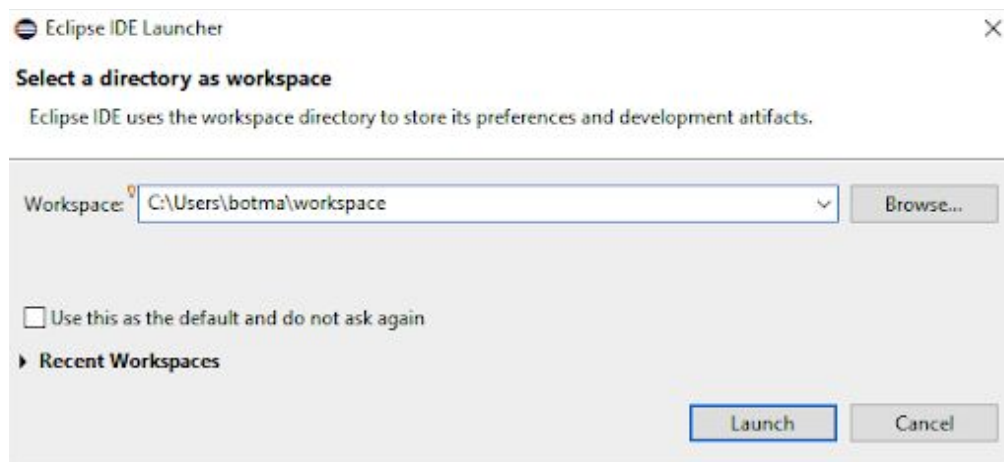# Java Blog

Just Beginning: The Hello World Tutorial
Henry Barthelemy and Evan Wologodzew, July 2020

The first thing programmers do when learning to code in a new language is a "Hello World" program. Down to the foundations, the first thing to learn is how to make the computer say "Hello World". It is a thrilling first step that leads to a world of possibilities.
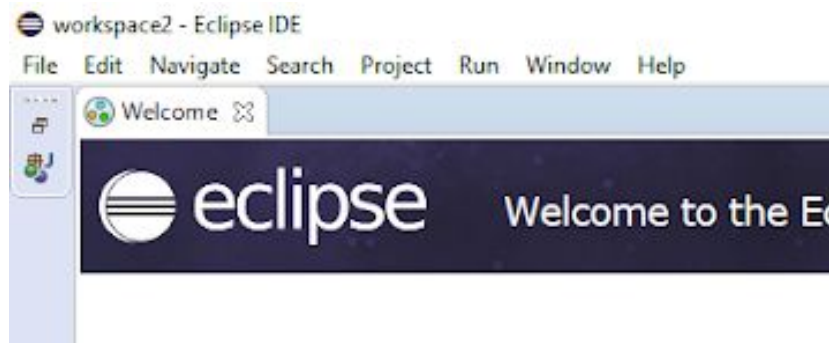
First step is to launch Eclipse.

After launching it will ask you to select a folder as workspace. This is where all your project files will be saved.
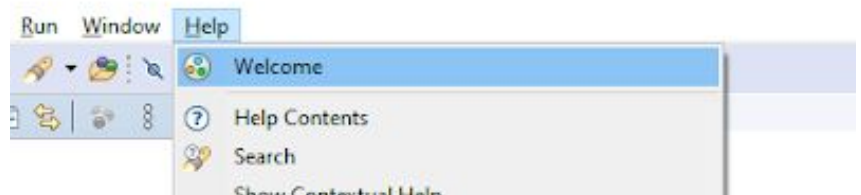
I would recommend going with the default, but you can create a new folder and select it using "Browse" if you'd like.

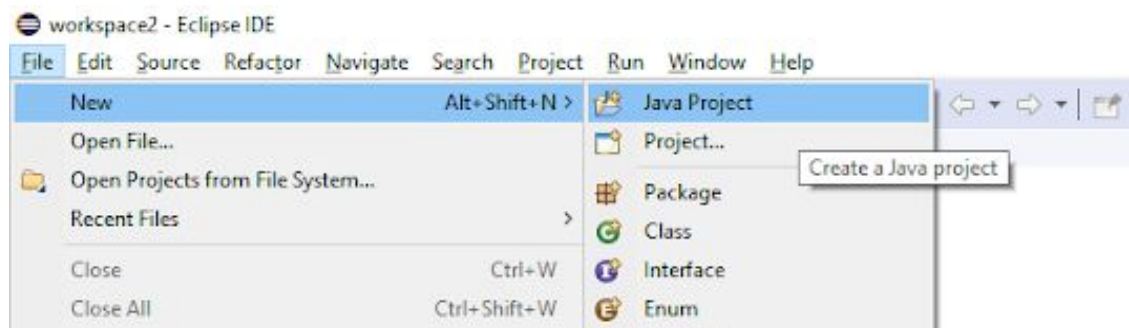Once you are ready to proceed, hit "Launch".

On the first launch your workspace should have a menu similar to this.

You can exit out of the welcome menu (if you'd ever like to go back and read it, you can hit Help and then Welcome).
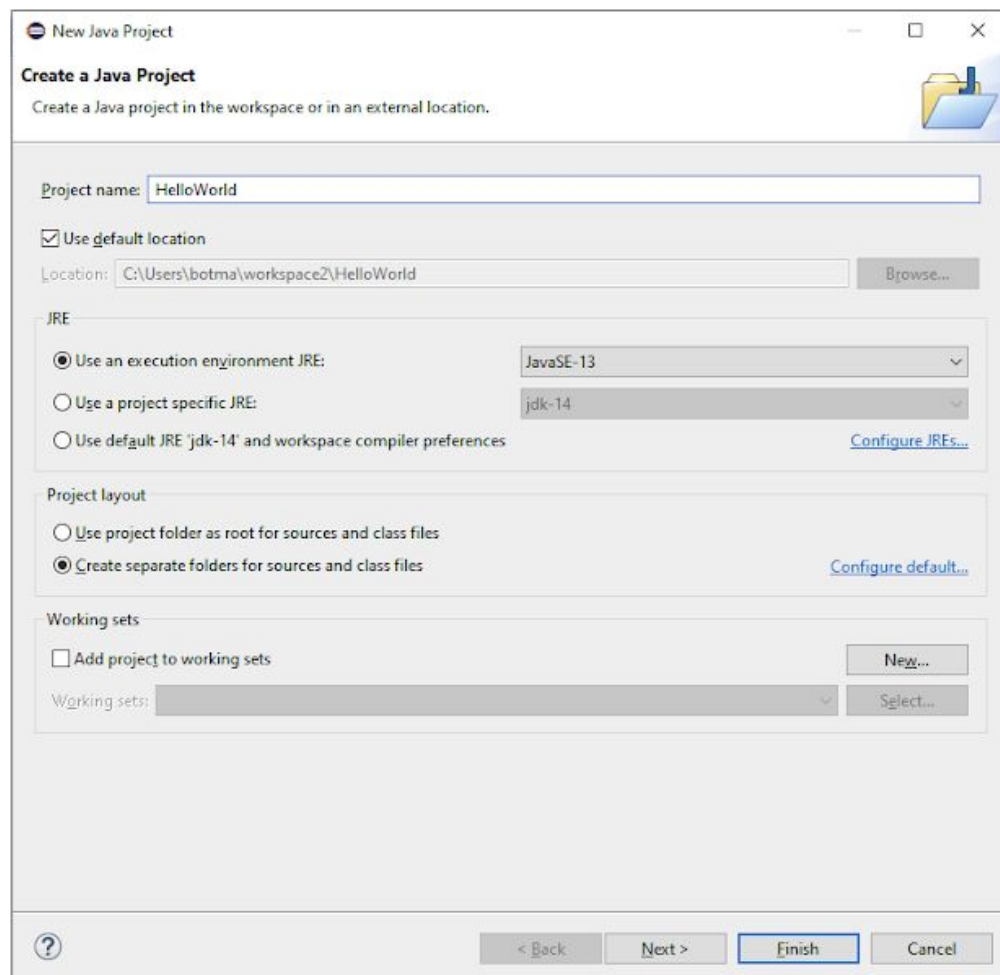

After exiting from the welcome screen we will create a new Java Project. This is where we will write our code to launch and compile.
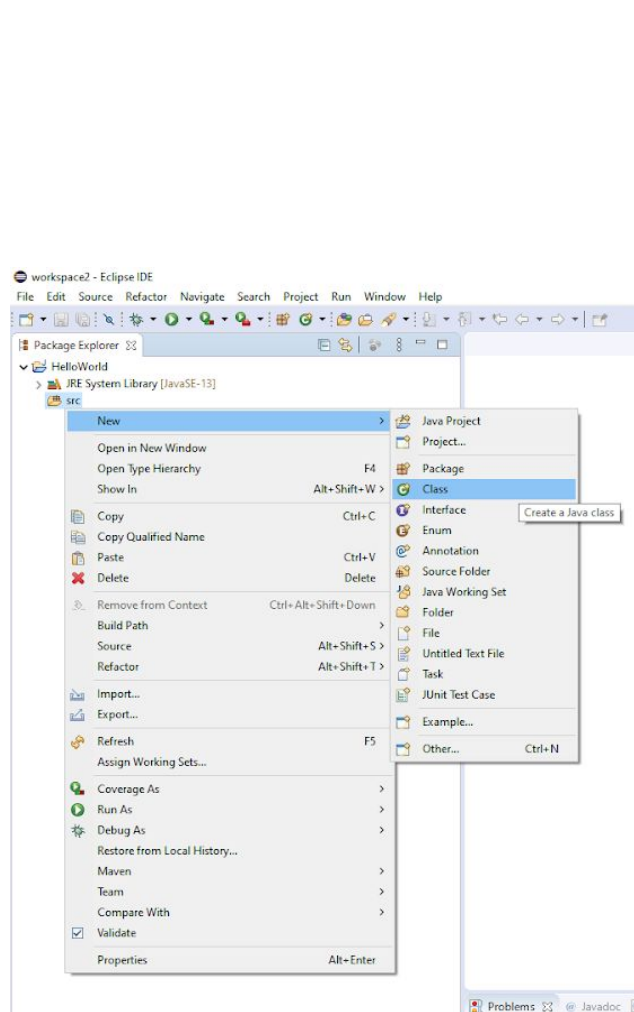
In order to accomplish this hit **File → New → Java Project**

When creating a Java project, this window pops up. Here you will name the project by writing in the "Project name:" tab, everything else can be left as is. For my project I named it HelloWorld. After you name yours, simply hit the Finish button.

If anything pops up asking to create a module you can either hit create or don't create, it won't affect what we're doing.
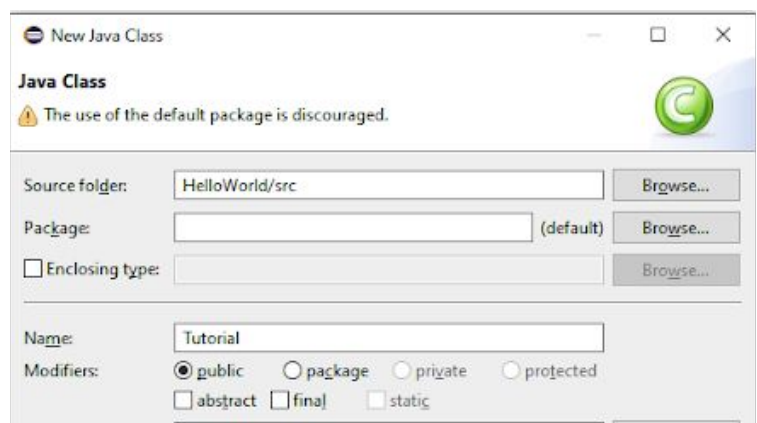

Following, we are going to create a class. This is where your code will be written. We create the class by right clicking the "src" file and then navigating to **New → Class**, as shown on the right.

A class in Java is the basis for object oriented programming, which we will go over in a later section.
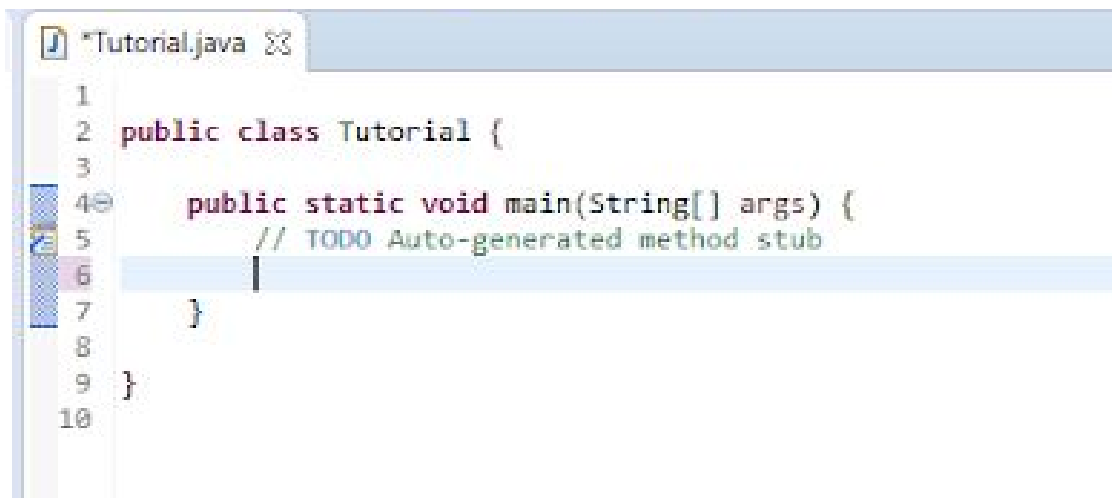
For now a Java class is where we write our lines of code, and the file we compile and deploy in order to accomplish tasks.

I named this class "Tutorial" under the name tab. Feel free to name is whatever you would like. The typical naming

convention would be to capitalize the first letter and have no spaces. For example "FirstLauncher" or "Dog" would be good class names.

Before hitting finish make sure to click the check marker for "public static void main(String[] args)". This will save us some typing in the next step.

Your newly created class should be similar to the image above. If it is not on the screen, double click the class on the project navigator on your left. By checking the box earlier, we automatically created what is called the main method. This is the first thing java will look for when running a program. To write inside something, we write between the squiggly brackets they have (These things: "{ }"). The lines inside are also intended to help organize the code. To format unneat code, you can also use the command: Ctrl + shift + F.
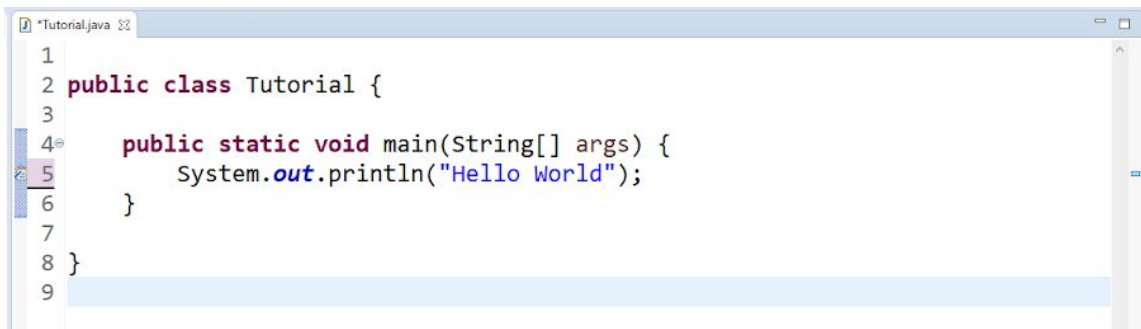
Our code will be written in the main method, as everything inside will be the first thing to be run when we hit that run button.

Inside we can delete the comment (the green line), it is not necessary to have. Following we can write our first line of code: a print line statement.

System.out.println("Hello World");

This line allows us to print out a line to the console, in this case we are printing out a string to the console. A string is a set of characters strung together.

Sidenote: println is pronounced print line and spelt as a shorthanded version of print line, where the space, i, and e is removed
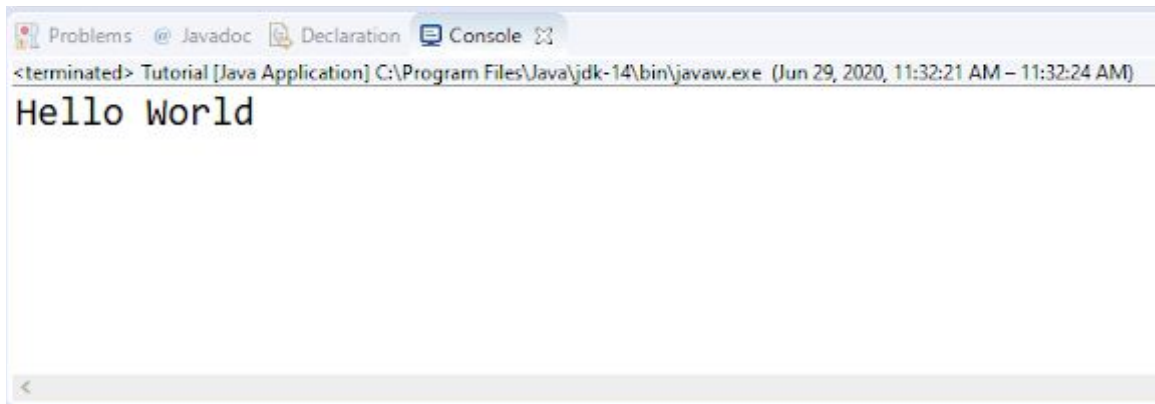
```
 1
 2 public class Tutorial {
 3
 4     public static void main(String[] args) {
 5         System.out.println("Hello World");
 6     }
 7
 8 }
 9
```

Your final code should look like something along these lines.

avigate   Search   Project   Run   Window   H

At this point you are done! Simply hit the run button, which looks like a green circle with a rightwards pointing arrow. It is located on the top menu bar.

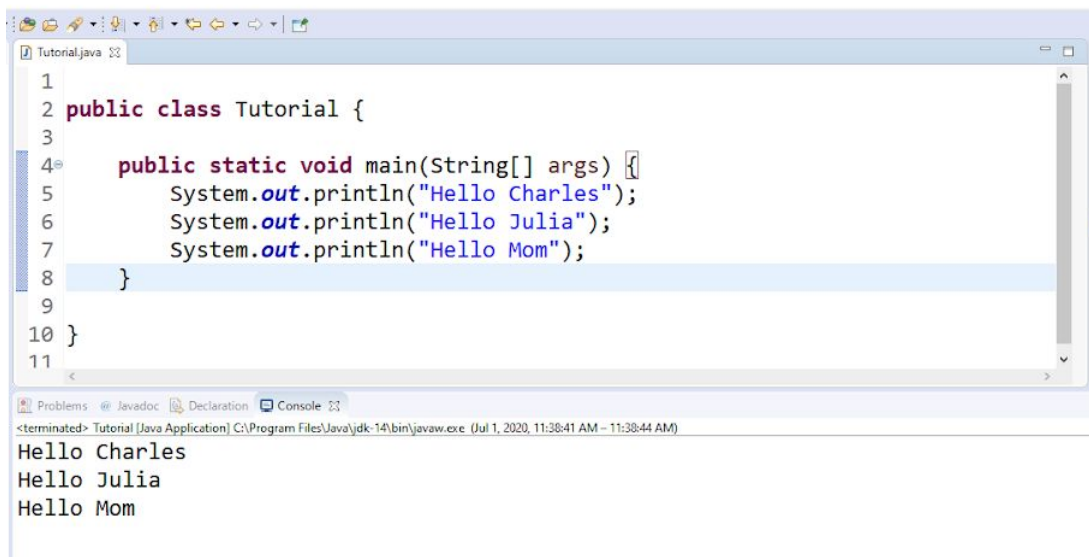Congratulations! You just just wrote and ran your first line of code!

```
Hello World
```

<

Challenges for Hello World Tutorial:

The more you get comfortable with what you're doing the faster you learn. Print a hello to everyone in your house, one person per line.

Solution:

```java
1
2 public class Tutorial {
3
4    public static void main(String[] args) {
5        System.out.println("Hello Charles");
6        System.out.println("Hello Julia");
7        System.out.println("Hello Mom");
8    }
9
10 }
11
```

```
Hello Charles
Hello Julia
Hello Mom
```