CS102A Introduction to Computer Programming
Fall 2020 Lab 1

Objective

1. Install JDK and configure environment variable.
2. Learn compilation and execution of your first java program in command line.

1 Course Logistics

1.1 Lab Exercise

The content of lab exercise is prepared according to the knowledge introduced in lectures. Normally, we provide three to seven exercises per lab.

1.2 Assignment

The assignment is independent from the lab exercise and will be sent to you from the Online Judge system.

1.3 Project

Before the end of this semester, all of you should work on a project. The number of members for each group is two.

2 Software Installation

2.1 Install JDK

The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets. It includes a Java Runtime Environment (JRE), an interpreter/loader

(`java`), a compiler (`javac`), an archiver (`jar`), a documentation generator (`javadoc`) and other tools needed in Java development.

You can download and then install JDK from the following link: `https://www.oracle.com/java/technologies/javase-downloads.html`

> **Note**
> Or you can download the JDK from Sakai.

Once downloaded, run the executable. Follow the prompts to install the JDK and **do not change the default installation path**. Today's exercise is to make sure that the Java compiler (JDK) is correctly installed on your machine. Open a terminal window or command prompt then type the following command:

```
java -version
```

If the command returns text like the following, you have installed it successfully.

```
java version "1.8.0_261"
Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
```

> **What is command line?**
> The command line is a text interface for your computer. It's a program that takes in commands, which it passes on to the computer's operating system to run. From the command line, you can navigate through files and folders on your computer, just as you would with Windows Explorer on Windows or Finder on Mac OS. The difference is that the command line is fully text-based.
>
> - On Windows, press `win` and `R` key together. In the prompted window, type `cmd` and then press `Enter`.
> - On MacOS, search for *Terminal* application.

> **What are JVM, JRE, and JDK?**
> `https://www.geeksforgeeks.org/differences-jdk-jre-jvm/`

2.2 Java Environment Variables

If somehow your Windows returns

```
'java' is not recognized as an internal or external command,
operable program or batch file.
```

it means that your computer cannot find the JDK you just installed. In such a case, you need to set up the environment variables.

What is environment variable?

An environment variable is a dynamic-named value that can affect the way running programs will behave on a computer. The *PATH* environment variable is used by your operating system (e.g., Windows) to locate needed executables (e.g., `java`) from the command line.

1. Press `win` and `R` key together. In the prompted window, type `SystemPropertiesAdvanced` and then press `Enter`.
2. Click on the *Environment Variables...* button on the bottom right corner.
3. Double click the *Path* line in *System variables* groupbox.
4. Click on the *New* button, then paste the following string and press `Enter`:

   `C:\Program Files\Common Files\Oracle\Java\javapath`
5. Click *OK* buttons on all opened windows.

Then you need to close and re-open your command line prompt to validate if `java` is now available.

2.3 Editor

Any text editor can be used to program in Java. In this lab, *Visual Studio Code* is recommended: `https://code.visualstudio.com/`

1. Download and then install Visual Studio Code with the link above.
2. Open Visual Studio Code, click on the Extension icon on the left sidebar(typically the fifth one).
3. Type in `java` and click the *Install* green button of *Java Extension Pack*. Wait for the installation to complete.

3 Exercise

3.1 Compile a Java Program in Command Line

Create a project folder on your computer. Then open your text editor, create a new text file with the following code:

```java
public class Demo {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Then use the *Save As* command in the *File* menu to save the file with the name `Demo.java` in the folder. Note that the `.java` file name has to be the same as the main class name in your program.

> **Note**
> If you are using Visual Studio Code and a *Java Extension Guide* popped up, click the *Install All* button.

Subsequently, open the command line and use `cd` command to go to the directory where you save your `Demo.java`; use `javac` command to compile your `.java` file; use `java` command to execute the `.class` file (which must have a main function). The procedure is as follows (the `>` sign means that the corresponding line is a command to be executed):

```
C:\Users\James>cd "Desktop\lab 1"

C:\Users\James\Desktop\lab 1>javac Demo.java

C:\Users\James\Desktop\lab 1>java Demo
Hello, world!

C:\Users\James\Desktop\lab 1>
```

> **Note**
> When executing the compiled `.class` file, the `.class` extension must be removed from the command. Otherwise,
>
> ```
> C:\Users\James\Desktop\lab 1>java Demo.class
> Error: Could not find or load main class Demo.class
> Caused by: java.lang.ClassNotFoundException: Demo.class
> ```

Note

If you are using Visual Studio Code, you can click on the *Run* link above the second line of the file. In this case, you do not need to manually compile and run the file from command line.

Note

Alternatively, you can also use other software to combine the code editor and compiler. A popular option is the JetBrain IDEA: `https://www.jetbrains.com/idea/download/`.