Assignment 01: Hello World

CS101 - Intro To Computer Science

Spring 2024

1 Introduction

This first assignment is to configure your computer for writing and running Java programs and practice submitting an assignment for this class. By the end of this assignment you will have written and run Java programs on your computer and submitted your programs to Gradescope so they can be graded.

This assignment is comprised of six steps:

1. Install the Java Development Kit to *compile* and *execute* the Java programs we will write.
2. Install Visual Studio Code to *edit* the programs we will write.
3. Configure a terminal to invoke the commands for compiling and executing java programs.
4. Utilize the aforementioned steps to run `HelloWorld` on your computer.
5. Extend `HelloWorld` to be a program called `HelloStudent` and run it on your computer.
6. Submit your assignment to Gradescope.

2 Questions

1. Java Development Kit

As we will discuss in the classroom, the Java Development Kit (JDK) provides the requisite tools for compiling, executing, and debugging the Java programs that you will write in this class. We recommend installing version 17 because that is what we will use when grading your assignments. The installer can be found here.

Depending on your computer you will want to use a different installer:

- **Windows:** Use the x64 Installer
- **Mac:** If you are using an M1/M2 Mac you are using an *arm64* processor. If you are using an Intel Mac you are using a *x64* Mac. Use the DMG installer for your appropriate platform
- **Linux:** It's easier to install using your package manager. For Ubuntu/Debian use `apt install openjdk-17-jdk`.

2. Visual Studio Code (VS Code)

Throughout this course, we will use VS Code to edit the software that we develop. In subsequent assignments we will also use it for generating "projects". Although there are a plethora of alternative editors and means of running Java programs this class requires the use of VS Code. If you ask for help using alternative tools that the teaching assistants are not familiar with they will have a difficult time helping you.

We will also require the installation of a VS Code extension called *Extension Pack For Java* by Microsoft. This extension lets VS Code, among other things, format, fix, and run the software you will write in this class by connecting it to the Java Development Kit. You can install the extension by navigating through the Extension Marketplace.

2.1. Install VS Code for your particular platform:

- **Windows:** Link
- **Mac:** Link. Make sure to also follow the Launching from Command Line instructions
- **Linux:** Link

2.2. Install Extensions

- Install the *Extension Pack For Java*.
- Install the extension *prettier* by searching for it on the Extension Marketplace to help format your code.

2.3. Enable *format on save* in the VS Code settings to automatically format whenever you save a file.

3. Configuring Terminal

Before graphical user interfaces, the text-based interfaces, often called shells, emerged as the primary interface for interacting with computers on a "terminal". Even though we usually use graphical interfaces to interact with computers now, every sort of computer still comes with shells and terminals. Mac and Linux share a common lineage (from UNIX) and therefore have compatible shell interfaces like Bash and Zsh. For our purposes either Bash or Zsh will be sufficient and we will just use the built-in shell/terminal interfaces.

On the other hand, Windows is based on the DOS terminal which does not use a compatible shell interface. Because of this, for this part of the class, Windows users are required to install a compatible shell and terminal.

3.1. **Windows:** Git comes with a Bash shell/terminal: Git for Windows. Please install it for this section of the class.

3.2. **Mac:** Use the Terminal app that comes with your operating system.

3.3. **Linux:** Use a terminal that comes with your operating system.

4. Hello World

With the above tasks done you can now write your first Java program.

4.1. First, decide upon a folder where you will place all your assignments and create a folder named `cs101_assignments`, for example:

- **Windows:** `C:\\Users\[your_user_name]\Documents\cs101_assignments`
- **Mac:** `/Users/[your_user_name]/Documents/cs101_assignments`
- **Linux:** `/home/[your_user_name]/Documents/csc101_assignments`

Within that folder, create a folder for this specific assignment: `01_hello_world`.

Note: We are purposefully using `_` instead of spaces here.

4.2. Start your terminal application

4.3. Navigate into the folder you just created using the `cd` command. It should look something like:

```
cd /home/[your_user_name]/Documents/csc101_assignments
cd 01_hello_world
```

4.4. Check your current path with the command `pwd`. It should be the path to `01_hello_world`.

4.5. Create a file called `HelloWorld.java` using the `touch` command in your terminal:

```
touch HelloWorld.java
```

4.6. Check if the file exists by running `ls`.

`HelloWorld.java` should be listed as the only file in that folder.

4.7. Open *HelloWorld.java* in VS Code:

```
code HelloWorld.java
```

On some systems this might be

```
vscode HelloWorld.java
```

4.8. From within VSCode, write the following content to that file:

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

4.9. Save the file.

Hint: You might get a warning that this is a non-project file. Just ignore it for now. You will learn about VS Code projects in the next assignment.

4.10. Return to your terminal to type some more commands.

4.11. Compile the source code into Java byte code:

```
javac HelloWorld.java
```

This will generate the file `HelloWorld.class`.

4.12. Verify that the class file exists in your folder by running the command `ls`.

4.13. Finally, run the code:

```
java HelloWorld
```

The terminal should print out `"Hello World!"`.

5. Hello Student

Let's go one step further and write a program that uses command line arguments. It will write a greeting to greets whatever name you want.

5.1. Using the terminal, create a copy of *HelloWorld.java*:

```
cp HelloWorld.java HelloStudent.java
```

5.2. Open *HelloStudent.java* in VS Code from the terminal.

5.3. Modify *HelloStudent.java* within VSCode to look like:

```
public class HelloStudent {
    public static void main(String[] args) {
        System.out.println("Hello " + args[0] + "!");
    }
}
```

5.4. Compile the code

5.5. Run the code and add your name to the command:

```
java HelloStudent YourName
```

The terminal will reply with `"Hello YourName!"`

6. Submission

You should be enrolled into this class in Gradescope. The link to Gradescope can be found in Content → Coding Assignments → Gradescope. If you cannot access this class's Gradescope please notify your instructor so they can manually add you.

To submit your assignment, please navigate to the *HelloWorld* assignment and upload the following files via the *Upload Submission* button:

- `HelloWorld.java`
- `HelloStudent.java`