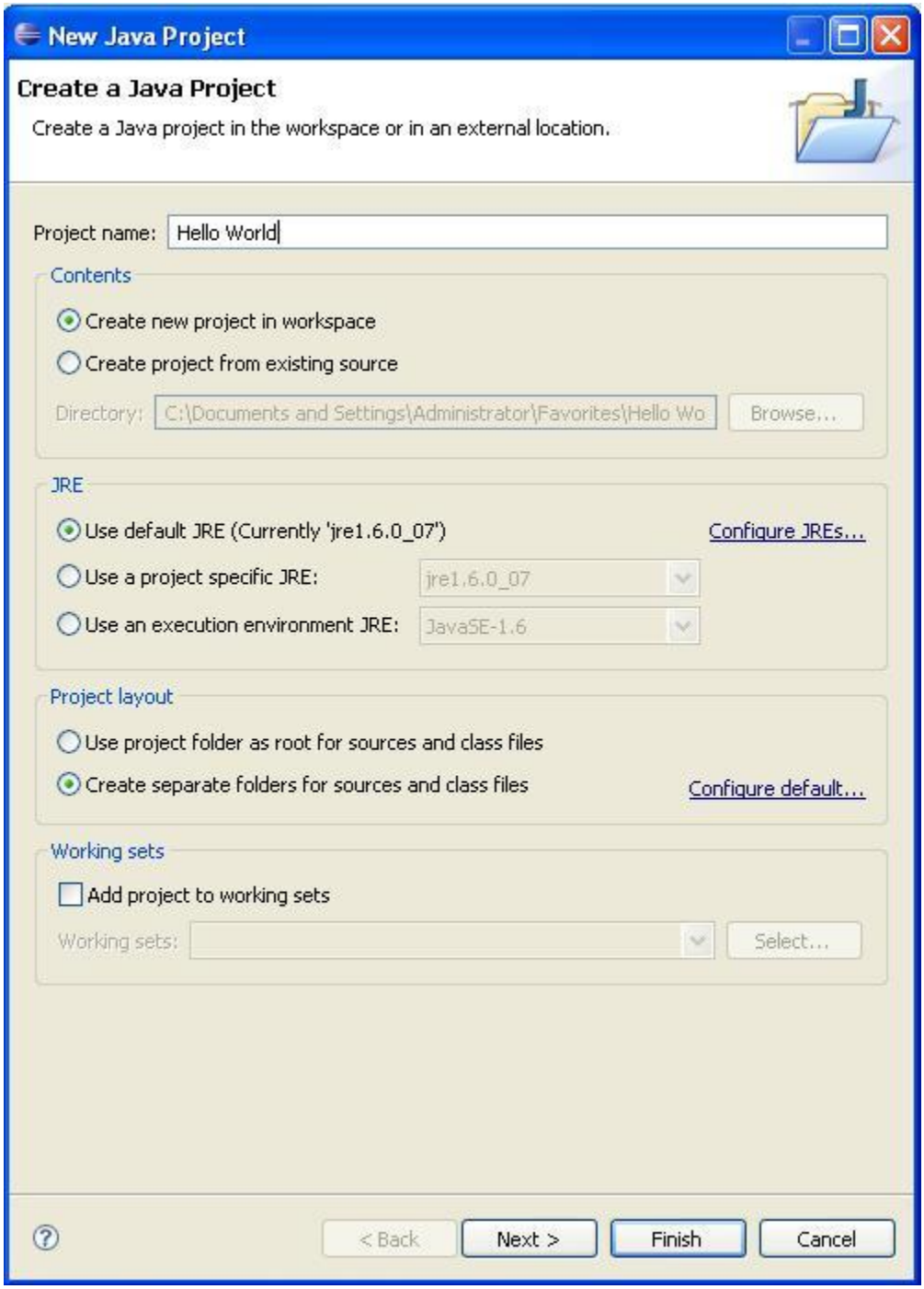
Specific Instructions using Eclipse:

- Because we are using Eclipse, we need to create a **Java Project** before we can begin coding.
- The project is simply a place to keep all the related files needed to create a Java Application
- Every new Activity or Project in this course should be created in its own new project

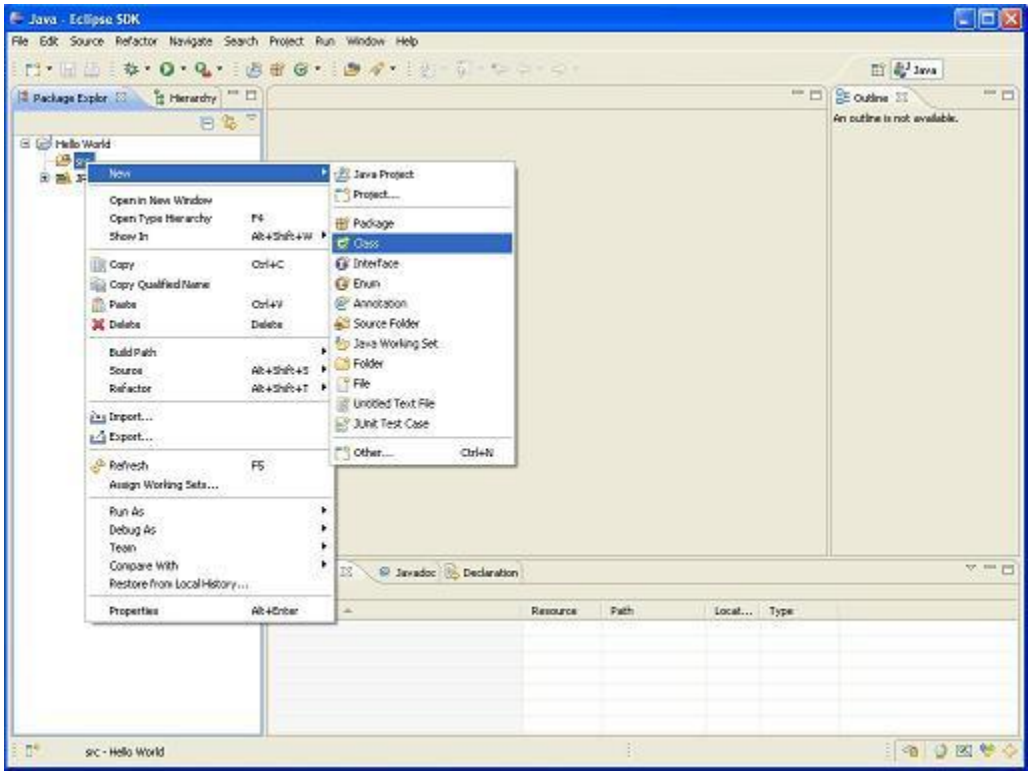
1) Start Eclipse
2) If the "Workspace Launcher" appears, click OK
3) Right click on the left side panel, choose new, and select Java Project



4) Name your new project "Last Name, First Name – Module 2 - Act 1c - Hello World", leave everything else how it was, and hit "Finish" (example name: "McDougall, Shannon – Act 1c, Hello World")




5) Create a new class. Expand your new project by clicking the arrow or plus sign next to the folder. There should be a folder called src and also a JRE System Library. Right-click on the src folder, go to new, and select Class.




6) Fill out the next form – give this new file a name – probably HelloWorld (NO SPACES!)
    a. Java files should always start with a capital letter as the first letter, no spaces
    b. Check the box that says "public static void main(String[] args)"
7) Click finish
8) Write your own program
    a. Some code is there for you already, here is an explanation of what you are seeing

**public class HelloWorld** is the beginning of your class file. Notice it has the same name as the file itself. The name, including the capitalized letters, must match the file name in Java or it won't work. Eclipse does this work for you of course. Helloworld is not the same as HelloWorld, because they are not exactly the same. The second HelloWorld has a capitalized W, the first one does not.

**public static void main(String[] args)** is called the main method. Without this statement, you cannot run your program. You'll notice two open brackets like this one in front of two of the lines of code:

**{**

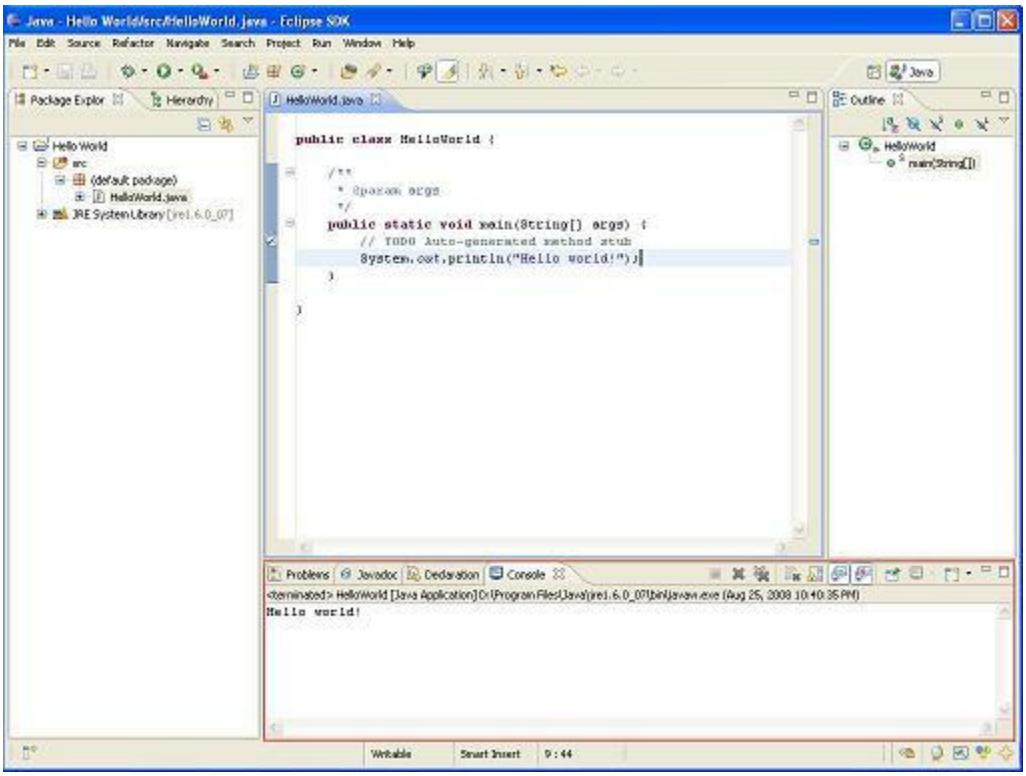
and then further down you see two closing brackets that looks like:

**}**

Anything you write inside the inner two brackets belong to the main method, and anything inside of the outer brackets belongs to the class. All methods and classes have opening and closing brackets

If something is wrong with what you typed, Eclipse will automatically underline the problems in red. If you get this, make sure everything is spelt correctly and that you are using correct capitalization.

When there are no longer any red underlines, right click the text screen (anywhere in the box where you wrote your code), go to **Run As**, and select **JavaApplication**. If it tells you that you need to save the file, save it, and then watch it run. In the little console at the bottom of the screen, you should see it display Hello world! That means that the program works, and you've now written your first program in Java! Congratulations!




I put a red border around the output screen. That is where Java displays its output in Eclipse.

Congratulations, you have just finished your first Java program!

6) Last Step – Hand in your activity. Activities will be submitted differently than assignments. For activities, open the Google Doc that was shared with you called: "ICS3U – Activity Submission Form". Find the appropriate activity in the table provided, add the current date, and then copy and paste the full source code (everything in the .java file). Make sure that your program runs properly before submitting!