Programming Exercise 0

Authors: Vince, Jack, Andrew

Purpose

Hello and welcome to your first CS1331 programming assignment! This assignment is just to verify you have Java installed properly on your machine and you are able to run a Java program.

Solution Description

`Installing Java`

In this class, we will be using Java 11 (openjdk). Make sure you have this version installed, even if you already have a previous version of Java. We have detailed instructions for this on the course Canvas page in "Guide to Installing Java.pdf".

After running `javac –version` , you should see similar output to the following:

```
javac 11.0.12
```

Take and save a screenshot of your terminal showing the result of `javac –version`

After running `java –version` , you should see similar output to the following:

```
openjdk version "11.0.12" 2021-07-20
OpenJDK Runtime Environment Temurin-11.0.12+7 (build 11.0.12+7)
OpenJDK 64-Bit Server VM Temurin-11.0.12+7 (build 11.0.12+7, mixed mode)
```

Take and save a screenshot of your terminal showing the result of `java –version`

NOTE: It's okay if your minor version reported is different. As long as the number reported is "11.0.x" you will be fine!

`Writing and Compiling a Java Class`

In order to create a Java class, you must first create a file with the .java extension. In our example here, we will use `Test.java`. Within the class, you will need to write a class header. For now, just memorize that you need to write `public class` before your class name. We will go more in depth about what exactly this means later. As for the class name, it must exactly match the name of the file.

Therefore, our example should have the following (note the capitalization):

```
public class Test {

}
```

Now that you have written a Java class, you should be able to compile your code! Navigate to the directory where you have saved this file in the command line.

After navigating to the proper directory, you will be able to compile the Java class you've just created! You do this by using the command `javac FileName.java`. Since we named our Java example file `Test.java`, we compile it using `javac Test.java`.

This should create a new file in your directory called `Test.class`! Created by the Java compiler, this file will contain Java bytecode, and it will not make much sense to you if you open it up. However, it makes a lot of sense to the Java Virtual Machine, or the JVM (a computing machine that allows our program to run).

If a new file was not created, your program did not compile. Read the compilation error produced in your command prompt and fix it before proceeding!

The Main Method

Remember, the method signature should look like this:

```
public static void main(String[] args) {

}
```

This method is critical for a Java class to be "runnable." Now, you can write the contents of your first Java program within your main method. For this assignment simply do the following:

- Print out "Hello World"

Now compile your code and correct any error messages that may appear. In the future, as programs get more complicated, it is good practice to compile as you make changes. This will help you correct any errors you may make as you go instead of dealing with them all at the end.

To run your program, use the command `java FileName`. Since we named our file Test.java, we will use the command `java Test`. Note that we didn't include the .class extension after `Test`; it won't run if you do `java Test.class`. If you ever want to run a new version of your program after making alterations, know that you must recompile before doing so.

Good luck, and happy hacking!

Turn-In Procedure
Submission
To submit, upload the files listed below to the corresponding assignment on Gradescope:
- `Test.java`
- Screenshot of your terminal showing the result of `javac -version`
- Screenshot of your terminal showing the result of `java -version`

Make sure you see the message stating "PE0 submitted successfully". From this point, Gradescope will run an autograder on your submission verifying all necessary files are included and working as intended. Note: the nature of some assignments means they are not fully autogradable and will require some manual grading after submission. Although all autograder tests will be made visible, this is NOT always your final grade.
You can submit as many times as you want before the deadline, so feel free to resubmit as you make substantial progress on the homework. We will only grade your last submission so be sure to **submit every file each time you resubmit**.

Collaboration

No collaboration is allowed on this assignment. See syllabus for more details.

In addition, note that it is **not allowed to upload your code to any sort of public repository**. This could be considered an Honor Code violation, even if it is after the homework is due.