***What is language?***

A way of communication is known as language.   
**Example:** Hindi, English etc.

***What is a Program?***

A set of instructions which is used to perform a specific task.

***What is a Programming Language?***

An artificial language used to write programs which can be translated into machine language and executed by computer with the help of some special software.

***What is a Platform?***

**Dictionary meaning**: A raised level surface on which things can stand.

**In programming:** Hardware or software on which a program can execute/run.

**Example:** c, c++, Java etc.

***Overview of Java***

***What is Java?***

Java is a high level programming language and also known as platform because of its JRE (java runtime environment).

***Brief History of Java.***

Java is one of the world’s most important and widely used computer languages, and it has held this distinction for many years. Unlike some other computer languages whose influence has weared with passage of time, while Java's has grown.

As of 2016, Java is one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers using and working on it.

Java language project initially started in June 1991 by James Gosling, Mike Sheridan, and Patrick Naughton. An oak tree stood outside Gosling’s office at that time and java named as oak initially. It later renamed as Green and was later renamed as java from java coffee.  
  
***Java released versions:***

JDK Alpha and Beta (1995)   
JDK 1.0 (23rd Jan, 1996)   
JDK 1.1 (19th Feb, 1997)  
J2SE 1.2 (8th Dec, 1998)   
J2SE 1.3 (8th May, 2000)  
J2SE 1.4 (6th Feb, 2002)  
J2SE 5.0 (30th Sep, 2004)  
Java SE 6 (11th Dec, 2006)  
Java SE 7 (28th July, 2011)  
Java SE 8 (18th March, 2014)

Java can be used to develop all kinds of software applications. So it is called as **Programming Suite**.  
As per the sun micro system standard the JAVA language is divided into three Editions.

1. **J2SE/JSE(java 2 standard edition)**
2. **J2EE/JEE(java 2 enterprise edition)**
3. **J2ME/JME(java 2 micro edition)**

**J2SE**

It is installable software as jdk software.

Latest version is 7.0 (Dolphin)

Java 6.0 is called as **Mustung** and Java 5.0 is called as **Tiger**.

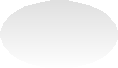
By using J2SE we can develop the standalone/Desktop applications/two tier applications.

**Example**: notepad, WordPad, paint, Eclipse IDE………..etc.

**Stand-alone applications:-**

The application which is specific to one computer and contains main() is called as **Standard/Stand-alone applications.**

1. Standalone applications are the java applications which don’t need the client server architecture.
2. The standalone applications applicable for the only one desktop. Hence it is called desktop applications or window based applications.
3. For the standalone applications doesn’t need internet connections.
4. It is a local application it doesn’t need any other external application support.
5. This type of the applications we can launch by using the command line or by using the executable jar.

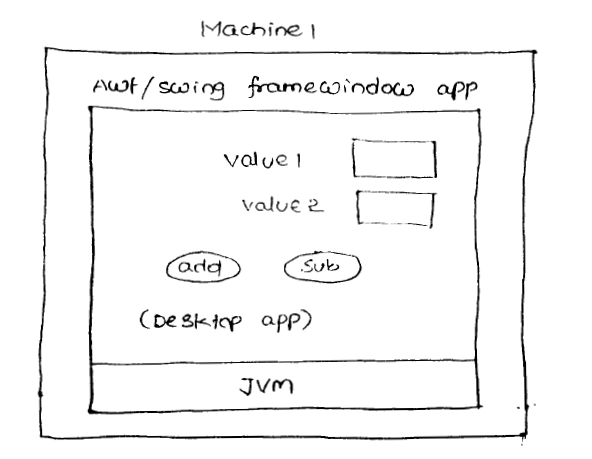


Client Database

**Desktop application:**

The stand-alone application that contains GUIness is called as **Desktop application.**

**Example:** Awt frame window application Swing frame window application.

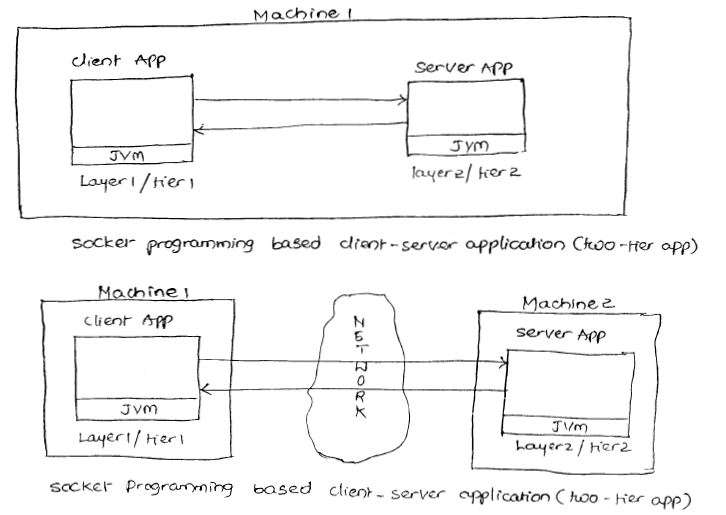
****

**Two-tier applications:**

The application that contains two layers communicating with each other is called as two-tier application.

A layer represents the logical partition in the application having logics.

The layers of two-tier application can be there in a single computer or can be there in two different computers.

To execute multiple java applications parallelly or simultaneously from a single computer, multiple JVM’s will be activated simultaneously or parallelly.

Applet is not desktop application. It is compiled Java class that can be sent over the network as webpage giving alternate to the HTML based form pages.

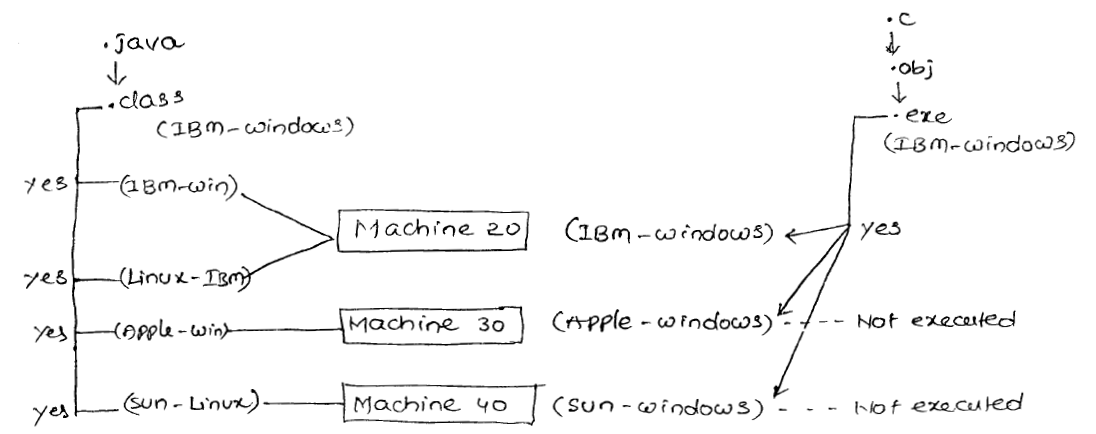
**JSE:**

Awt, swings, collection framework, JDBC, JNDI (Java naming and directory interface etc).

The alternate technologies for JSE module are VB, VB.Net, D2K. But these technologies based applications are platform dependent and architectural specific whereas the JSE module applications are architectural neutral and platform independent.

As of now three popular architectures are there to manufacture computers. Each architecture use methodology and plan for manufacturing computer. They are.

1. IBM Architecture.
2. Apple Architecture and
3. Sun Architecture.



The C, C++ generated .exe files not only contains operating system related instructions but they also contain instructions related to underlying computer architecture. Due to this we can say C, C++ languages are platform dependent and architecture specific whereas Java programming language is platform independent and architecture neutral because the Java compiler generated .class fie doesn’t contain underlying OS, computer architecture instructions.

**J2EE**

JEE means Java Enterprise Edition.

It is not installable software, it is given as a specification.

JEE specification contains rules and guidelines to develop web server and application server software like webogic, tomcat.

**Example**: Facebook, IRCTC, Flipkart etc.…

**Note:**Working with JEE is nothing but working with one or other web server or app server to develop the app.   
These web server, application server softwares are installable softwares.

For JEE module JSE module is base module.

Using JEE module the following applications can be developed

1. **Web Applications (web sites)**
2. **Distributed applications**
3. **Enterprise Applications**
4. **N- tier applications.**

The applications which can provide 24X7 access to their resources using internet environment are called as **Web Sites.**

The client-serverapplications which provide location transparency are called as **distributed applications**. If client application is able to recognize the server application location change dynamically are called as **location transparency.**

A web application/web site can be developed as distributed or non distributed application.

All credit card, debit card processing applications will be developed generally as **distributed applications.**

A large scale application that deal with complex and heavy weight business logic by having middleware service support is called as **Enterprise Applications.**

**Example:** Banking Application

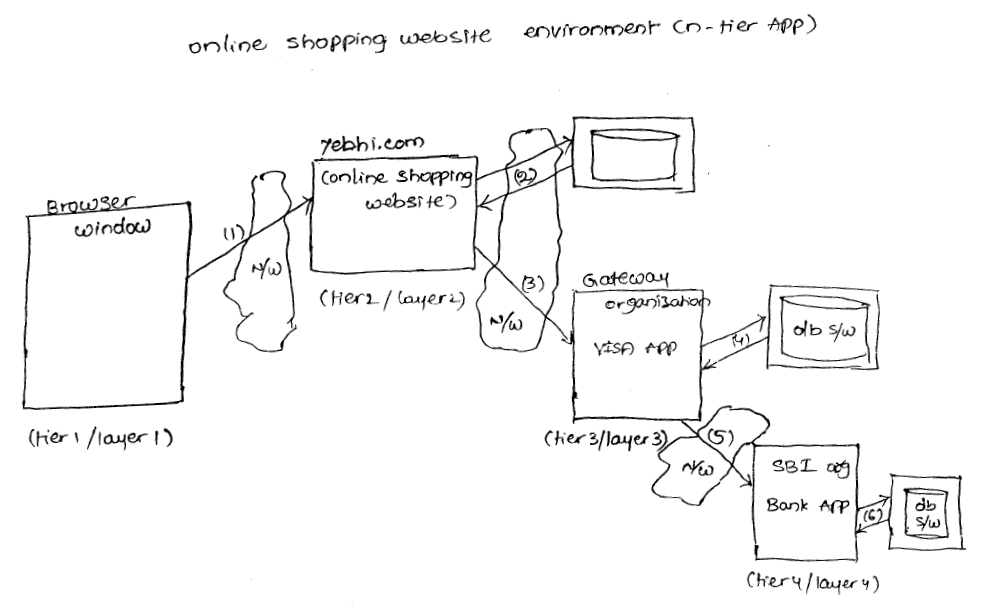
The additional services that are configurable on the applications to make applications running perfectly in all the situations are called as **Middleware services.**

**Example:** Security Service, Transaction management service, pooling service etc…

**Note:** Middleware services are not minimum logics of the application, they are additional and optional logics to develop.

The application that contains multiple layers/tiers (more than two) is called as n-tier applications.

**Example:** Online Shopping website environment.



Generally enterprise applications will be developed as n-tier applications.

JEE module concepts are servlets, JSP, EJB, JTA (Java Transaction API), JMS (Java Messaging Service), Java mail, JAAS (Java Authentication and Authorization Service), JMX (Java management Console), JCA (Java Connector API) and etc.

Struts, spring and Hibernate technologies are not part of sun micro systems JEE module. They are given by third party organizations as alternate and enhancement to JEE concepts.

While working with all JEE concepts we must take support of web server or application server software’s.

Majority projects of Java environment will be developed based on JEE module.

Every project belongs one domain. This JEE module is suitable for developing banking financial service, insurance, health care domain projects.

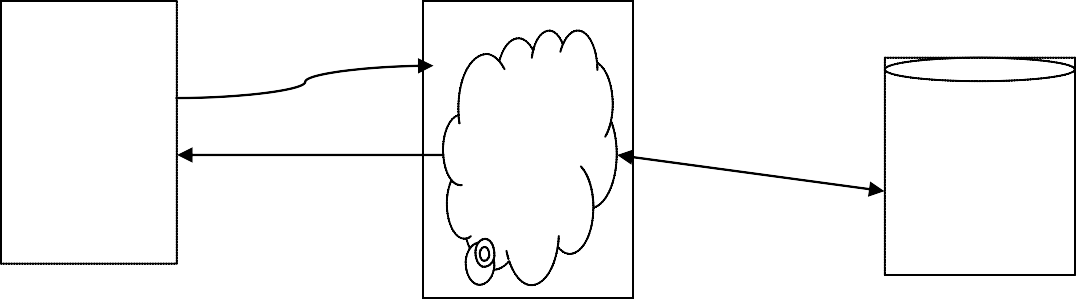
JSE module is suitable to develop gaming, automation and retail domain projects.

JME module is suitable for developing mobile gaming, telecommunications domain projects.

The latest version of JEE module is 6.

**Web-applications**:-

1. Web applications are the java applications which needs client and server concept.
2. Web applications must need the internet connections to access the application.
3. The application which is present in the internet is called the web application.
4. Web application can be launched by using HTTP driven. HTTP request sent to the Servlet present in the server side.



Client

server

database

Html

CssJsp

velocity

**request**

(java)(.net)(php)

**Hibernate**

**response**

**Jdbc**

Used to

store thedata.

**Client:**  
  
The Person who is sending the request is called client. All browsers comes under Clients.

**Example**: Google chrome, Mozilla Firefox, Internet Explorer etc.…

**Server:**  
  
In information technology, a server is a computer [program](http://searchsoftwarequality.techtarget.com/definition/program) that provides services to other computer programs (and their clients) in the same or other computers.

**Example**: Apache Tomcat, Oracle WebLogic, IBM WebSphere’s (WAS) etc.….

**Database:**  
  
Database is used to store the details like client details, application details, registration details……etc.

**Example**: Oracle, DB2, MySQL etc.….

**J2ME**

JME means Java mobile/micro edition.

For JME module JSE module is base module.

It is installable software.

Latest version is 2.X.

Given to develop mobile and micro applications in Java environment like sim cards, mobile games and etc.

Use WML (Wireless Markup Language), WAP in the application development.

These applications can be embedded with different types of devices( electronics, electrical, automobile and etc).

JSE, JEE module based applications cannot be attached with or embedded with different types of devices by bringing the code outside the computer. But this is possible with JME module.

The above process is called as embedded system programming and this is useful to add artificial intelligence to devices that means we can make the devices to think and take the decisions.

**Example:** A fully automated washing machine, robot.

To develop mobile related and artificial intelligence related applications in Java environment use JME module.

J2ME is used to develop mobile/micro based applications.

**Example**: WhatsApp, mynthra etc...

***API (Application Programming Interface)***

It is the base for the programmer to develop certain software technology based software applications.

Every software technology used built-in APIs. Using these APIs the programmers can develop user defined APIs and software applications.

In ‘C’ language, API is set of functions which come in the form of header files.

In ‘C++’ language API is set of functions, classes which come in the form of header files.

In ‘Java’ language API is set of classes, Interface, enums, annotations which come in the form of packages.

***Example***

1. *AWT API—Working* with java.awt, java.awt.event packages.

***Why java is used?***

The prime reason behind creation of Java was to bring portability and security feature into a computer language. Beside these two major features, there were many other features that played an important role in moulding out the final form of this outstanding language. Those features are given below

***Java features:***

1. **Simple, easy and familiar:**

Java is easy to learn and familiar because

a) The syntaxes of java language are very similar with C and C++ syntaxes

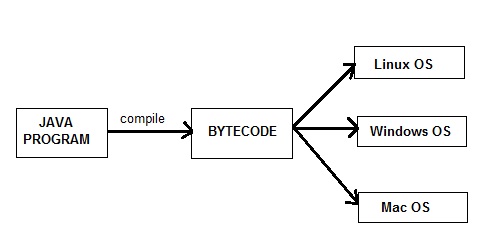
b) Eliminated the complex topics of C and C++ in Java, those are pointers and operator overloading.

Because of these factors Java is very easy to learn.

**2. Platform Independent:**

Unlike other programming languages such as C, C++ etc which are compiled into platform specific machines. Java is guaranteed to be write-once, run-anywhere language.

On compilation Java program is compiled into bytecode. This bytecode is platform independent and can be run on any machine, plus this bytecode format also provide security. Any machine with Java Runtime Environment can run Java Programs.



3. **Object-Oriented:**

In java everything is Object which has some data and behaviour. Java can be easily extended as it is based on Object Model.

**4. Robust**:

Any technology if it is good at two main areas it is said to be ROBUST

1. **Exception Handling**
2. **Memory Allocation**

JAVA is Robust because

JAVA is having very good predefined Exception Handling mechanism whenever we are getting exception we are having meaning full information.

JAVA is having very good memory management system that is Dynamic Memory (at runtime the memory is allocated) Allocation which allocates and deallocates memory for objects at runtime.

JVM – It will take care of allocating the memory

Garbage Collector – It will take care of de-allocating the memory

**5. Secure:**

When it comes to security, Java is always the first choice. With java secure features it enable us to develop virus free, temper free system. Java program always runs in Java runtime environment with almost null interaction with system OS, hence it is more secure.

**6. Distributed:**

Java provides the network facility. I.e. programs can be access remotely from any machine on the network rather than writing program on the local machine. HTTP and FTP protocols are developed in java.

**7. Portable:**

Means able to be easily carried or moved. Write once, run anywhere (WORA) feature makes it portable.

**8. Architecture-Neutral:**

Java code is translated into byte code after compilation which is independent of  any computer architecture, it needs only JVM (Java Virtual Machine) to execute.

**9. High performance:**

JVM can execute byte codes (highly optimized) very fast with the help of  Just in time (JIT) compilation technique.

**10. Multithreading:**

Java provides multitasking facility with the help of lightweight processes called threads.

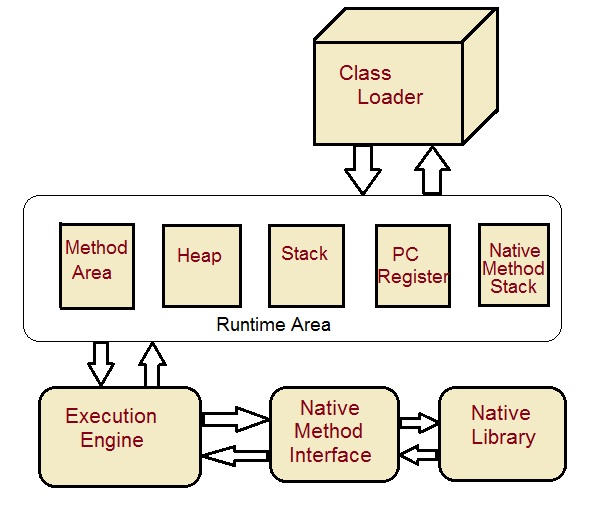
**11. Dynamic:**

Java have the capability of linking dynamic new classes, methods and objects.

***What is JVM?***

Java virtual Machine (JVM) is a virtual Machine that provides runtime environment to execute java byte code. The JVM doesn't understand Java typo, that's why you compile your \*.java files to obtain \*.class files that contain the bytecodes understandable by the JVM.

JVM control execution of every Java program. It enables features such as automated exception handling, Garbage-collected heap.



***JVM details:***

**1. Class loader subsystem:**

It is a part of JVM that care of finding and loading of class files.

**2.  Class/method area:**

It is a part of JVM that contains the information of types/classes loaded by class loader. It also contain the static variable, method body etc.

**3.  Heap**:

It is a part of JVM that contains object. When a new object is created, memory is allocated to the object from the heap and object is no longer referenced memory is reclaimed by garbage collector.

**4. Java Stack:**

It is a part of JVM that contains local variable, operands and frames. To perform an operation, Byte code instructions takes operands from the stack, operate and then return the result in to the java stack.

**5. Program Counter:**

For each thread JVM instance provide a separate program counter (PC) or pc register which contains the address of currently executed instruction.

**6. Native method stack:**

As java program can call native methods (A method written in other language like c).  Native method stack contains these native methods.

**7. Execution Engine:**

It is a part JVM that uses Virtual processor (for execution), interpreter (for reading instructions) and JIT (Just in time) compiler (for performance improvement) to execute the instructions.

***How JVM is created(Why JVM is virtual):***

When JRE installed on your machine, you got all required code to create JVM. JVM is created when you run a java program, e.g. If you create a java program named FirstJavaProgram.java. To compile use – java FirstJavaProgram.java and to execute use – java FirstJavaProgram. When you run second command – java FirstJavaProgram, JVM is created. That’s why it is virtual.

**Lifetime of JVM:**    
  
When an application starts, a runtime instance is created. When application ends, runtime environment destroyed. If n no. of applications starts on one machine then n no. of runtime instances are created and every application run on its own JVM instance.

**Main task of JVM:**

1. Search and locate the required files.
2. Convert byte code into executable code.
3. Allocate the memory into ram
4. Execute the code.
5. Delete the executable code.

***Difference between JVM, JRE and JDK***

**JVM (Java Virtual Machine):**

JVM is a virtual machine or a program that provides run-time environment in which java byte code can be executed. JVMs are available for many hardware and software platforms. The use of the same byte code for all JVMs on all platforms make java platform independent.

***JRE (Java Runtime Environment):***

JVM + java runtime libraries + java package classes (e.g. util, Langetc.). JRE provides class libraries and other supporting files with JVM. It not provide any development tool like compiler, debugger etc.



***JDK (Java Development Kit):***

 JRE+ development tool (compiler, debugger etc.). JDK contains tools to develop the application and JRE to execute the application.



***JAVA ENVIROMENT SETUP***

***Install the software and set the path:-***

Download the software from internet based on your operating system. The software is different from 32-bit operating and 64-bit operating system.

***To download the software open the following web site.***

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

**For 32-bit operating system please click on**

Windows x86 :- 32- bit operating system

**For 64-bit operating system please click on**

Windows x64 :- 64-bit operating system

After installing the software the java folder is available in the fallowing location

LocalDiskc:--->programFiles->java-->jdk (java development kit), jre (java run time environment)

To check whether the java is installed in your system or not goto the command prompt. To open the command prompt

In the command prompt type: - javac (Click on Enter)

‘javac’ is not recognized is an internal or external command, operable program or batch file.

Whenever we are getting above information at that moment the java is installed but the java is not working properly.

Whenever we are typing javac command on the command prompt

1. Operating system will pick up javac command search it in the internal operating system calls. The javac not available in the internal command list.
2. Then operating system goes to environmental variables and check is there any path is sets or not. Up to now we are not setting any path. So operating system don’t know anything about javac command Because of this reason we are getting error message.

Hence we have to set environmental variables. The main aim of the setting environmental variable is to make available the following commands javac,java,javap (softwares) to the operating system.

**To set the environmental variable:**-

MyComputer (right clickon that)---->properties----->Advanced--->Environment Variables---->

User variables > click on new button and enter below values

Variable name : Path

Variable value :C:\Program Files (x86)\Java\jdk1.7.0\_79\bin;.;

Click on Ok -> Ok ->

Now the java is working good in your system. open the command prompt to check once C:>javac--> now list of commands will be displayed

If List of commands displayed on Console, that means java installed successfully. (Happy Coding..!!)