CS 152 Lab 1

**Description:**

In this lab we are going to be creating new Java projects and attempting to both compile and run them. To do this, we’ll need to begin using VSCode and the JDK.

Let’s Begin.

**Step 1**

If you haven’t already, download and install VSCode and the JDK. Links can be found in the syllabus. Once you have VSCode installed you must also add the Java Extension Pack to your environment.

A screenshot of a computer

Description automatically generated

This extension will allow you to more easily debug and run your Java code.

**Step 2**

Now we are going to create a new java project. To do this first click on the search bar at the top of your editor.

A screenshot of a computer

Description automatically generated

Next type in the **“>”** greater than symbol to initiate a command. After that type in

**“Java: Create Java Project”** and hit enter.

**A screenshot of a computer

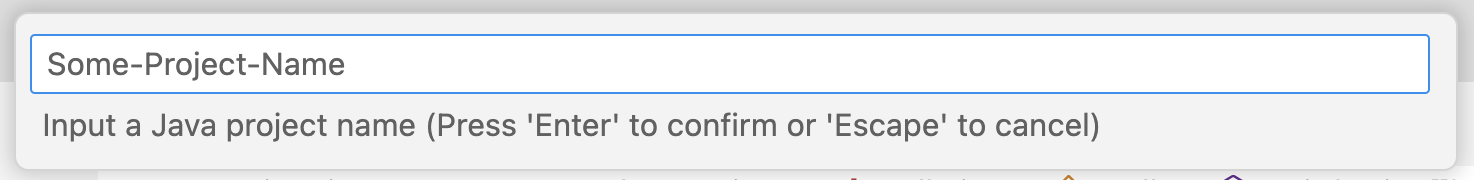
Description automatically generated**

Next select **“No Build Tools”** and after that find a safe place in your file system to store this new java project.

A blue and white box with white text

Description automatically generated

Finally, name your project something and hit enter.



**Step 3**

You should now be inside of your new project.

A screenshot of a computer

Description automatically generated

Inside of the window that just opened there are a few panels and buttons to be aware of:

1. **The File Explorer which allows you to find and open files within your projects as well as add and move files if necessary.**
2. **The File Editor which allows you to open several files in separate windows and edit them.**
3. **The Play button, used to run your java project. You can also use this button to debug projects if you wish.**

If your project and editor is set up correctly you should be able to simply hit the play button, and a new panel will open up at the bottom of the window with the terminal and output from the run attempt.

**Step 4**

Inside of this terminal that opened when you hit play you can also compile and run your files manually.

**A screenshot of a computer

Description automatically generated**

To run your project manually you can into your terminal type:

**cd src**

**javac App.java**

and then

**java App**

These commands took you into the src folder, compiled the App.java file with javac, and then ran the App.class file with java. You might notice that there is now an App.class file inside of the src folder in your file explorer.

A screenshot of a computer

Description automatically generated

When you ran your program with the play button you generated a .class file as well but this onw was placed out of the way in the bin folder.

A screenshot of a computer

Description automatically generated

This is a better place to store these .class files because they do pile up and aren’t very useful to us when editing the program. You can compile to the bin by simply changing our previous command a bit like so:

**cd ../**

**javac src/App.java -d bin**

and then

**java -cp bin App**

Before you run these try deleting all your .class files and seeing where the .class file ends up.

Now you can run your java project manually and automatically while keeping your src folder nice and clean.

**Step 5**

Inside of your App.java create a header comment to denote details about the program like this:

A screenshot of a computer

Description automatically generated

**Step 6**

Now inside of your code try and create a program that, when run, will return the following content to the terminal window:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Your Name \*

\* Your Track, e.g. Data Science \*

\* Your Favorite Color, e.g. Blue \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Remember that the println() method will go to a newline while the print() method will not.

Finally, run your program and tell me to come check your output.