**Assignment 0 – JDK Installations and Compile and Run Java Program from cmd**



**Installing JDK, Compile and Run Java Program Using command line (Mac Terminal)**

The purpose of this lab is to prepare you with the tools and software you need in order to be able to set the environment on your computer to successfully start working on your assignments. It is crucial that you complete this lab to save you time in order to focus on the course work and projects.

**Objectives**

* No need to install the JDK again if you can run “Hello World” in your Eclipse IDE
* Next step – be able to run javac in a Command window
* Installation of JDK (1.14 and above)
* For Windows: Set Environment Variables in Windows OS: Path and Classpath
* Test java and javac commands via CommandLine in Windows or Terminal in Mac OS
* Compile and Run Java program using cmd (command line)

**Step 1 – Install Java Development Kit (JDK/SDK)**

**1**. Download and install the latest JDK, which allows you to code and run Java programs (recommendation: Java SE 14.0.1). Use the following link to start the download:

<https://www.oracle.com/java/technologies/javase-downloads.html>

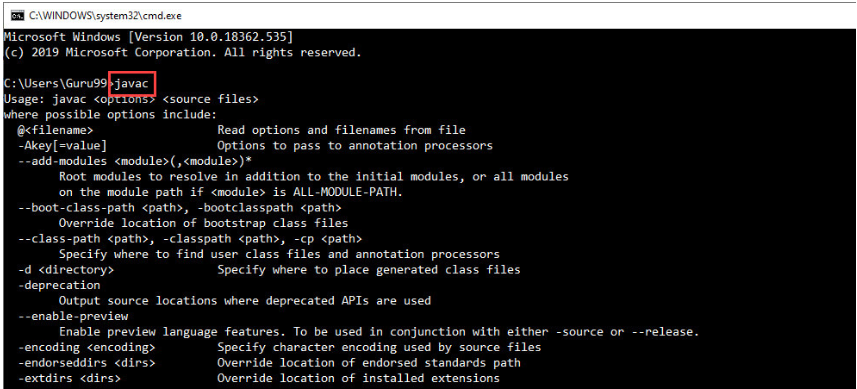
**2**. Accept License Agreement and download the latest JDK for your version of operating system (32-bit or 64-bit)

**3**. Once the download is completed, run the exe to install JDK, and click Next.

**4**. Select the PATH for Java installation and click Next

**5**. Once the installation is completed click Close.

**6**. To verify, go to command prompt by (Click on **Start** and type **cmd**). This will open a black screen that you can type. Type “javac”. If the result is similar to the following screen shot, Java is installed.



**7**. Set Environment Variables in Java: Path and Class path (**Only for Windows**)

1. Right Click on the **My Computer** and Select the **properties**
2. Select on **advanced system settings**
3. Click on **Environment** **Variables**
4. Click on **new** Button of **User variables**
5. Type **PATH** in the Variable name text box.
6. Copy the path of bin folder which is installed in JDK folder



1. Paste Path of bin folder in **Variable value** text box and click on OK Button.

**Note**: In case you already have a PATH variable created in your PC, edit the PATH variable to *PATH = <JDK installation directory>\bin;%PATH%;*

*(%PATH% appends the existing path variable to our new value)*

1. You can follow a similar process to set CLASSPATH.  
   

**Note:** In case your java installation does not work after installation, change classpath to

CLASSPATH = <JDK installation directory>\lib\tools.jar;

1. Click on OK button



1. Go to command prompt and type **javac** commands. If you see a screen similar to the following, Java is installed. 
2. Take a screen shot of the Command Line Screen and save it in a word document. Text

   Description automatically generated

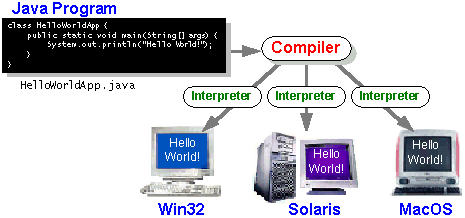
**Step 2 Compile and Run Java program from Command Line**

## “Hello Word” Java Program for Mac OS

<https://docs.oracle.com/javase/tutorial/getStarted/cupojava/unix.html>

## “Hello World!” Java Program for Microsoft Windows

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Creating Your First Application | |  | | --- | |  | | |  | | --- | |  | |  |
| Your first program, HelloWorldApp, will simply display the greeting "Hello world!". To create this program, you will:   * **Create a source file.** A source file contains text, written in the Java programming language, that you and other programmers can understand. You can use any text editor to create and edit source files.      * **Compile the source file into a bytecode file.** The *compiler*, javac, takes your source file and translates its text into instructions that the *Java Virtual Machine*(Java VM) can understand. The compiler converts these instructions into a bytecode file.      * **Run the program contained in the bytecode file.** The Java interpreter installed on your computer implements the Java VM. This interpreter takes your bytecode file and carries out the instructions by translating them into instructions that your computer can understand. |



## Create a Source File.

To create a source file, you have two options:

* You can save the file [HelloWorldApp.java (in a .java source file)](https://www.iitk.ac.in/esc101/05Aug/tutorial/getStarted/application/example/HelloWorldApp.java) on your computer and avoid a lot of typing. Then, you can go straight to [step b](https://www.iitk.ac.in/esc101/05Aug/tutorial/getStarted/cupojava/win32.html#2b).
* Or, you can follow these longer instructions:

**1.** Start NotePad. In a new document, type in the following code:

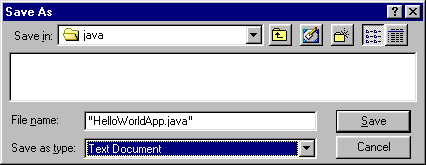
|  |
| --- |
| /\*\*  \* The HelloWorldApp class implements an application that  \* displays "Hello World!" to the standard output.  \*/  public class HelloWorldApp {  public static void main(String[] args) {  // Display "Hello World!"  System.out.println("Hello World!");  }  } |

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Be Careful When You Type** |  |  |   Type all code, commands, and file names exactly as shown. The Java compiler and interpreter are *case-sensitive*, so you must capitalize consistently.  HelloWorldApp  helloworldapp |

**2.**Save this code to a file. From the menu bar, select File > Save As. In the Save As dialog box:

* + Using the Save in drop-down menu, specify the folder (directory) where you'll save your file. In this example, the directory is java on the C drive.
  + In the File name text box, type "HelloWorldApp.java", including the double quotation marks.
  + From the Save as type drop-down menu, choose Text Document.

When you're finished, the dialog box should look like this:



Now click Save and exit NotePad.

## b. Compile the Source File.

From the Start menu, select the MS-DOS Prompt application (Windows) or Command Prompt application (Windows). When the application launches, it should look like this:



The prompt shows your *current directory*. When you bring up the prompt for Windows, your current directory is usually WINDOWS on your C drive (as shown above) or WINNT for Windows NT. To compile your source code file, change your current directory to the directory where your file is located. For example, if your source directory is java on the C drive, you would type the following command at the prompt and press Enter:

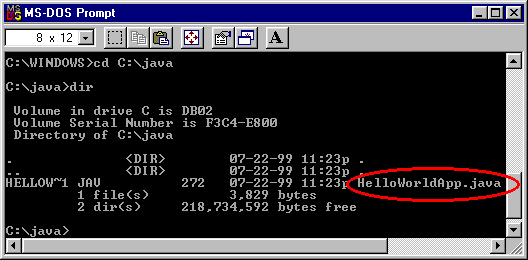
|  |
| --- |
| cd c:\java |

Now the prompt should change to C:\java>.

**Note:**To change to a directory on a different drive, you must type an extra command.

|  |  |
| --- | --- |
|  | As shown here, to change to the java directory on the D drive, you must reenter the drive, d: |

If you enter dir at the prompt, you should see your file.



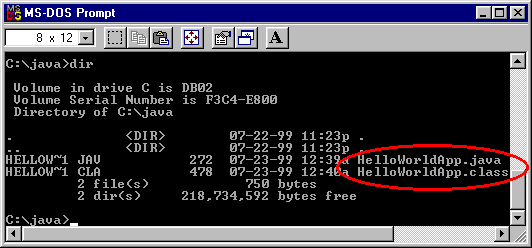
Now you can compile. At the prompt, type the following command and press Enter:

|  |
| --- |
| javac HelloWorldApp.java |

If your prompt reappears without error messages, congratulations. You have successfully compiled your program.

|  |
| --- |
| **Error Explanation**  **Bad command or file name** *(Windows)*  **The name specified is not recognized as an internal or external command, operable program or batch file** *(Windows)*  If you receive this error, Windows cannot find the Java compiler, javac.  Here's one way to tell Windows where to find javac. Suppose you installed the Java 2 Software Development Kit in C:\jdk1.4. At the prompt you would type the following command and press **Enter**:  C:\jdk1.4\bin\javac HelloWorldApp.java  **Note:**If you choose this option, each time you compile or run a program, you'll have to precede your javac and java commands with C:\jdk1.4\bin\. To avoid this extra typing, consult the section [*Update the PATH variable* (outside of the tutorial)](http://java.sun.com/j2se/1.4/install-windows.html#Environment)in the installation instructions. |

The compiler has generated a Java bytecode file, HelloWorldApp.class. At the prompt, type dir to see the new file that was generated:



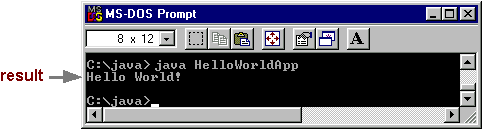
Now that you have a .class file, you can run your program.

## c. Run the Program.

In the same directory, enter at the prompt:

|  |
| --- |
| java HelloWorldApp |

Now you should see:



Congratulations! Your program works.

|  |
| --- |
| **Error Explanation**  **Exception in thread "main" java.lang.NoClassDefFoundError: HelloWorldApp**  If you receive this error, java cannot find your bytecode file, HelloWorldApp.class.  One of the places java tries to find your bytecode file is your current directory. So, if your bytecode file is in C:\java, you should change your current directory to that. To change your directory, type the following command at the prompt and press **Enter**:  cd c:\java  The prompt should change to C:\java>. If you enter dir at the prompt, you should see your .java and .class files. Now enter java HelloWorldApp again.  If you still have problems, you might have to change your CLASSPATH variable. To see if this is necessary, try "clobbering" the classpath with the following command:  set CLASSPATH=  Now enter java HelloWorldApp again. If the program works now, you'll have to change your CLASSPATH variable. For more information, consult the section [*5. Update the PATH variable* (outside of the tutorial)](http://java.sun.com/j2se/1.4/install-windows.html)in the installation instructions. |

Reference: <https://www.iitk.ac.in/esc101/05Aug/tutorial/getStarted/cupojava/win32.html>

<https://docs.oracle.com/javase/tutorial/getStarted/cupojava/win32.html>

Screen Shot that the Program was able to Compile and Run

Text

Description automatically generated

**Deliverables**

1. Upload the word document containing the screen shots from Step 1 and 2 to Blackboard, Assignment 0.
2. Upload the word document containing the screen shots from Step 1 and 2 (Assignment 0) to your GitHub account.