

Core Java



CONTENTS AT A GLANCE

Chapter 1: Introduction to Java

Topic 1: Evolution of java & Java in real time areas

Topic 2: Java installation in Windows

Topic 3: Recommended java IDE's

Chapter 2: Features of java

Chapter 3: A Simple hello world Program in java

Topic 1: Java keywords

Topic 2: Different types of comments in java

1. Single line comment
2. Multiline comments
3. Doc type comments

Topic 3: Internal structure of JVM

Chapter 4: Test



Chapter 1: Introduction

Before heading to learn java, let's see the brief introduction about java. Java is a high level programming language developed by Sun Microsystems. Sun announced java in 1995. Java is platform independent - it runs on different platforms like Windows, Mac, Linux, etc. There is a popular phrase in java "Write Once Run anywhere". We can write program in one platform and run that program on any different platform.

Topic 1: Evolution of java

Java EE: Past & Present

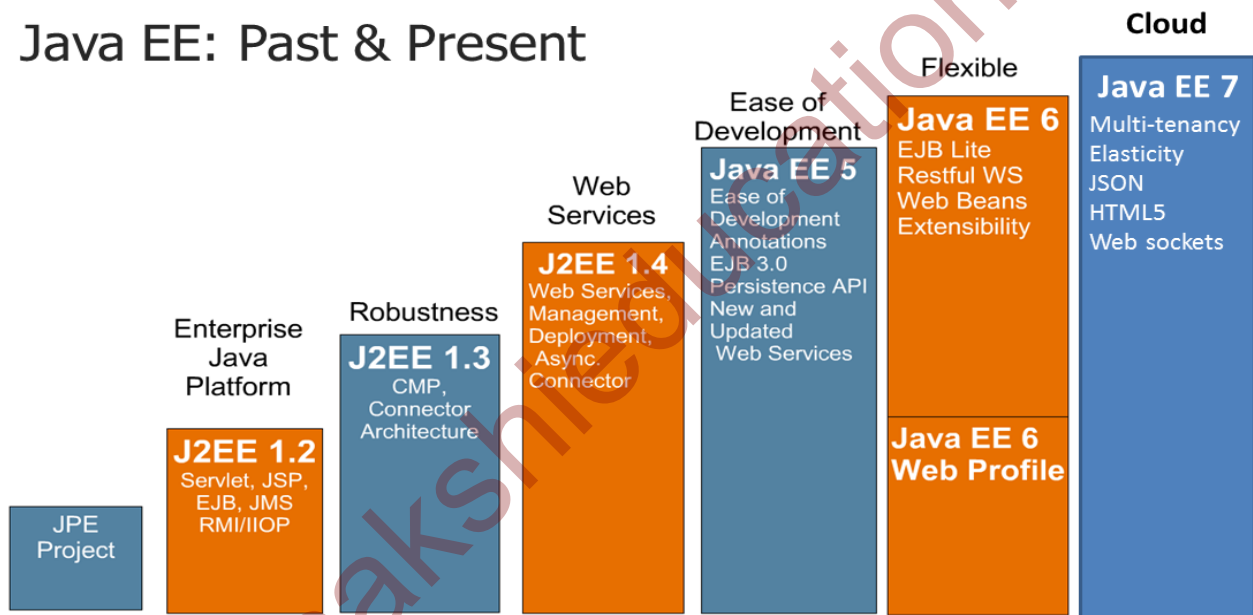


Fig 1.0 Java evaluation

Java in real-time:

- A. Desktop applications
- B. Mobile Applications
- C. Embedded systems
- D. Web applications
- E. Industrial robot control
- F. Mobile OS development
- G. Game development

Topic 2: Java installation in windows

Step 1:

Click on the given below URL to open the java download page in browser

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Step 2:

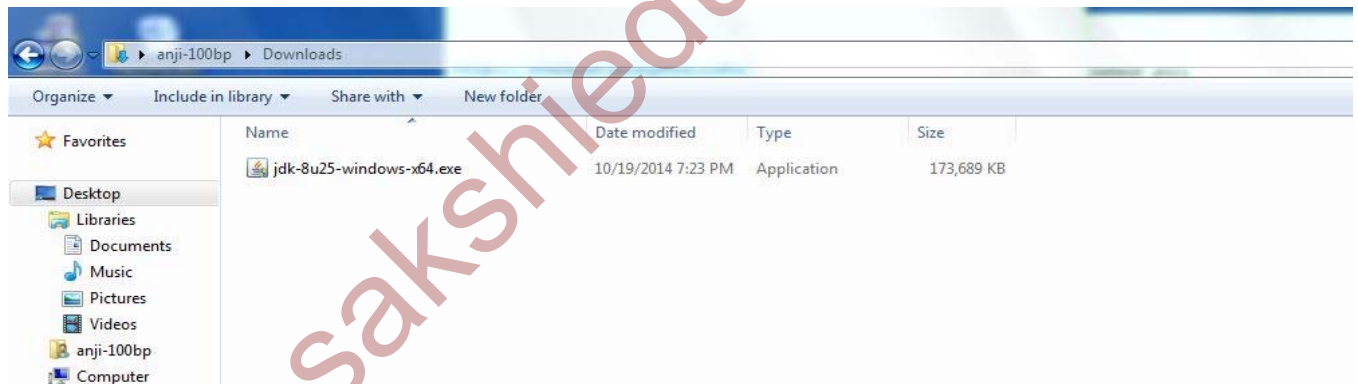
Select the file according to the operating system (32 bit / 64bit) to get the latest versions of java.

Step 3:

Close all the applications including the browser before installing the java in computer.

Step 4:

Double click on the file to start the java installation.

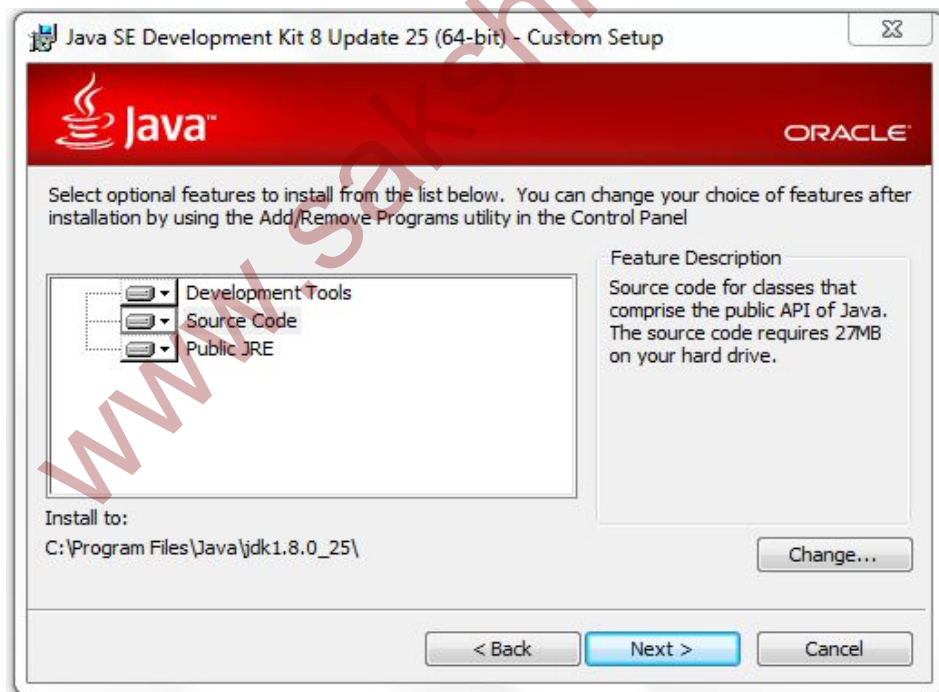


Step 5:

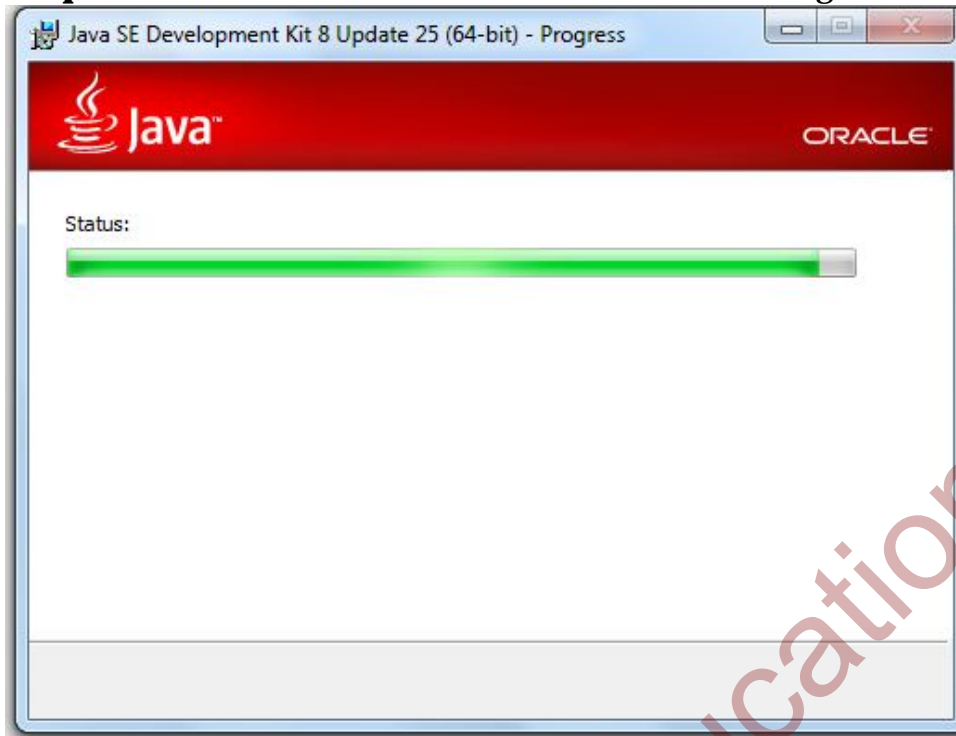
Welcome screen will appear as below now java installation has started.



Step 6: Follow the highlighted next button



Step 7: Installation status will be shown in this dialog box



Step 8: usage of java will be shown in this dialog box



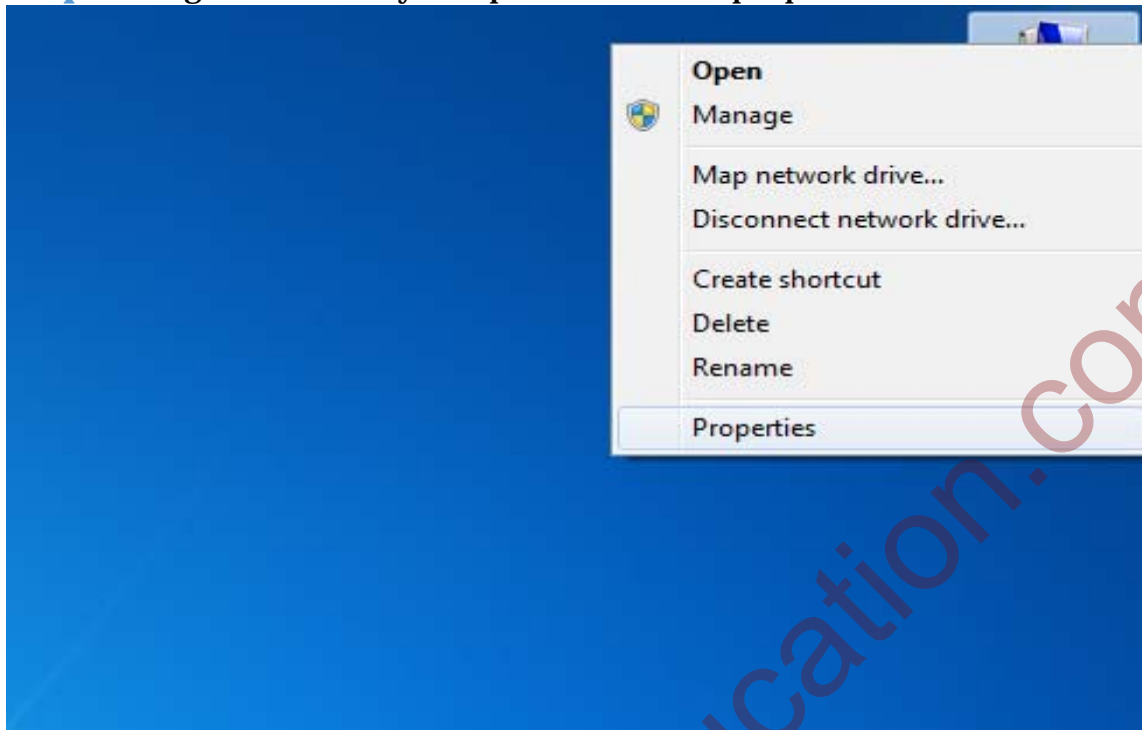
Step 9: Java installation completed successfully



Step 10: We can change the destination folder

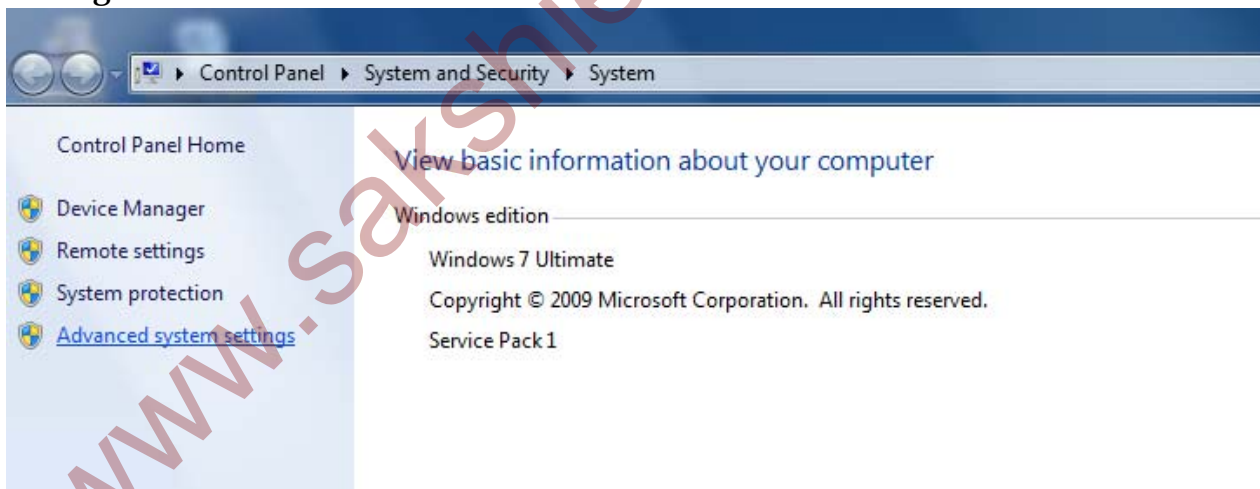


Step 11: Right click on my computer select the properties

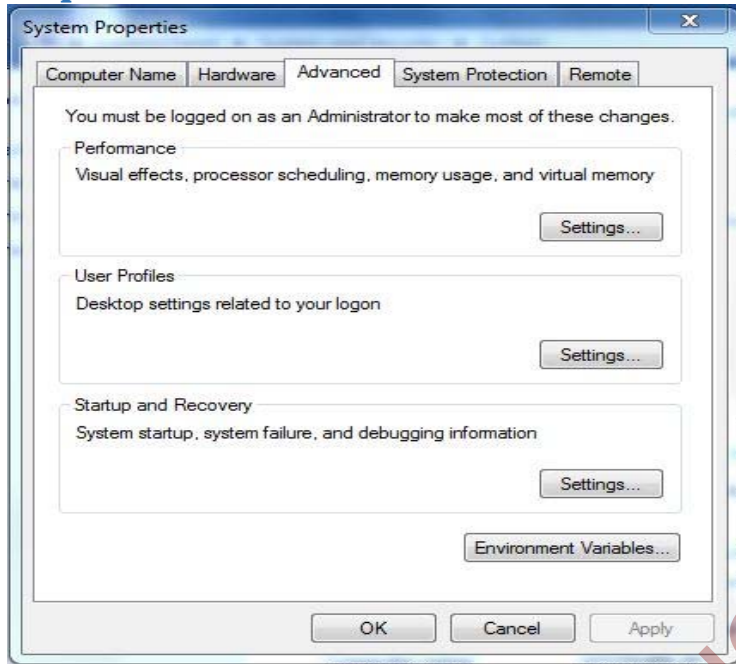


Step 12:

Pop up screen will be shown. Click on the left side highlighted Advanced System settings link

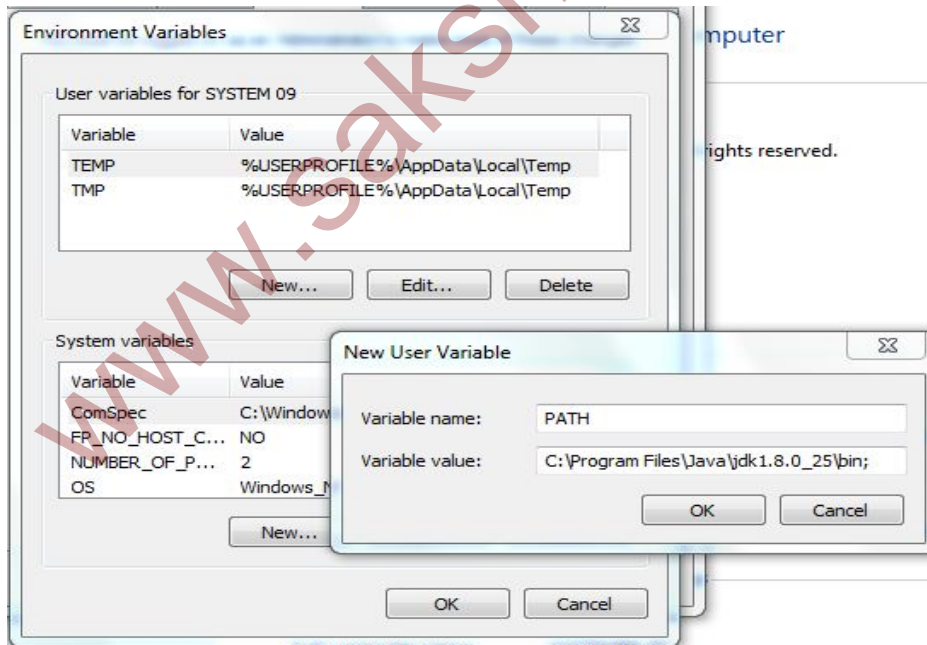


Step 13: Click on the Environmental Variables button

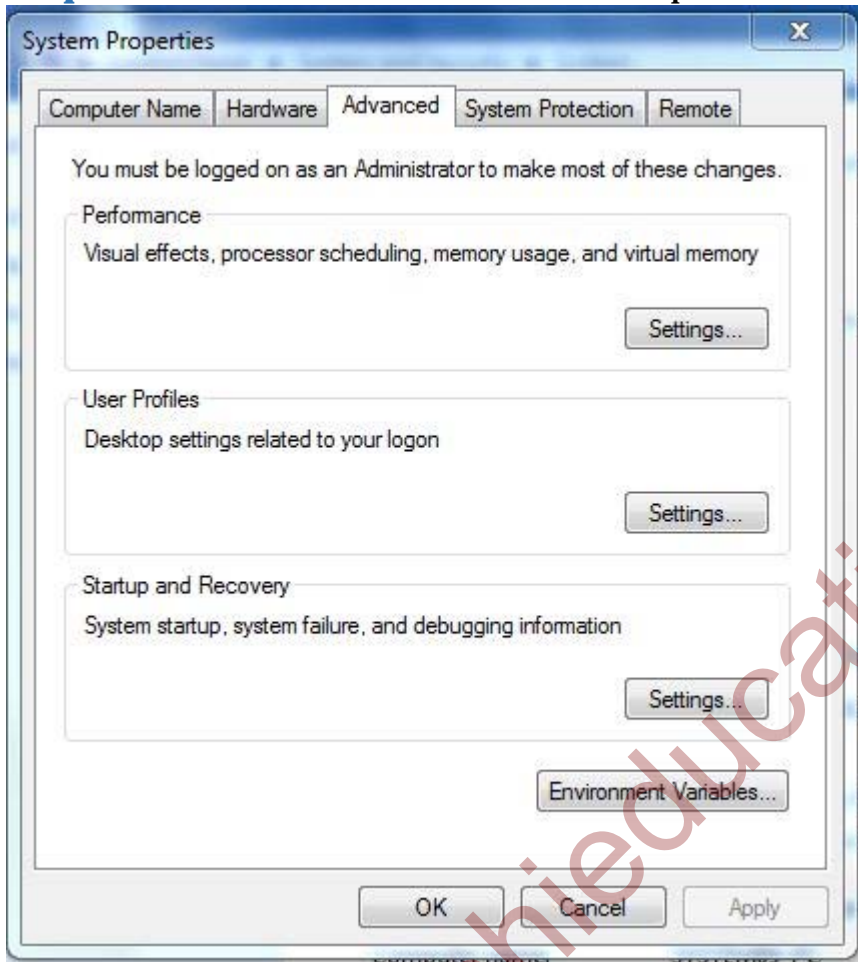


Step 14: Below screen will be appeared if we click on environmental variables button. Click on the new button in User Variables section a popup will be appear as shown in below figure. Write variable name as "PATH". In second text box write java JDK path shown in path

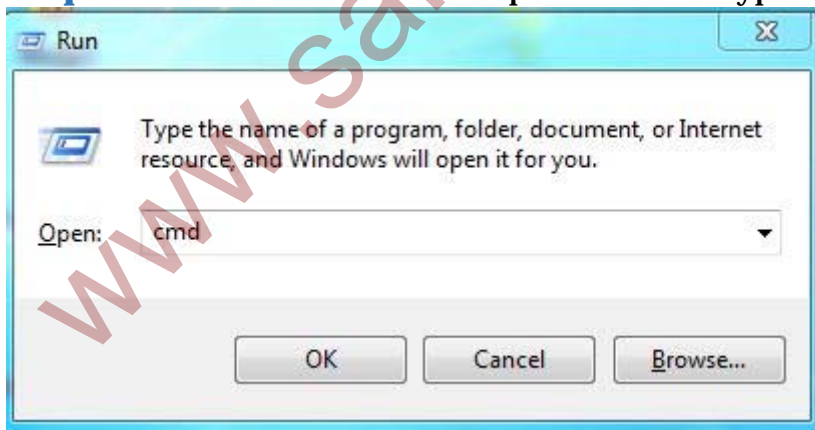
For path selection: C:\Program Files\Java\jdk1.8.0_25\bin



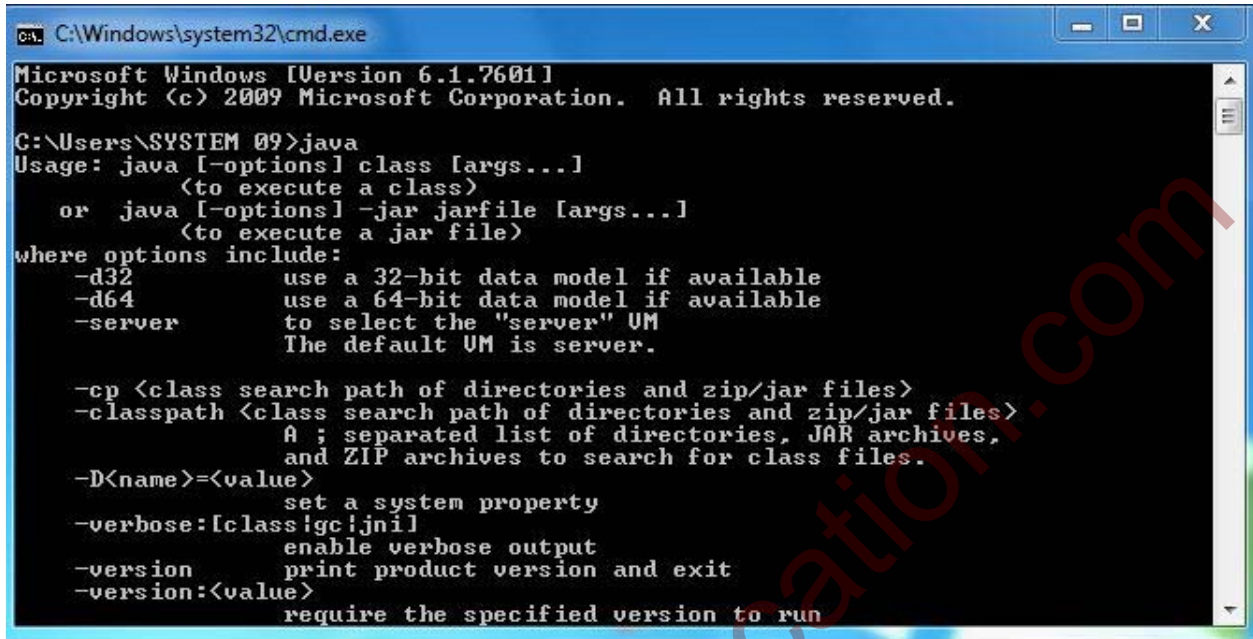
Step 15: Click on the ok button to save the path



Step 16: Press Windows + r to open RUN and type CMD



Step 17: Type java and press enter, result will be like below screen



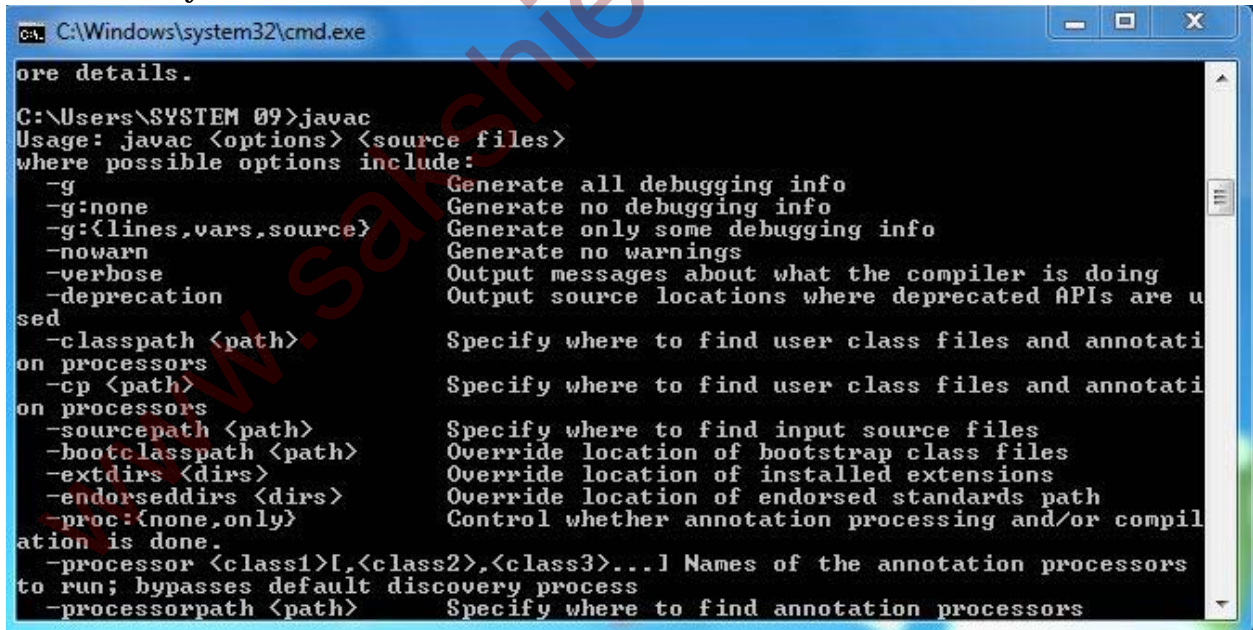
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\SYSTEM 09>java
Usage: java [-options] class [args...]
           (to execute a class)
   or java [-options] -jar jarfile [args...]
           (to execute a jar file)
where options include:
-d32                use a 32-bit data model if available
-d64                use a 64-bit data model if available
-server            to select the "server" VM
                   The default VM is server.

-cp <class search path of directories and zip/jar files>
-classpath <class search path of directories and zip/jar files>
                 A ; separated list of directories, JAR archives,
                 and ZIP archives to search for class files.
-D<name>=<value>   set a system property
-verbose:[class|gc|jni]
                   enable verbose output
-version           print product version and exit
-version:<value>   require the specified version to run
```

Step 18:

Type javac and press enter it will be visible like below screen - javac successfully installed in your machine



```
C:\Windows\system32\cmd.exe
ore details.

C:\Users\SYSTEM 09>javac
Usage: javac <options> <source files>
where possible options include:
-g                Generate all debugging info
-g:none          Generate no debugging info
-g:<lines,vars,source>
                 Generate only some debugging info
-nowarn          Generate no warnings
-verbose         Output messages about what the compiler is doing
-deprecation     Output source locations where deprecated APIs are used
-classpath <path>
                 Specify where to find user class files and annotations processors
-cp <path>       Specify where to find user class files and annotations processors
-sourcepath <path>
                 Specify where to find input source files
-bootclasspath <path>
                 Override location of bootstrap class files
-extdirs <dirs>  Override location of installed extensions
-endsdirs <dirs>
                 Override location of endorsed standards path
-proc:<none,only>
                 Control whether annotation processing and/or compilation is done.
-processor <class1>[,<class2>,<class3>...]
                 Names of the annotation processors to run; bypasses default discovery process
-processorpath <path>
                 Specify where to find annotation processors
```

Topic 3: Recommended java IDE's



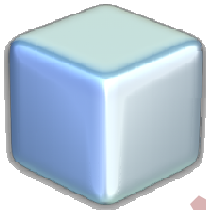
Eclipse is an open source java recommended IDE that will generate auto suggestions. Eclipse is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment. Written mostly in Java, Eclipse can be used to develop applications. By means of various plug-ins, Eclipse may also be used to develop applications in other programming languages: Ada, ABAP, C, C++, COBOL, FORTRAN, Haskell, JavaScript, Lasso, Natural, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others.

Download Link:

In this link you will be shown updates latest eclipse ide's

<https://eclipse.org/downloads/>

Netbeans



NetBeans is an integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP, C/C++, and HTML5. It is also an application platform framework for Java desktop applications and others.

The NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM. The NetBeans Platform allows applications to be developed from a set of modular software components called modules. Applications based on the NetBeans

Platform (including the NetBeans IDE itself) can be extended by third party developers.

Download Link:

<https://netbeans.org/downloads/>

IntelliJ IDEA



IntelliJ IDEA 13.1, the Java IDE with the fastest-growing mindshare, includes support for Java 8 and Java EE 7, new tools for Android development, the editor enhancements, and refined Gradle integration.

Download Link:

<http://www.jetbrains.com/idea/download/>

Chapter 2: Features of java

1. Simple
2. Secure
3. robust
4. Object-Oriented
5. Platform independent
6. Architecture neutral
7. Portable
8. Interpreted
9. Multithreaded
10. High Performance
11. Dynamic
12. Distributed

Simple

1. Java is simple easy to learn and easy to write
2. Java syntax mostly derived from C++
3. Complex concepts from C++ are removed from java
 - a. Ex: Pointers, Operator overloading.

Secure

1. Java is not only simple but also provides high lever security to the code
2. It prevents from creating the harmful code.
3. If we compile the java code a byte code means machine understandable code will be generated.
4. When we transferring the code from one system to another system machine code will be transferred.
5. This machine code will be executed inside the JVM. Apart from JVM we cannot execute the java code.

Robust

Java code is robust. It provides error free coding and java is mainly robust for its memory management, automatic garbage collection. Exception handling is one of the key points to make java robust.

Object oriented

We can say java is purely object oriented programming because we use objects and classes. Everything in java will be written in terms of object and classes.

Oops concepts are

1. Inheritance
2. Abstraction
3. Polymorphism
4. Encapsulation

Platform Independent

Java is fully platform independent why because there is a famous phrase in java "Write once Run Any Where". We can write java code on one platform and we can run that code on any other platform.

Ex: If we write java program in windows and we can run that program in Linux, mac and Ubuntu etc...

Architecture neutral

Java architecture is neutral there is no dependent features it is machine independent.

Portable

1. Java is portable because if its machine independent we can run java byte code anywhere if there is JVM.
2. Java code can be run on any platform regardless of its operating system.

Interpreted

1. Java is both interpreted and compiled language. Java compiler converts java code into byte code which is machine understandable code.

2. JVM will interpret that byte code regardless of any platform (Windows, Linux, and Mac).

Multithreaded:

1. Multithreading is the main feature of java
2. By using multithreading concept we can do several tasks in concurrent time.
3. Multithreading shares same memory

High Performance:

1. Java provides high performance in terms of code execution because java code can be converted into native byte code (machine understandable code)
2. By using Just-in-Time (JIT) compiler also java provides high performance.

Dynamic:

1. By using oops java provides dynamic code
2. In oops there is an inheritance concept in which we can use reusable code by this java got dynamic feature also.

Distributed:

Java mainly developed for internet. This means java programs run on internet in which it will transfer from one network to another network.

By using java we connect to remote URLs and we can access the data in that url.

Chapter 3: A Simple hello world Program in java

Explanation:

1. **Class** is a keyword used to point a class in java
2. **public static void main**
3. **public** it's the access specifier means from everywhere we can access it
4. **static** is a keyword by using static keyword we can call the method directly without creating the object. JVM starts its execution from the main method so its calls directly main method without creating the object
5. **void** is method it doesn't return any value
6. **main** is method name, JVM start execution form this method.

Topic 1: Java keywords

These are the reserved keywords in java we cannot modify those keywords.

Abstract, default, if, package, synchronized, assert, do, implements, private, this, Boolean, double, import, protected, throw, break, else, instanceof, public, throws, byte, extends, int, return, transient, case, false, interface, short, true, catch, final, long, static, try, char, finally, native, strictfp, void, class, float, new, super, volatile, const, for, null, switch, while, continue, goto.

Topic 2: different types of comments in java

Single line comments

```
A=10;  
B=20;  
If(a < b)
```

```
{ // block starts here
```



```
System.out.println("b is greater than a ");    //single  
line comments  
} // block ends here
```

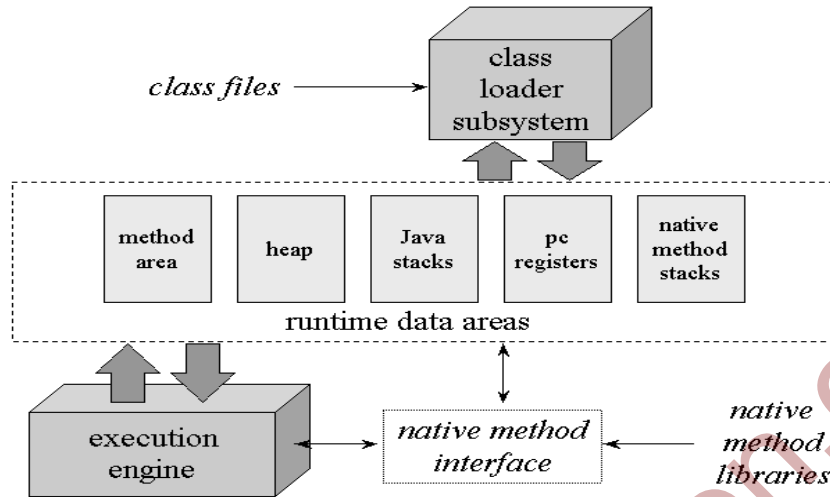
Multiline comments

```
packagesakshi;  
// multiline comments starts here  
// SakshiEducation is the class name  
public class SakshiEducation {  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        System.out.println("Hello Sakshi Education");  
    }  
}
```

Doctype comments:

```
/**  
 * This program will display hello world as standard output  
 *  
 * @author Ramanjaneya  
 * @version 1.0  
 */  
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

Topic 3: Internal structure of JVM



Chapter 4: Test

1. Which method is used for cloning the object ?
 - a. notify()
 - b. hashCode()
 - c. clone()
 - d. getClass()

Ans: clone()

2. Algorithm used by Garbage collector/
 - a. Sweep
 - b. Mark
 - c. a & b
 - d. None

Ans: Both a and b

3. Java is platform independent?
 - a. Yes
 - b. No

Ans: yes

4. JVM Is platform independent?

- a. Yes
- b. No

Ans: b

5. JAR Stands for?
- a. Java Archive
 - b. Java application Runner
 - c. Java Archive Runner
 - d. None

Ans: Java Archive

6. What is the byte code extension in java?
- a. .class
 - b. .java
 - c. Javac
 - d. None

Ans : .class

7. Java is strongly typed language?
- a. Yes
 - b. No

Ans: Yes

8. Java code is robust?
- a. Yes
 - b. No

Ans: yes

9. Java is architecture neutral
- a. Yes
 - b. No

Ans: yes

10. Java is object oriented programming language?
- a. Yes
 - b. No

Ans: yes