Assignment 00: Hello World!

CS 140 with Dr. Sam Schwartz

Due: Sunday, September 7 at 11:59pm via Canvas Upload

1 Purpose

The computer science faculty have voted on Java as the introductory programming language for CS 1. This assignment is to ensure that you are able to code in Java post-haste.

In all environments where you start programming in a new-to-you programming language – including in this course – the first set of tasks will always include setting up the infrastructure to go about actually coding in that language. By tradition, this involves developing a program called "*Hello World!*" which simply displays the phrase "Hello World!" on the screen, usually in a terminal.

The work of getting your infrastructure set up to develop and launch code is known as "developer operations" or – more commonly – "DevOps." DevOp engineers earn a median of $165,000/year. [1] You may soon learn why these developers are in such high demand as to command this kind of salary. There are a lot of steps involved in getting started, and things can go wrong very quickly. Pay close attention to small details; computers can be very unforgiving.

2 Tasks – High Level Overview

In this assignment, you will:

1. Install the Java Development Kit.
    - The Java Development Kit, or JDK for short, contains the underlying engine which allows you to turn human-understandable code into machine-understandable code. Think of it like Google Translate.
2. Install Eclipse, the traditional and most widely used IDE for Java.
    - IDE stands for "Integrated Development Environment." Think of it like Microsoft Word or Google Docs, but for Java. It has baked-in tools to let you use the JDK easily.
3. Write a "Hello World!" program.
    - The point of doing this is to verify that the JDK and Eclipse IDE were installed correctly and that you can, indeed, successfully write and run Java code.

On the next page are detailed steps of what needs to happen.

[1] https://www.levels.fyi/t/software-engineer/focus/devops?country=254

3 Tasks – Detailed Requirements

3.1 Install JDK (Java Development Kit) 21

Note: JDK 24 is the newest version, but also is less stable. JDK 21 is the most recent "Long Term Support" (LTS) version. You can use any version of Java, but I would go with JDK 21 for this class.

1. Go to the official Oracle download page at https://www.oracle.com/java/technologies/downloads/#java21.
2. Download the Windows x64 Installer for JDK 21.

   Alternatively, download the x64 .dmg file for Mac.
3. Run the installer, keep all the default settings, accept the license, and finish.
4. Open *Command Prompt* (Windows) or *Terminal* (Mac) and type

   ```
   java -version
   ```

   followed by the enter key.

   Same for this command:

   ```
   javac -version
   ```

   You should see the installed version of Java. Congratulations! Your first task is complete.

3.2 Install Eclipse

1. Go to: https://www.eclipse.org/downloads/
2. Download Eclipse IDE for Java Developers.
3. Install: Run the installer and follow prompts. Keep all the defaults.
4. On first launch, Eclipse will ask for a workspace folder (where your projects are stored).

   Select the folder wherever you want to keep your homework files for this class and assignment. (I, Dr. Schwartz, keep all my files under `Desktop/2025_Fall_Workbench/CS140/HWCode`)

3.3 Write Hello World!

1. Open your Eclipse Application.
2. Create a new file by going to the menu bar and clicking File → New → New Java Project.
3. Call this project *HW00*. No spaces.
4. Deselect "Create module-info.java file".[2]
5. Keep all other defaults the same and click "Finish."
6. In the Package Explorer (it's on the left panel - it may be minimized within the window), expand the HW00 project and then right-click on `src` → New → Class. (`src` stand for "source code" and is where actual code is located. A Java *Class* isn't like a college course. Rather, it means 'Class' as in 'Classification', or a type of thing. In this case, in our source code folder we are creating a new Java file called `HelloWorld.java` which contains a new type (or class) of thing: our HW00 code.)

[2] Modules are used to help coordinate the logistics of tying multiple files of code together. We will be starting with code in a single file; modules are overkill for our purposes for now.

7. In the window that pops up, complete the following:
    - Name: `HelloWorld`
    - Check `public static void main(String[] args)` near the bottom of the panel (so Eclipse will auto-generate `main`).
8. Click Finish.
9. Edit the code so that it looks *exactly* like the below. (Copy-paste of the below snippit is fine):

```
public class HelloWorld {

    public static void main(String[] args) {
        System.out.println("Hello World!");
    }

}
```

10. Click the green *Run* button in the toolbar at the top of the screen (it looks like a play button, and it's the biggest one). Tada! A terminal should open and display "Hello World!"

4 Grading Criteria / Rubric

Elements of nearly all assignments in this class will be broken down into a 0-1-2 scale.

- "2" means, "Nailed it!"
- "1" means, "Umm, kinda got it, but not really."
- "0" means, "Uh-oh. Didn't get it."

This zeroth assignment is different from most other assignments in this class. It is so important that you get all the parts of this assignment right – your ability to do any more work in the class depends on it – that the only final score for this assignment is either 0% or 100%. **Partial credit will result in an overall score of zero (0%).** If you earn less than 100%, it is required for you to resubmit this assignment again (and again, and again) until you earn 100%. I will grade this assignment daily; expect a score within 24 hours of submission. There is no late penalty for resubmits, so long as the assignment is resubmitted within 7 calendar days of the due date noted above.

Rubric

The student submitted a screenshot to Canvas	0	1	2.
... which contained an image of Eclipse	0	1	2.
... which is opened to a file named `HelloWorld.java`	0	1	2.
... which contains *correct* code for a Hello World program	0	1	2.
... and also displays output to the console/terminal.	0	1	2.

Subtotal (of 10) .....

Total ($\lceil 100 \cdot \frac{\text{subtotal}}{10} \rceil$) .....

Example Screenshot Worthy of 100% Credit