

Lab Exercise #1
Compilation, Debugging, and Javadoc Generation with and without an IDE
CS 2334, Fall 2013

Due by: Friday, August 23, 2013, 2:30 pm CST

This lab is individual work. Each student must complete this assignment independently.

Name: _____

Learning Objectives (Milestones):

1. Install Java Development Kit.
2. Successfully compile and debug a sample Java program and generate Javadoc documentation for it *without using an integrated development environment (IDE)*.
3. Successfully compile and debug a sample Java program and generate Javadoc documentation for it *using the Eclipse IDE*.

Instructions:

This lab exercise *requires a laptop with an Internet connection*. Once you have completed the exercises in this document, you will submit it for grading. You should legibly write your name at the top of this lab handout.

1. Check to see if you already have the current version of the Java Development Kit (JDK) installed by opening a command or terminal window. (For Windows, click on *Start | Run* and then enter the text “cmd” and click “OK”; for Mac you can use the *Spotlight* to search for “Terminal” and click on the top hit.) If you have the correct version installed and your path set correctly the following commands should give the results listed below the commands.

```
>java -version
java version "1.7.0_25"
...

>javac -version
javac javac 1.7.0_25
...
```

If you already have the current JDK installed, skip to Step 3. Otherwise, proceed to Step 2.

2. Install the JDK. For Mac, refer to *MacJDKInstallSteps.pdf* on the class website. For Windows 8, refer to *JDK_Installation_Windows8.pdf*. For Windows 7, refer to *JDK_Installation_Instructions_Windows7.pdf*. Please note that the Windows 7 instructions are out of date since they have not been updated since Spring 2012. (You will download and install Java 7 update 25 even though the Windows 7 instructions depict installation of Java 6 update 23.) There are no instructions for Linux because if you’re running Linux, you can probably manage to install the latest JDK without detailed instructions.

Once you have the JDK installed, check your installation to ensure everything is correctly installed and your PATH environment variable is correctly set by opening a command/terminal window and typing the commands as shown in Step 1. If everything checks out, proceed to Step 3. Otherwise seek help in completing Step 2.

3. Download the sample “Lab1.java” source code from the class website. Save this file into a folder on your laptop where you will keep your CS 2334 Projects and Labs. Create a `cs2334` folder in the root of your laptop's hard drive (for Windows, this would be `C:\cs2334`) and then create a `lab1` folder inside the `cs2334` folder (`C:\cs2334\lab1`) and store these files in the `lab1` folder. You will also need to download the “docs.opt” file from the class website. Save it in the same folder as the `Lab1.java` file.

4. Add the following code to the main method of `Lab1.java` using Notepad or any text editor you desire but make sure you do not use a word processor, or any other program that does any markup/formatting at all. Notepad can be found in Windows under *Start | All Programs | Accessories*. **Do NOT use Eclipse or another IDE, either.**

```
Lab1 lab1Program;
```

```
lab1Program = new Lab1( "This is the first lab exercise for CS 2334." );
```

5. Compile the `Lab1.java` file from the command prompt with the following command (you need a command window/terminal to do this — **do NOT use Eclipse or another IDE**). You should receive several compilation errors that you need to identify and correct in the source code. List these errors in the space provided below with a short explanation of how you fixed them. Be sure to list the **line number** that the actual error occurred in the `Lab1.java` file. Once you have removed all of these errors, the file should compile without any errors or warnings.

```
javac Lab1.java
```

List the errors found and give a short explanation of how you fixed each one:

6. Run the `Lab1` program from the command prompt (again, without **Eclipse or another IDE**). This should result in one or more runtime errors. Fix the program source code and verify that the program works by running it again. The command required to run the program is:

```
java Lab1
```

List all runtime errors encountered and briefly explain the problems:

7. Generate Javadoc documentation for the `Lab1` program with the following command. (This must be done at a command prompt – **do NOT use Eclipse or another IDE.**)

```
javadoc @docs.opt *.java
```

This command will create a new sub-folder named `javadocs` and place several files in the folder including an `index.html` file. The output of the `javadoc` command should be similar to:

```
Creating destination directory: "javadocs\  
Loading source file Lab1.java...  
Constructing Javadoc information...  
...
```

Next, open the `javadoc\index.html` file using a web browser and inspect it's contents.

Describe the contents of the `index.html` file. **(Your description should contain at least 5 sentences to get full credit.)**

8. Download and install Eclipse. Instructions are in the “Eclipse_install_instructions” slides in the Lab 1 directory.
9. Download and save the sample “Lab1-eclipse.zip” source archive from the class website. Import the archive into Eclipse using the instructions given in the “Basic Eclipse Tutorial” slides. This is the first part of the slide set labeled “Lab2-slides” in the Lab 1 directory. (Note that this exercise was previously part of Lab 2, and the slide set has not been updated. Your Lab 2 will cover different material.) Once the archive has been imported, you may delete “Lab1-eclipse.zip” from your computer.

10. Add the following code to the main method of `Lab1.java` using Eclipse.

```
Lab1 lab1Program;  
  
lab1Program = new Lab1( "This is still the first lab exercise for CS 2334." );
```

11. Eclipse should highlight several errors that you need to correct in the source code. List these errors in the space provided below with a short explanation of how you fixed them. Be sure to list the line number that the actual error occurred on in the `Lab1.java` file. Once you have removed all of these errors, the file should not contain any errors or warnings.

16. Fix the error by changing ONLY the `work()` method in `Lab2.java`. Write down the modified `work()` method.

17. Using Eclipse debugger, what is the value of `myArray[4]` after `work()` is done?