# How to set path in Java

The path is required to be set for using tools such as javac, java etc.
If you are saving the java source file inside the jdk/bin directory, path is not required to be set because all the tools will be available in the current directory.

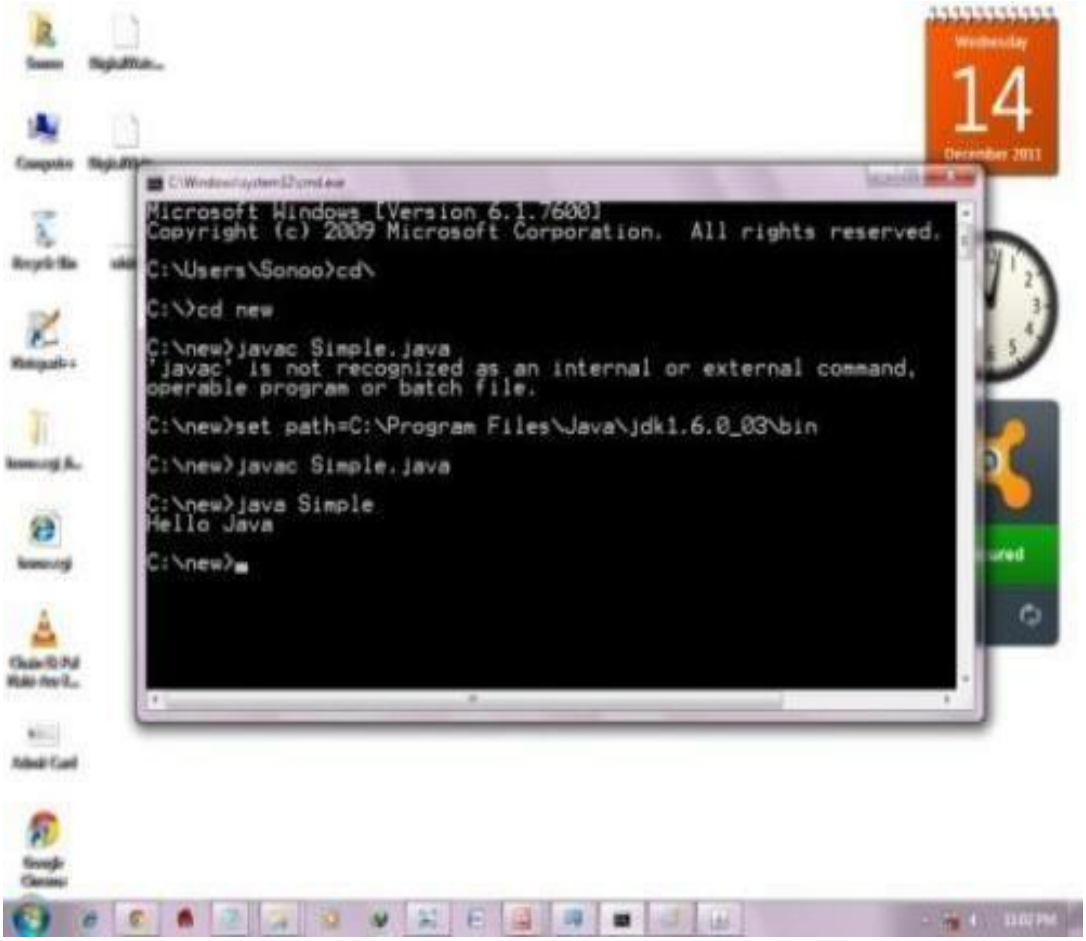
But If you are having your java file outside the jdk/bin folder, it is necessary to set path of JDK.
There are 2 ways to set java path:

1. temporary
2. permanent

1) How to set Temporary Path of JDK in Windows
To set the temporary path of JDK, you need to follow following steps:

- Open command prompt
- copy the path of jdk/bin directory
- write in command prompt: set path=copied_path

For Example:
set path=C:\Program Files\Java\jdk1.6.0_23\bin
Let's see it in the figure given below:



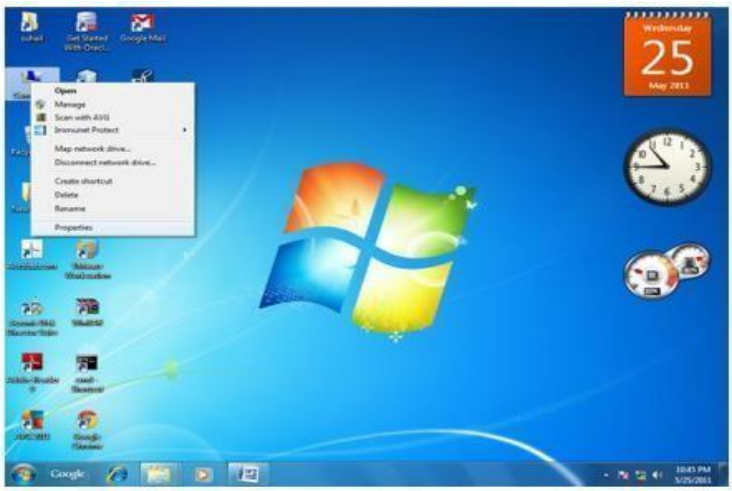

2) How to set Permanent Path of JDK in Windows

For setting the permanent path of JDK, you need to follow these steps:

- Go to MyComputer properties -> advanced tab -> environment variables -> new tab of user variable -> write path in variable name -> write path of bin folder in variable value -> ok -> ok -> ok

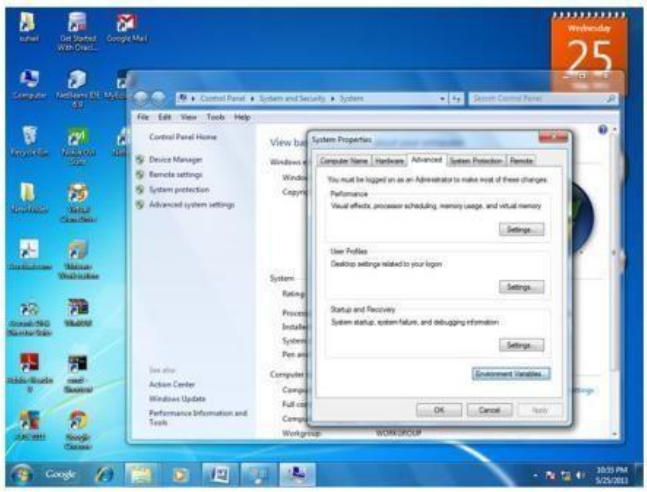
For Example:

1) Go to MyComputer properties

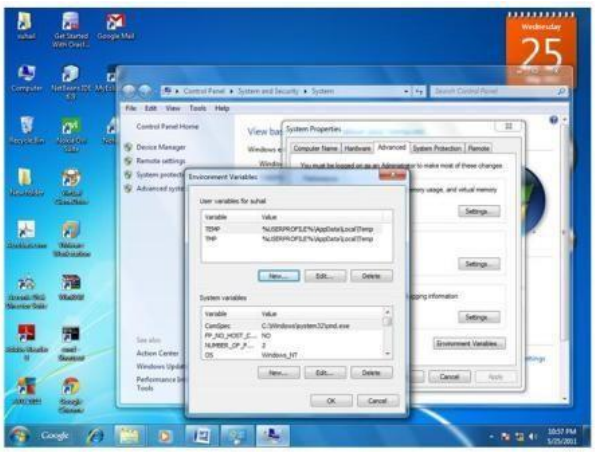


2) click on advanced tab

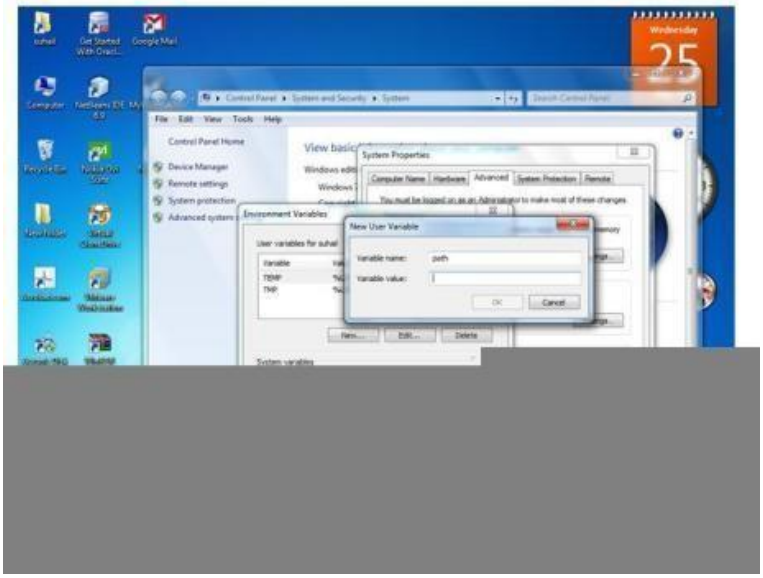
3) click on environment variables



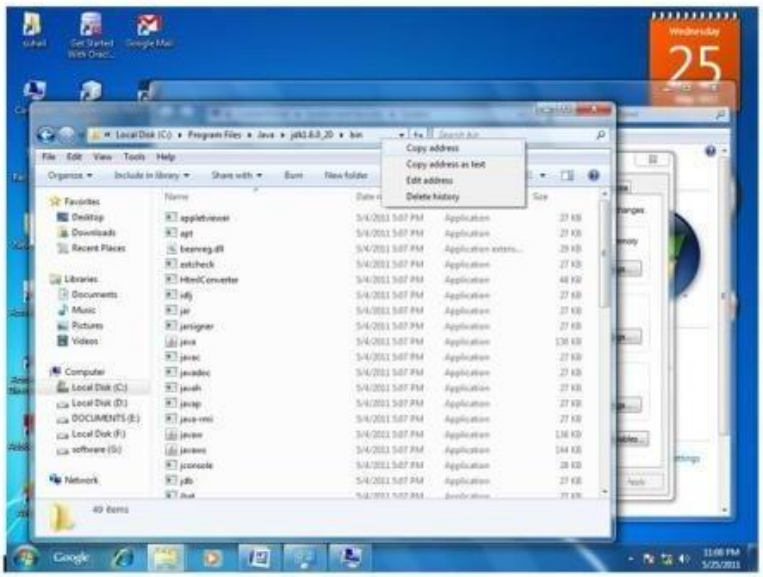
4) click on new tab of user variables



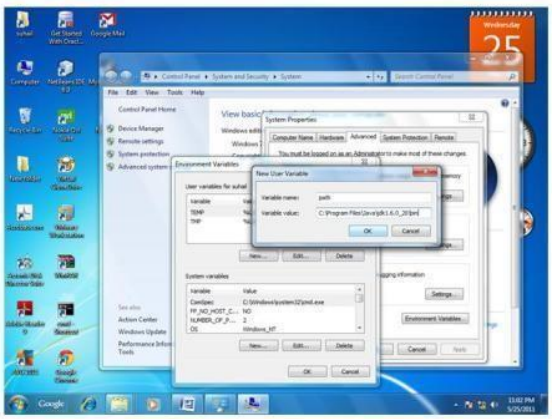
5) write path in variable name



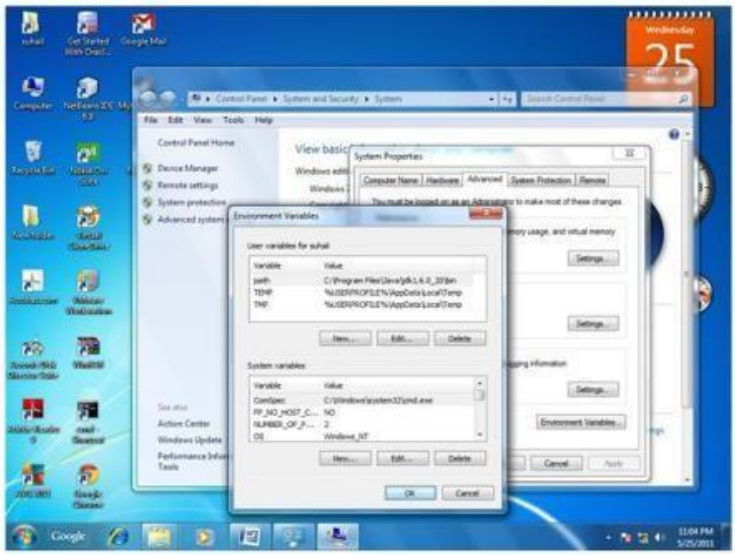
6) Copy the path of bin folder



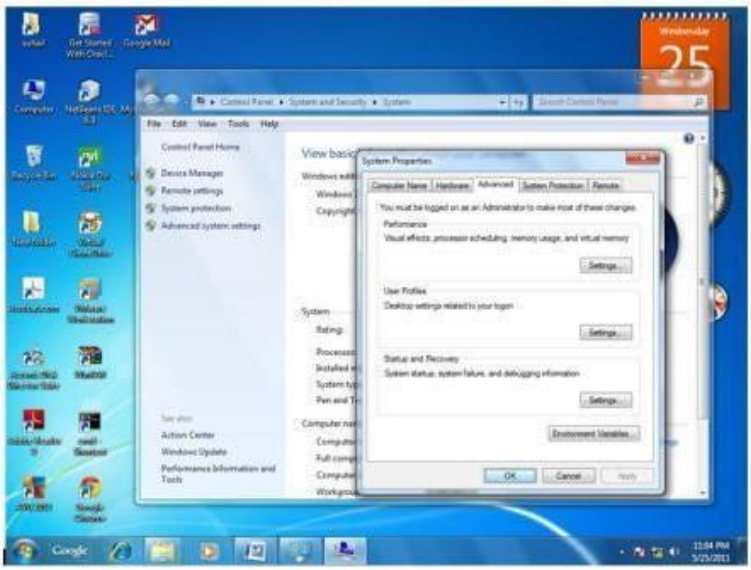
7) paste path of bin folder in variable value



8) click on ok button



9) click on ok button



Now your permanent path is set. You can now execute any program of java from any drive.

## Difference between path and classpath in Java

Path

Path variable is set for providing path for all Java tools like java, javac, javap, javah, jar, appletviewer. In Java to run any program we use java tool and for compile Java code use javac tool. All these tools are available in bin folder so we set path upto bin folder.

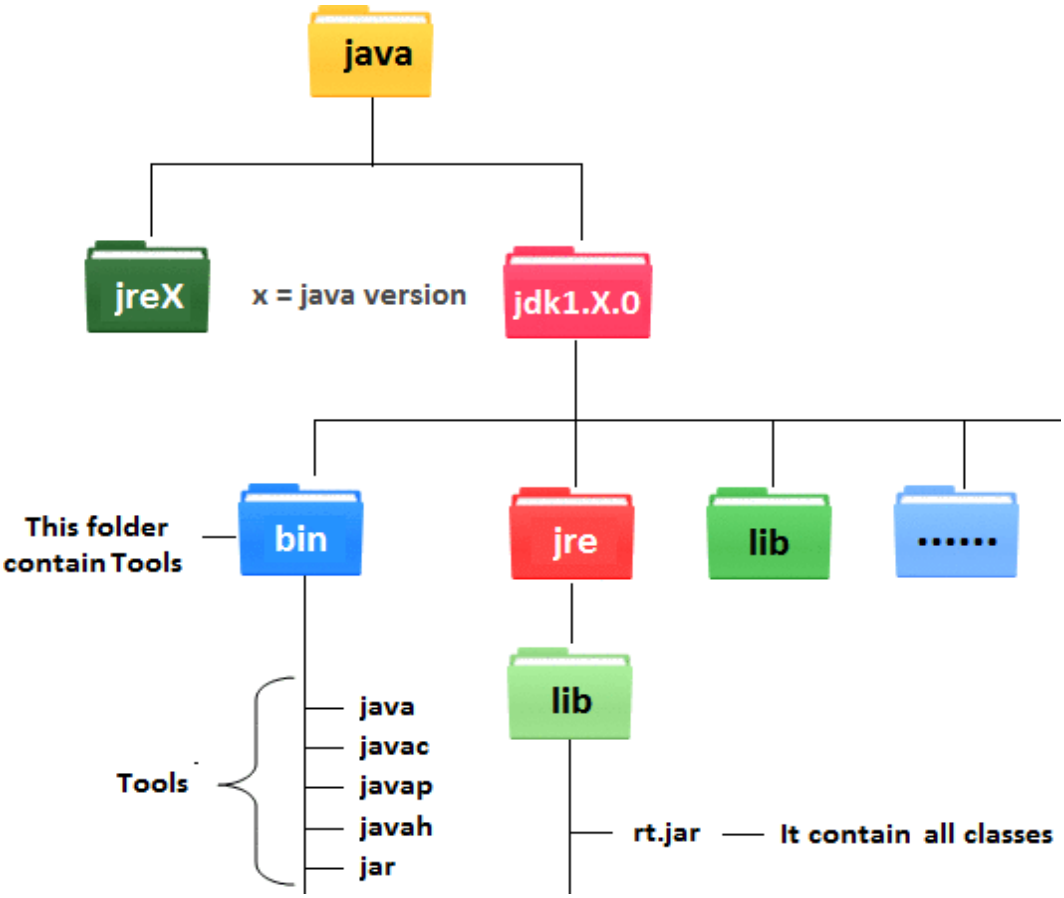
classpath

classpath variable is set for providing path of all Java classes which is used in our application. All classes are available in lib/rt.jar so we set classpath upto lib/rt.jar.

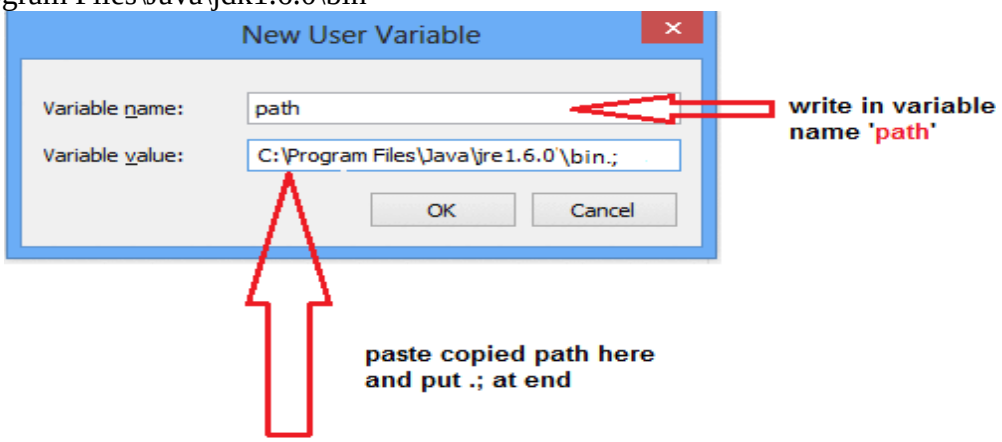
Difference between path and classPath

| path | classpath |
|---|---|
| path variable is set for providing path for all java tools like java, javac, javap, javah, jar, appletviewer | classpath variable is set for provide path of all java classes which is used in our application. |

JDK Folder Hierarchy

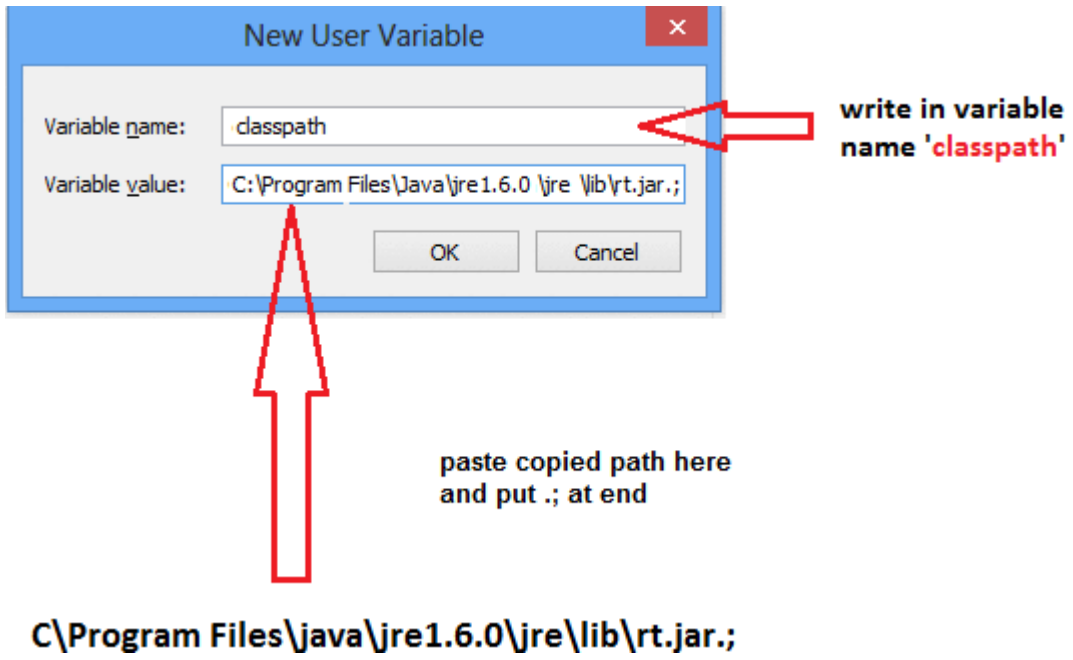



Path variable is set for use all the tools like java, javac, javap, javah, jar, appletviewer etc.
Example
"C:\Program Files\Java\jdk1.6.0\bin"




**C\Program Files\java\jre\1.6.0\bin.;**

All the tools are present in bin folder so we set path upto bin folder.
Classpath variable is used to set the path for all classes which is used in our program so we set classpath upto rj.jar. in rt.jar file all the .class files are present. When we decompressed rt.jar file we get all .class files.
Example
"C:\Program Files\Java\jre1.6.0\jre\lib\rt.jar"
In above rt.jar is a jar file where all the .class files are present so we set the classpath upto rt.jar.




First Java Program

Requirements for java Program

For executing any java program we need given things.

- Install the JDK any version if you don't have installed it.
- Set path of the jdk/bin directory.
- Create the java program
- Compile and run the java program

Steps For compiling and executing the java program

Java is very simple programming language first we write a java program and save it with program class name.

In below program we create a java program with "First" name so we save this program with "First.java" file name. We can save our java program anywhere in our system or computer.

Create First program

Example

```
Class First
{
public static void main(String[] args)
{
System.out.println("Hello Java");
System.out.println("My First Java Program");
}
}
```

Compile and Execute Java Code

To compile:  javac First.java

To execute: java First

Output

Hello Java

My First Java Program

Save Java Program

Syntax:

Filename.javc

Example:

First.java

Compile Java Program

Syntax:

Javac Filename.java

Example:

Javac First.java

Note: Here Javac and Java are called tools or application programs or exe files developed by sun micro system and supply as a part of jdk 1.5/1.6/1.7 in bin folder. Javac is used for compile the java program and java is used for run the java program.

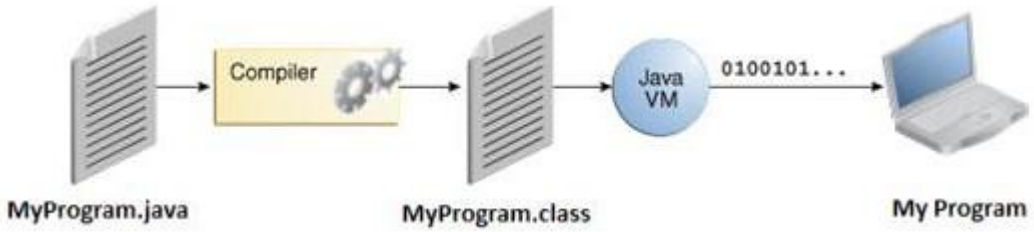
During the program execution internally following steps will be occurs.

- Class loader subsystem loads or transfer the specified class into main memory(RAM) from secondary memory(hard disk)
- JVM takes the loaded class
- JVM looks for main method because each and every java program start executing from main() method.
- Since main() method of java is static in nature, JVM call the main() method with respect to loaded class (Example: First as First.main(--))

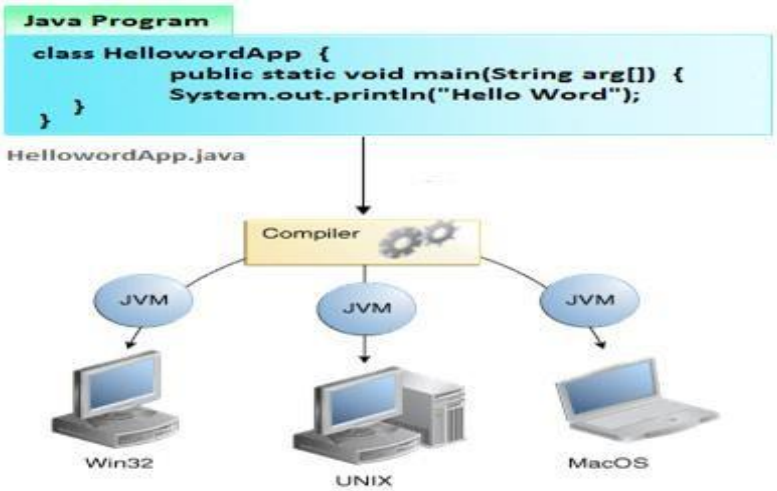
Note: A java program can contain any number of main method but JVM start execution from that main() method which is taking array of object of String class.

Compile and Run Java Program

In the Java programming language, all source code is first written in plain text files and save with the .java extension. After compilation, .class files are generated by javac compiler. A .class file does not contain code that is native to your processor; it instead contains bytecode (it is machine language of the Java Virtual Machine (JVM)).




The java launcher tool then runs your application with an instance of the Java Virtual Machine (JVM).




Steps for compile Java Program

- First Save Java program with same as class name with .java extension.
  Example: Sum.java
- Compile: javac Filename.java
  Example, javac Sum.java

Note: Here javac is tools or application programs or exe files which is used for Compile the Java program.

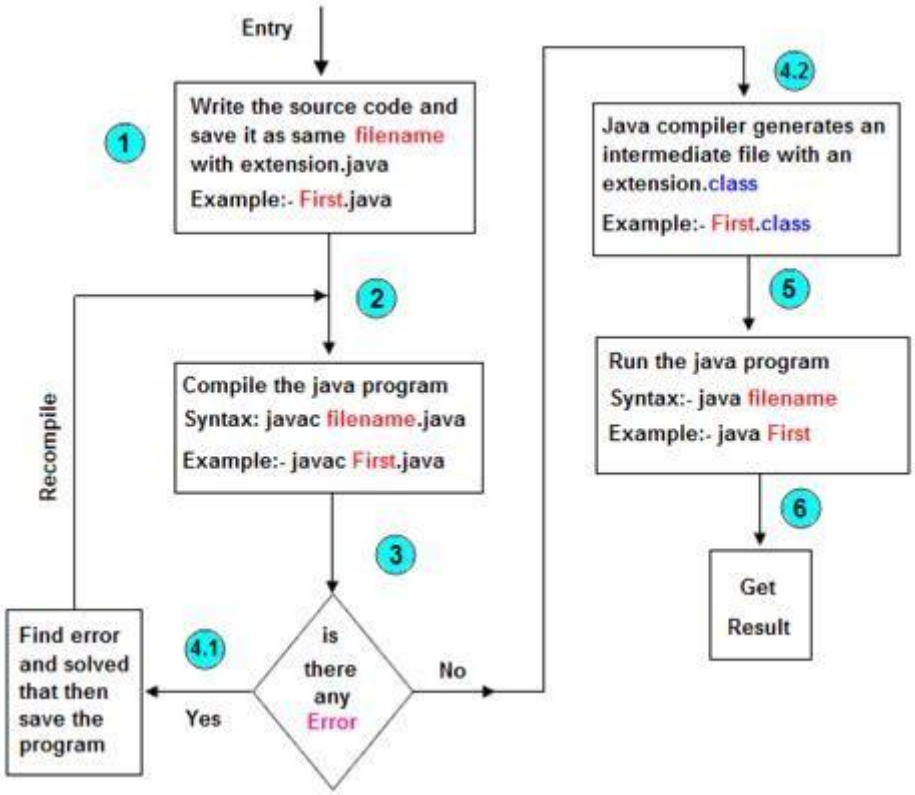
Steps for Run Java Program

- For run java program use java tool.
- Run by: java Filename
  Example: java sum

Note: Here java is tools or application programs or exe files which is used for run the Java program.

Steps For compiling and executing the java program

The following sequence of steps represented in the diagram use compiling the java program and executing the java programs.




In the above diagram javac and java are called tools or application programs or exe files developed by sun micro system and supply as a part of jdk 1.5/1.6/1.7 in bin folder (starting directory of java is called java home directory)