DEPARTMENT OF COMMERCE (CA)
JAVA PROGRAMMING AND HTML (Semester-IV)
II M.COM (CA) Sub Code-18MCC42C
UNIT - I

History of java-Java Features- Java and internet- Java and www- Java language: Introduction – Simple java program structures - JVM.

HISTORY OF JAVA:

James Gosling initiated the Java language project in June 1991 for use in one of his many set-top box projects. The language, initially called 'Oak' after an oak tree that stood outside Gosling's office, also went by the name 'Green' and ended up later being renamed as Java, from a list of random words.

Sun released the first public implementation as Java 1.0 in 1995. It promised Write Once, Run Anywhere (WORA), providing no-cost run-times on popular platforms.

On 13 November, 2006, Sun released much of Java as free and open-source software under the terms of the GNU General Public License (GPL).

On 8 May, 2007, Sun finished the process, making all of Java's core code free and open-source, aside from a small portion of code to which Sun did not hold the copyright.

Tools Needed

Pentium 200-MHz computer with a minimum of 64 MB of RAM (128 MB of RAM recommended).

The following software also needed

- Linux 7.1 or Windows xp/7/8 operating system
- Java JDK 8
- Microsoft Notepad or any other text editor

Java was originally designed for interactive television, but it was too advanced technology for the digital cable television industry at the time. However, it was suited for internet programming. Later, Java technology was incorporated by Netscape.

Java Version History

Many java versions have been released till now. The current stable release of Java is Java SE

10.

1. JDK Alpha and Beta (1995)
2. JDK 1.0 (23rd Jan 1996)
3. JDK 1.1 (19th Feb 1997)

4. J2SE 1.2 (8th Dec 1998)
5. J2SE 1.3 (8th May 2000)
6. J2SE 1.4 (6th Feb 2002)
7. J2SE 5.0 (30th Sep 2004)
8. Java SE 6 (11th Dec 2006)
9. Java SE 7 (28th July 2011)
10. Java SE 8 (18th Mar 2014)
11. Java SE 9 (21st Sep 2017)
12. Java SE 10 (20th Mar 2018)

JAVA FEATURES:

Object Oriented:
In Java, everything is an Object. Java can be easily extended since it is based on the Object model.

Platform Independent:
Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into a platform specific machine, rather into platform-independent bytecode. This byte code is distributed over the web and interpreted by the Virtual Machine (JVM) on whichever platform it is being run on.

Simple:
Java is designed to be easy to learn. If you understand the basic concept of OOP Java, it would be easy to master.

Secure:
With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.

Architecture-neutral:
Java compiler generates an architecture-neutral object file format, which makes the compiled code executable on many processors, with the presence of Java runtime system.

Portable:
Being architecture-neutral and having no implementation dependent aspects of the specification makes Java portable. The compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.

Robust:
Java makes an effort to eliminate error-prone situations by emphasizing mainly on compile time error checking and runtime checking.

Multithreaded:
With Java's multithreaded feature it is possible to write programs that can perform many tasks simultaneously. This design feature allows the developers to construct interactive applications that can run smoothly.

Interpreted:
Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light-weight process.

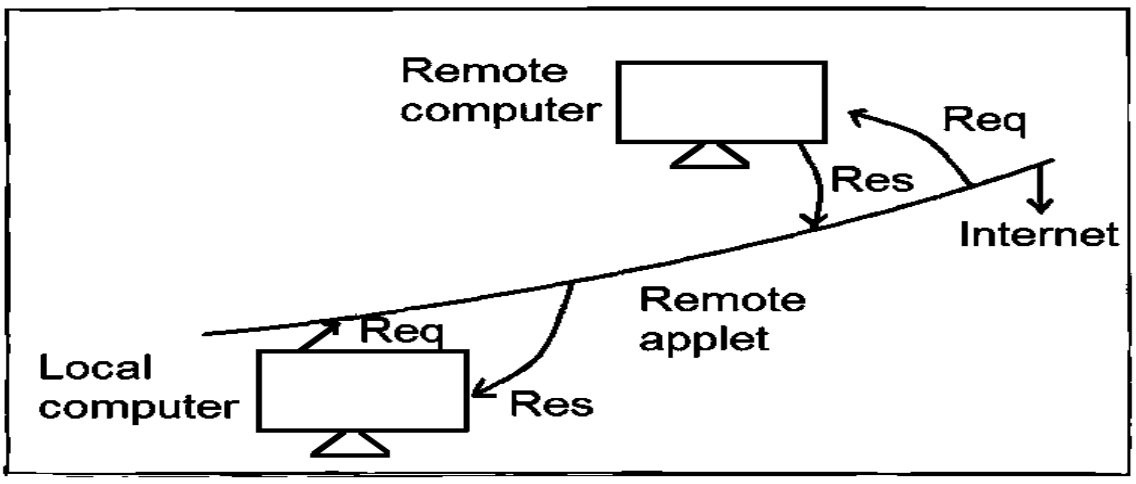
High Performance:
With the use of Just-In-Time compilers, Java enables high performance.

Distributed:
Java is designed for the distributed environment of the internet.

Dynamic:
Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry an extensive amount of run-time information that can be used to verify and resolve accesses to objects at run-time.

JAVA AND INTERNET:

Java is strongly associated with the Internet. Internet users can use Java to create applet programs and run them locally using a "Java-enabled browser" such as Hot Java. They can also use a Java-enabled browser to download an applet located on a computer anywhere in the Internet and run it on his local computer. In fact, Java applets have made the Internet a true extension of the storage system of the local computer.



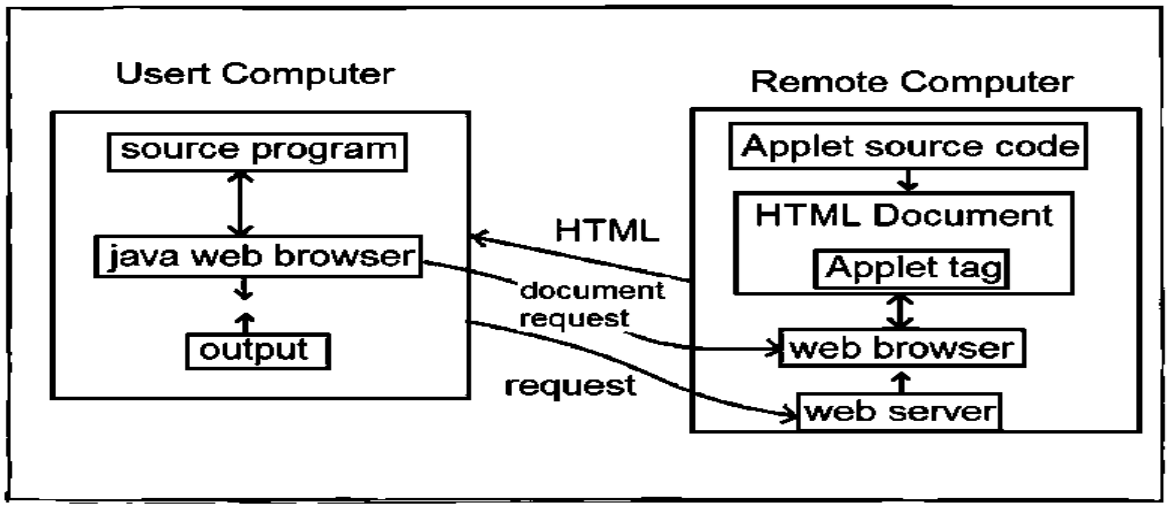
Java and Internet

JAVA AND WWW:

World Wide Web (WWW) is an open-ended information retrieval system designed to be used in the Internet's distributed environment. This system contains Web pages that provide both information and controls. Web system is open-ended and we can navigate to a new document in any direction. This is made possible with the help of a language called Hypertext Mark-up Language (HTML). Web pages contain HTML tags that enable us to find, retrieve, manipulate and display documents worldwide.

Java was meant to be used in distributed environments such as Internet. Since, both the Web and Java share the same philosophy, Java could be easily incorporated into the Web system. Before Java, the World Wide Web was limited to the display of still images and texts. However, the incorporation of Java into Web pages has made it capable of supporting animation, graphics, games, and a wide range of special effects.

1. Java communicates with a web page through a special tag called <applet>.
2. A Java user sends a request for an HTML document to the remote computer's net browser.
3. The web-browser is a program that accepts a request, processes the request and sends the required documents.
4. The HTML document is returned to that user browser.
5. The document contains the applet tag which identifies the applet. The corresponding applet is transferred to the user computer.
6. The Java enabled browser on the user's computer interprets the byte code and provides output.



Java and World Wide Web (www)

JAVA LANGUAGE INTRODUCTION:

Java programming language was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995 as core component of Sun Microsystems' Java platform (Java 1.0 [J2SE]).

The latest release of the Java Standard Edition is Java SE 8. With the advancement of Java and its widespread popularity, multiple configurations were built to suit various types of platforms. For example: J2EE for Enterprise Applications, J2ME for Mobile Applications.

The new J2 versions were renamed as Java SE, Java EE, and Java ME respectively. Java is guaranteed to be Write Once, Run Anywhere. gauge was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995 as core component of Sun Microsystems' Java platform (Java 1.0 [J2SE]).

Java is a high-level, general-purpose, object-oriented, and secure programming language developed by James Gosling at Sun Microsystems, Inc. in 1991. It is formally known as OAK. In 1995, Sun Microsystem changed the name to Java. In 2009, Sun Microsystem takeover by Oracle Corporation.

SIMPLE JAVA PROGRAM STRUCTURES:

Open Notepad and type in this program, maintaining the upper and lower case, because Java is a case-sensitive programming language.

```
class Krishna {
public static void main (String [] args) {
System.out.println("("Welcome to Basic concept of Java");
}
}
```

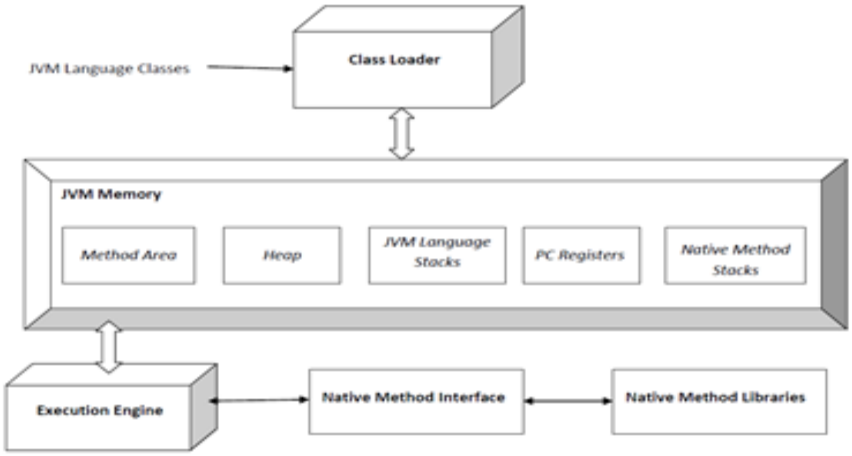
- After writing this code, save the program. When you save it, you need to save the program with only the class name like:
  Krishna.java //.java is extension of Java file

- After saving, compile and run the program, so you need to open a "cmd". Click the Window button and type "cmd" then hit Enter and open a "cmd". Then type the cmd command, for going to the location where your Java program is. For example, mine is at "desktop" so I need to type: cd desktop.

- Then show desktop, on your cmd, then type the following to compile:
  javac Krishna.java
  //javac for Java compile

- When your Java program isl compiled successfully, with no error, then you have an auto-created .class file.
- If the compile was successful, then to run type:
  Java Krishna
- The following is the output.

Java Virtual Machine Architecture:



Java virtual machine (JVM) is a virtual machine that enables a computer to run Java programs as well as programs written in other languages that are also compiled to Java bytecode. The JVM is detailed by a specification that formally describes what is required in a JVM implementation. Having a specification ensures interoperability of Java programs across different implementations so that program authors using the Java Development Kit (JDK) need not worry about idiosyncrasies of the underlying hardware platform.

1) Class Loader
The class loader is a subsystem used for loading class files. It performs three major functions viz. Loading, Linking, and Initialization.

2) Method Area
JVM Method Area stores class structures like metadata, the constant runtime pool, and the code for methods.

3) Heap
All the Objects, their related instance variables, and arrays are stored in the heap. This memory is common and shared across multiple threads.

4) JVM language Stacks
Java language Stacks store local variables, and its partial results. Each thread has its own JVM stack, created simultaneously as the thread is created. A new frame is created whenever a method is invoked, and it is deleted when method invocation process is complete.

5) PC Registers
PC register store the address of the Java virtual machine instruction which is currently executing. In Java, each thread has its separate PC register.

6) Native Method Stacks
Native method stacks hold the instruction of native code depends on the native library. It is written in another language instead of Java.

7) Execution Engine
It is a type of software used to test hardware, software, or complete systems. The test execution engine never carries any information about the tested product.

8) Native Method interface
The Native Method Interface is a programming framework. It allows Java code which is running in a JVM to call by libraries and native applications.

9) Native Method Libraries
Native Libraries is a collection of the Native Libraries (C, C++) which are needed by the Execution Engine.

Software Code Compilation & Execution process
In order to write and execute a software program, you need the following
1) Editor – To type your program into, a notepad could be used for this
2) Compiler – To convert your high language program into native machine code
3) Linker – To combine different program files reference in your main program together.
4) Loader – To load the files from your secondary storage device like Hard Disk, Flash Drive, CD into RAM for execution. The loading is automatically done when you execute your code.
5) Execution – Actual execution of the code which is handled by your OS & processor.

REFERENCE

1. World Wide Web design with HTML - C. Xavier TMH Publications, 2000. 2. The Complete Reference Java
2. https://www.tutorialspoint.com/java/java_overview.htm
3. https://en.m.wikipedia.org/wiki/Java_virtual_machine

PREPARED BY Dr.N.SHANMUGAVADIVU