JAVA

Q. what is JAVA?

JAVA is pure object oriented language

Q. what is object oriented?

Object oriented is a concept of programming and which based on four pillars and seven features.

There are two types of object oriented languages

1) Semi object oriented: Semi object oriented means those language can use OOP concept but there is no compulsion to use object oriented every time called as semi object oriented.

Example of semi object oriented language is: CPP

2) Pure Object oriented: those languages must be use OOP concept compulsory called as pure object oriented language means in java we cannot run program without class so java is pure object oriented.

Q. what is diff between CPP and JAVA?

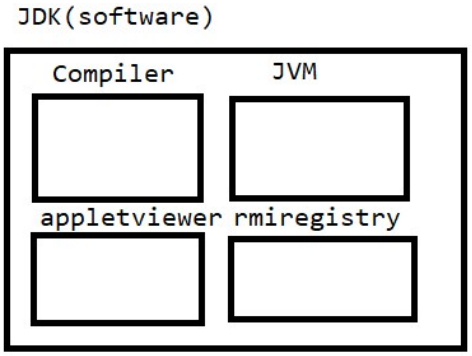
CPP	**JAVA**
1.CPP is semi object oriented	JAVA is Pure object oriented
2. CPP use operator overloading	JAVA not use operator overloading
3.CPP us delete operator	JAVA not use delete operator
4 CPP use Destructor	JAVA not use Destructor use garbage collection
5.CPP use pure virtual function for achieve abstraction	JAVA provide separate abstract keyword for abstraction
6. CPP use Template as generics data	JAVA use generics and it is denoted by < >
7.CPP use Pointer externally	JAVA not use pointer externally
8. CPP is platform dependent	JAVA is platform independent
9.CPP contain three access specifier private ,public and protected	JAVA use four access specifier private public protected and default.

If we want to work with JAVA we have to know how to create program in java and execute it.

Steps to work with program in JAVA

1) Installed JDK: JDK stands for Java Development kit and it is Software or application which contains some supporting application which is responsible to create enviourment for develop and execute JAVA Application.

IT contain some software like as Compiler, JVM etc



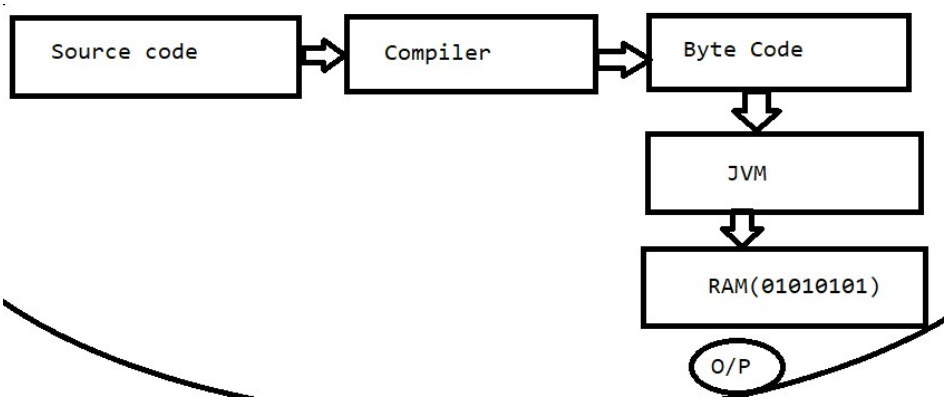
if we think about above diagram we demonstrate we have JDK and JDK contain some supporting software's.

Q. what is Compiler?

Compiler is application or software which is used for convert your source code to Byte code in java shown in following diagram.

Q. what byte code?

Byte code is intermediate code which easily converts in machine code with the help of JVM or byte code is machine understandable code means byte code read by JVM and load in Memory and generate machine from that and generate output to end user.



Q. What is JVM?

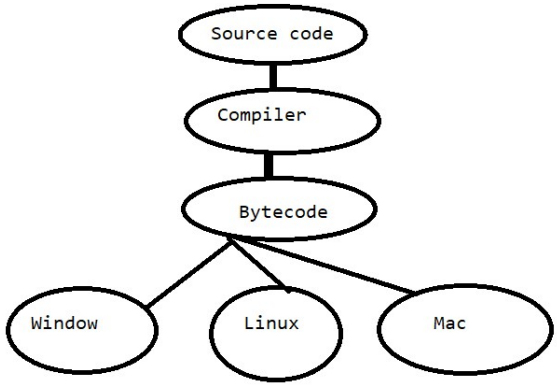
JVM stands for java virtual machine and it is application which easily read byte code and generates output.

Q. why use byte code?

Byte code is platform independent code of java means byte code can execute on any operating system so JAVA is platform independent language.

Example: suppose consider if we compile your code on window operation then we can run this code on Linux without taking support any third party software called as platform independent.

Following Diagram shows Platform Independency



In above diagram we demonstrate byte code and execute on any operation system so byte code is platform independent code so java is also platform independent code.

2) Create Sample program: if we want to create any program in java we need to write minimum single class so if we want to work with java we have some following standard and generalize syntax.

Generalize syntax is

```
access specifier  class classname{
        public static void main(String x[])
        { write here logics
        }
}
```

Example:

```
public class FirstDec
  {
        public static void main(String x[])
        { System.out.println("good morning");
        }
  }
```

Code Description

public class FirstDec: Here public is access specifier

Q. what is access specifier?

access specifier are some keywords which is used for apply restrictions on class and its member.

There are four types of access specifier?

private : private access specifier means we can cannot access class member outside of class.

public: public is access specifier which is used for access class member outside of class or outside of package.

protected: protected class member is only accessible in child class

default : default class member can accessible outside of class but within same package.

class is keyword for class declaration

FirstDec is classname and user can give any name to his class.

public static void main(String x[]): it is main function of java same like as main function in C or CPP. Interview questions on main method we will discuss later

i) why main method is static ii) can we overload main method iii) can we pass different parameter in main method other than string etc

System.out.println("good morning"): it is output statement of java same like as printf in c language

Internal Meaning of System.out.println()

System is class out is static reference of PrintStream class System and PrintStream maintain HAS-A relationship between them and println() is overloaded method for display output on output screen.

Note: we will discuss this meaning in inheritance chapter of JAVA

if we want to create Java program we have number of text editors as well as tool

notepad, wordpad, sublime text , visual code studio,eclipse,netbeans, interllJ etc
but now this our first lecture of JAVA so we will use notepad

3) Save Program: if we want to save java program then save in bin folder where JDK installed and give classname and filename same with .java extension.
Example: our file name should be FirstDec.java

4) Compile Program:

Q. what is compilation?

Compilation is process where we convert your source code to byte code in java.

Q. How to compile java program when we create program using notepad?

Steps: start menu --- search --- command prompt --- go where java file save -- type command javac filename.java.

C:\Users\Admin>cd C:\Program Files\Java\jdk1.8.0_291\bin
C:\Program Files\Java\jdk1.8.0_291\bin>javac FirstDec.java

if we think about above code when we open command prompt then it is by default on c:\\user\\Admin but our java program in bin folder so when we want to change path using command prompt we have command CD(Change Directory).
javac – stands for java compiler javac filename.java means we provide file to java compiler for compilation purpose means after this command java compiler can generate byte code and it is stored in .class file means after program compilation java compiler create new file automatically with extension of .class file and in this file contain your byte code.

As per our example we have two files after java program compiled

i) FirstDec.java – source code

ii)FirstDec. Class after compilation it contain byte code.

5) Run Program: when we want to run java program we have command java filename

Here java work as JVM

Java FirstDec

Example: Now we want to create program to calculate addition of two values

```
public class AddApp
{   public static void main(String x[])
    {    int a,b,c;
                a=100;
        b=200;
         c=a+b;
   System.out.printf("Addition is %d\n",c);
    }
}
```

If we think about above code we calculate addition of two numbers but we calculate fix number addition and we want to accept input from keyboard and then calculate its addition

If we want to accept input from keyboard we have two ways
1) Use Command Line Arguments
2) Use Scanner class.

Command line Arguments

Q. what is command line arguments?

Command line argument is infinite string array present in main function called as command line arguments.

```
public class AddApp
{
    public static void main(String x[])
    {    int a,b,c;
        a=100;
         b=200;
          c=a+b;
   System.out.printf("Addition is %d\n",c);
    }
}
```

Command Line Arguments

Means using command line argument we can accept n number of input or infinite input from keyboard but the first input of command line argument is at position of zero because it is array and array start from 0^{th} index up to n-1.

Example

```
public class AddApp
{
   public static void main(String x[])
   {   int a,b,c;
              a=x[0];
       b=x[1];
       c=a+b;
  System.out.printf("Addition is %d\n",c);
   }
}
```

if we think about above code we get compile time error in computable types.
because we have statement int a=x[0] and int b=x[1] here x[0] and x[1] is type of string but we want to accept input of type integer and string cannot store in integer directly so compiler will generate error to us.
so if we want to avoid this problem or solve this error we need to convert string value to integer value.

How to convert String value to integer value

if we want to perform conversion between string value to integer value we have following statement

int variable =Integer.parseInt(String) : convert string to integer

in this statement Integer is class and parseInt() is static function of Integer class which is used for return string value to integer value.

float variable=Float.parseFloat(String): convert string to float

Float is class and parseFloat() is a static function which is used for convert string value to integer value.

long variable=Long.parseLong(String): convert string to long

```
public class AddApp
{
   public static void main(String x[])
   {    int a,b,c;
           a=Integer.parseInt(x[0]);
            b=Integer.parseInt(x[1]);
       c=a+b;
  System.out.printf("Addition is %d\n",c);
   }
}
```

Example: WAP to input number and reverse it using command line argument

```
public class ReverseNum
{
  public static void main(String x[])
  {   int r=0;
     int no=Integer.parseInt(x[0]);
     while(no!=0)
     {  int rem = no % 10;
        no  = no /10;
        r=r*10+rem;
     }
 System.out.println("Reverse is  "+r);
  }
}
```

If we think about above code we accept input from keyboard but command line argument can accept input from keyboard on single line.

if we want to accept input on new after program running statement for that we have Scanner class.

Steps to work with Scanner class

1) add java.util package in application

Q. what is package?

package is a collection of classes and interfaces it is like as header file in c language.

if we want to add package in program we have import keyword

import packagename.* : here if we use * means we import all member from package in application.

or

import packagename.membername: if we use this syntax then we can import specific member from package in application.

Example: import java.util.*;

or

import java.util.Scanner;

2) Create object of Scanner class

if we want to create object of Scanner class we have following syntax

Scanner ref = new Scanner(System.in);

Example: Scanner xyz = new Scanner(System.in);

3) Use its method to work with Scanner

if we want to work with Scanner we have following methods

int nextInt(): this method is used for accept input of type integer

float nextFloat(): this method is used for accept input of type float

double nextDouble(): this method is used for accept input of type double

long nextLong(): this method is used for accept input of type long

short nextShort(): this method is used for accept input of type short

String nextLine(): this method is used for accept input of type string etc

Example:

```
import java.util.*; //step1
public class AddUsingScannerApp
{  public static void main(String x[])
  {Scanner xyz  = new Scanner(System.in);//step2
      int a,b,c;
          System.out.println("Enter two values");
           a=xyz.nextInt();
    b=xyz.nextInt();
    c=a+b;
 System.out.printf ("addition is %d\n",c);
  }
```

```
}
```

Example: WAP to input name id and salary of employee from keyboard and display it.

```
import java.util.*;
public class EmployeeApp
{
  public static void main(String x[])
  {
    Scanner xyz  = new Scanner(System.in);
     int id,sal;
      String name;
   System.out.println("Enter name of employee");
   name=xyz.nextLine();
   System.out.println("Enter id of employee");
   id=xyz.nextInt();
   sal=xyz.nextInt();
   System.out.printf("Name is %s\n",name);
   System.out.printf("Id is %d\n",id);
   System.out.printf("Salary is %d\n",sal);
  }
}
```

Arrays in JAVA

Q. array is a collection of similar data type.

How to use array in java?

If we want to use array in java we have following syntax

1) Declare array: when we declare array then it is by default null means in java array variable work as reference or pointer.

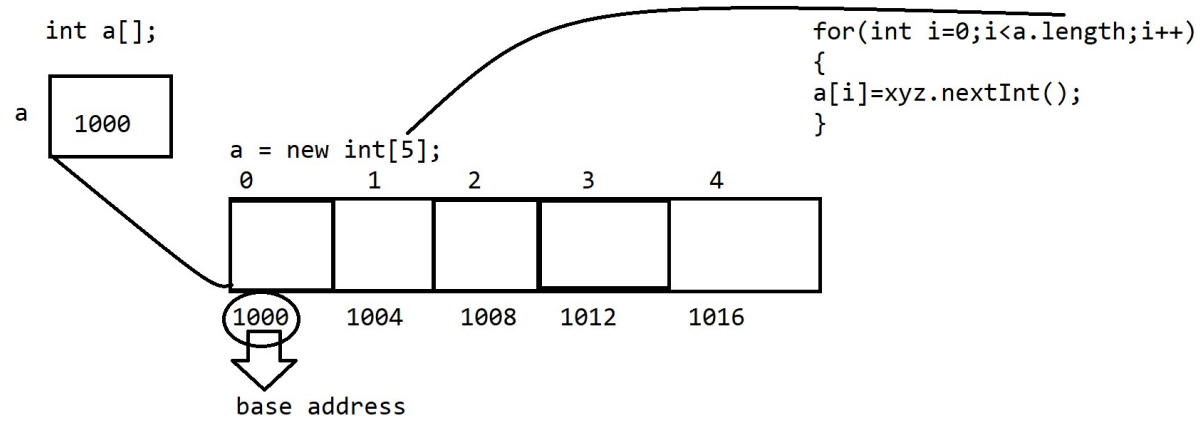
Syntax: datatype variablename[]; //this is reference variable and contain default value as null

e.g int a[]; //here a work as reference.

2) Allocate memory of array: if we want to allocate memory of array we have to use new keyword means using a new keyword we can allocate memory of array.

variablename = new datatype[size];

e.g a = new int[5]; here we create new 5 block by using new keyword and store its base address in reference variable a shown in following diagram.



Example: WAP to create array of size 5 and store values in it and display it.

```
import java.util.*; //step1
public class ArrayApp
{
  public static void main(String x[])
  {  Scanner xyz  = new Scanner(System.in);
    int a[]; //declaration
     a = new int[5];
    System.out.println("Enter values in array");
          for(int i=0; i<a.length; i++)
    { a[i]=xyz.nextInt();
    }
          System.out.println("Display array values");
                for(int i=0; i<a.length;i++)
                { System.out.printf("a[%d] ---->%d\n",i,a[i]);
                }
  }
}
```

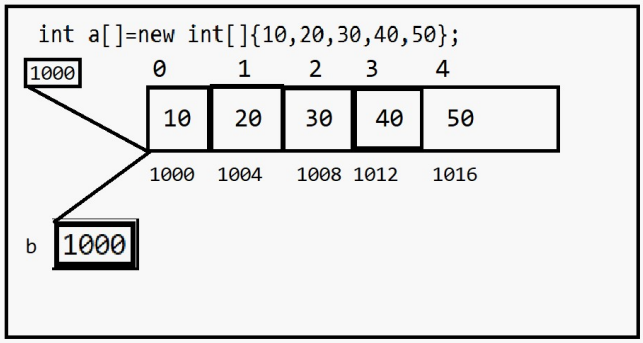
Example: WAP to input five values in array and find maximum value from Array.

```
import java.util.*; //step1
public class ArrayApp
{ public static void main(String x[])
  {  Scanner xyz  = new Scanner(System.in);
    int a[]; //declaration
     a = new int[5];
    System.out.println("Enter values in array");
          for(int i=0; i<a.length; i++)
```

```
    { a[i]=xyz.nextInt();
    }
          int max=a[0];
         System.out.println("Display array values");
              for(int i=0; i<a.length;i++)
              { System.out.printf("a[%d] ---->%d\n",i,a[i]);
                    if(a[i]>max)
                    { max=a[i];
                    }
              }
        System.out.printf("Maximum value is %d\n",max);
  }
}
```

Q . what will be output of given code?

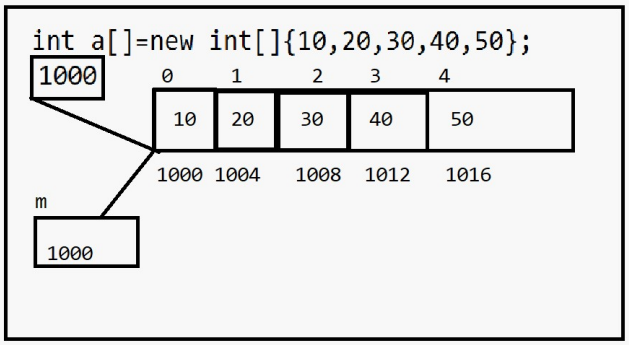
```
import java.util.*; //step1
public class ArrayApp
{
   public static void main(String x[])
   {
     int a[]=new int[]{10,20,30,40,50};
     int b[];
     b=a;
      b[1]=200; 1000+1*4 = 1004
    System.out.println("Display array values");
       for(int i=0; i<a.length;i++)
       { System.out.println(a[i]);
       }
   }
}
```



if we think about above code we have int a[]=new int[]{10,20,30,40,50} here we create array of size 5 internally and we store base address of array i.e 1000 in reference variable a and we have one more statement int b[] means we declare reference of integer type of array and we have again one more statement b=a; here we store base address array a i.e 1000 in reference variable b means reference variable b and reference variable a points to same memory and again we have statement b[1]=200 means we store 200 on address location 1004 or a[1] reference variable a and reference variable b points to same array so 200 get override on a[1] so when we print array then we get output 10 200 30 40 50

Q. what will be output of given code?

```
public class ArrayApp
{
   public static void main(String x[])
   {
     int a[]=new int[]{10,20,30,40,50};
      show(a,0);
     System.out.println("Display array values");
         for(int i=0; i<a.length;i++)
         { System.out.println(a[i]);
         }
   }
   public static void show(int m[],int count)
   {
        if(count!=m.length)
        { m[count]=m[count]+10;
        show(m,++count);
        }
   }
}
```



Output: 20 30 40 50 60

if we think about above code we have two array int a[]=new int[]{10,20,30,40,50} here we have array with 5 values and we store its base address i.e 1000 in reference variable a and we have one more function name as show(a,0) here we call show() function and pass base address of array to variable m means m is array which contain base address of a means m contain 1000 so m points to variable a so we have statement if(count!=m.length) so here initially count is 0 so the condition 0!=5 is true so we have m[count]=m[count]+10 means m[0] =m[0]+10 means a[0]=a[0]+10 means m[0]=10+10 so this value 10+10 is 20 and it is stored on address location 0^{th} or 1000 address where a and m points so previous value of 1000 location get override and 20 value store on 1000 location means a[0]=20 again we have statement show(m,++count) here we perform recursion again count is 1 and so on means we increase ever value by 10 and store in array so we get final output 20 30 40 50 60

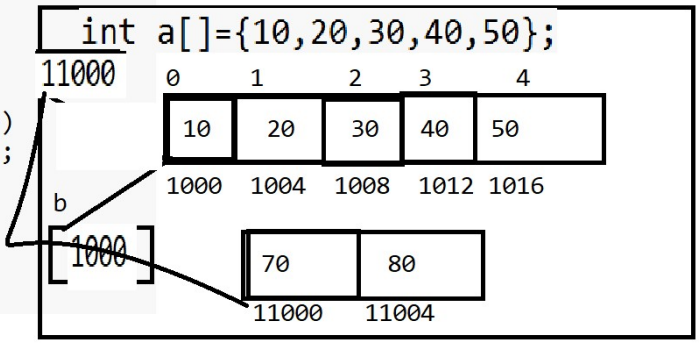
Q. what will be output of given code?

```
import java.util.*; //step1
public class ArrayApp
{
  public static void main(String x[])
  {
    int a[]=new int[]{10,20,30,40,50};
     show(a,0);
          System.out.println("Display array values");
//10 1 30 40 50
                  for(int i=0; i<a.length;i++)
                  { System.out.println(a[i]>>1);  //5  0  10  15  20  25
                  }
  }
  public static int[] show(int m[],int count)
  {  m[1]=4>>2;
     return m;//return base address of array
  }
```

}

```
import java.util.*; //step1
public class ArrayApp
{
   public static void main(String x[])
   {
     int a[]={10,20,30,40,50};
      int b[]=a;
       a=new int[]{70,80};
     for(int i=0; i<a.length;i++)
      { System.out.println(b[i]);
      }
   }

}
```



Ways to declare array in java

Syntax: datatype variablename [];

Or

datatype [] variablename;

Two dimensional arrays in java

It is used for create matrix

How to declare two dimensional arrays in java

Syntax: datatype variablename[][]

or

datatype [][]variablename;

or

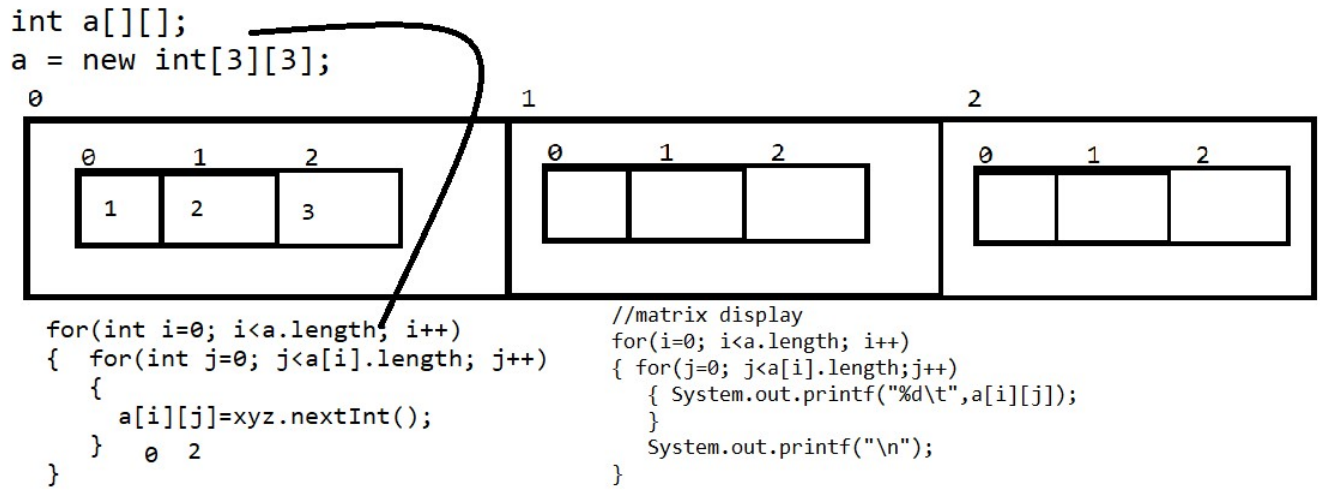
datatype []variablename[];

```
int a[][],b,c[];//here a is two d array b is normal variable and
                //c is single dimension array

int [][] a,b,c,d;//here all variables are two d array

int[] a[], b,c[][]; //here a is two dimensional array
                    //b is single dimension array
                    and c is three dimensional array.
```

Example: **WAP to create 3 x 3 matrixes and display it.**



Example

```
import java.util.*;
public class MatrixApp
{ public static void main(String x[])
  {  Scanner xyz  = new Scanner(System.in);
     int a[][]=new int[3][3];
     System.out.println("Enter values in matrix");
     for(int i=0; i<a.length;i++)
     {
                    for(int j=0; j<a[i].length; j++)
       {
          a[i][j]=xyz.nextInt();
       }
     }
    System.out.println("Display matrix");
    for(int i=0; i<a.length;i++)
     {
                    for(int j=0; j<a[i].length; j++)
       {
          System.out.printf("%d\t",a[i][j]);
       }
                  System.out.printf("\n");
     }
  }
}
```

Q. what will be output of given code?

```
import java.util.*;
public class MatrixApp
{ public static void main(String x[])
  {
    int a[][] = new int[][]{
                                                    {1,2,3},
                                                    {4,5,6},
                                                    {7,8,9}
                                                };
    int b[]=a[1];
           for(int i=0; i<b.length;i++)
     { b[i]+=10;
             }
  System.out.println("Display array");
          for(int i=0; i<a.length; i++)
          {
                  for(int j=0; j<a[i].length;j++)
                  { System.out.printf("%d\t",a[i][j]);
                  }
                  System.out.printf("\n");
          }
  }
}
```

Jagged Array in JAVA

Jagged Array is used for create matrix every row having a different column list called as Jagged Array.

1 2 3
4 5 6 7
8 9

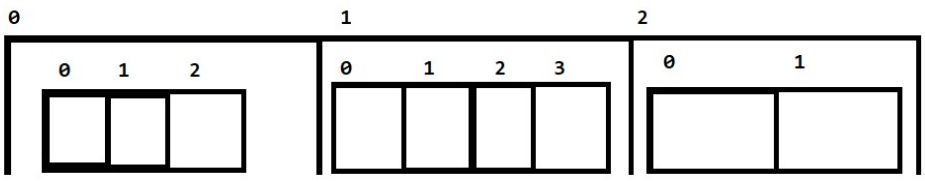
Syntax:

data type variablename[][]=new datatype[rowsize][];
variablename=new datatype[rowindex][colsize];

suppose consider we want to create matrix of 3 x 3 like as

```
1   2   3
4   5   6   7
8   9
```

```
int a[][]=new int[3][];
a[0] = new int[3];
a[1] = new int[4];
a[2] = new int[2];
```




```
for(int i=0; i<a.length; i++)
{  for(int j=0; j<a[i].length;j++) a[1].length
   { a[i][j]=xyz.nextInt();
   }
}
```

```
import java.util.*;
public class JaggedArray
{
  public static void main(String x[])
  {
    Scanner xyz  = new Scanner(System.in);
    int a[][]=new int[3][];
    a[0]=new int[3];
    a[1] = new int[4];
    a[2] = new int[2];
    System.out.println("Enter values in matrix");
                for(int i=0; i<a.length;i++)
                {
                    for(int j=0; j<a[i].length;j++)
                            {
                             a[i][j]=xyz.nextInt();
                            }
    }
                System.out.println("Display matrix");
                for(int i=0; i<a.length;i++)
                {
                    for(int j=0; j<a[i].length;j++)
                            {
                             System.out.printf("%d\t",a[i][j]);
                            }
                   System.out.printf("\n");
    }
  }
}
```

Q. why java is platform independent language?

Because of byte code means when we compile java program then java compiler generate byte code and byte code is platform independent code so java is platform independent language.

Q. Why java pure object oriented?

Because java program cannot create without inheritance as well as without class and both feature belong from object oriented concept so we can say java is pure object oriented language.

Q. Why Java is not pure object oriented language?

Because java support to primitive type of data as well as java class not use by default encapsulation so we can say java is not pure object oriented.

Q. what is name of java compiler?

JIT is name of java compiler and JIT stands for Just in Time.

Q. what is JVM and why use it?

JVM stands for Java virtual machine and it is application present under JDK and JRE and which is used for load class file means load byte code generated by java compiler and perform different operation on it means verify byte code, perform garbage collection,object memory allocation ,thread management ,memory management etc these are the roles of JVM means we can say JVM is responsible for manage run time environment required for java program or application.

Q. is JVM platform dependent or independent?

JVM is platform dependent

Q. Why JVM is platform dependent?

JVM is platform dependent because the major goal of JVM is to generate machine code from byte code and every operating system has different machine code generation algorithm so Java design JVM separate for every operating system and common compiler for all operating system means when java want to give setup for windows then compiler is same but JVM different , when java want to give setup for Linux then compiler is same but JVM is different , when java want to give set up for mac then Compiler is same but JVM is different so we can JVM design separate for every operating system of JVM is platform dependent

Q. can we overload main method in java?

Yes we can overload the main method in java but JVM can identify only one method for program execution in which string [] argument is present.

```
public class Test
{ public static void main(int x[])
  {
  }
  public static void main(String x[])
  {
  }
}
```

JVM run program by using this method.

Q. what will be output of this program?

```
public class ABC
{
   public static void main(int x[])
   {
     System.out.println("I am main method");
   }
}
```

if we think about the above code program compile successfully but not run because in main method we pass int x[] so compiler think main may be overloaded by user but JVM run program using main method in which contain string [] parameter.

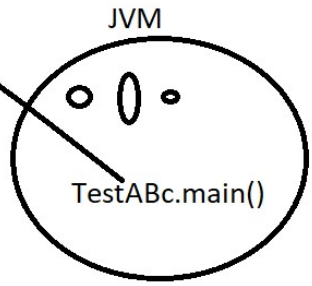
Q. why main method of java is static?

Because main method of java call by JVM internally and JVM need to call main method without creating object of and if we have static member then we can call it without object means we can call it using class name so java main method is static

Following diagram describe how to call main method by JVM

```
public class TestABC
{
    public static void main(String x[])
    {

    }
}
```



Q. Why JVM need to call main method without object?

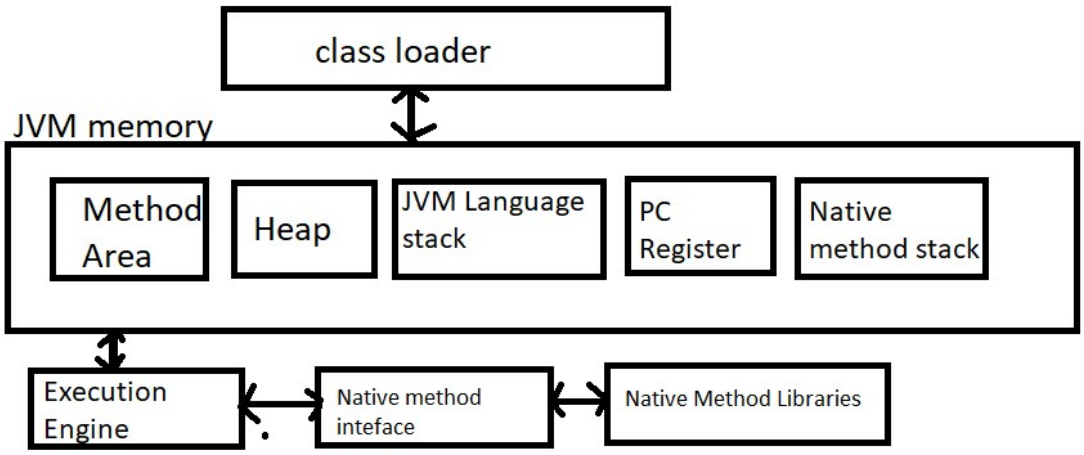
Because main method present in class and every class contain one constructor called as implicit constructor and if JVM create object of class in which main method present then there is possibility constructor of class call before main method means when we have parameterized constructor in class and if JVM create object of class in which main method present then JVM need to pass value to constructor at the time of object creation but JVM cannot pass value to use constructor so avoiding this problem java want to call main method by JVM without object so main method is static.

Q. Explain JVM Architecture?

JVM is responsible for manage the overall activity of Java program execution

Means when we compile java program then java compiler create .class file and store byte code in it and load .class file by JVM in memory and perform different operation on it means

Allocate memory for class and its member, allocate memory for objects, destroy objects from memory as well as manage n number of things for java program shown in following diagram or show by JVM architecture.



If we think about above diagram then we have first component in JVM architecture is class loader.

What is purpose of class loader?

The major goal of class loader is read .class file generated by java compiler and perform three different operations on it i.e. loading, linking and initialization of .class file**.**

Q. what is loading?

when compiler generate .class file then JVM class loader reads .class file and generate corresponding binary data from .class file and save it in the method area and JVM store the following information in method area

I) Fully Qualifier name of the loaded class and its immediate parent class

ii) Store information about interface, class or enum

iii) Store information related with modifier, variables and method information.

Example: suppose consider we have class name as

```
class ABC
{ private int a;
  public void show()
   {
   }
}
```

when we compile this program then compiler generate file name as ABC.class and when we run this program then JVM class loader read file name as ABC.class and store class name , private,public int a and void show() its information in method area means it load all class details in method area.

Q. what is linking?

Linking is process to verify and prepare byte code as well as binary code which loaded by class loader. means in short we can in the linking processes check .class file is created properly or not means in linking we have BytecodeVerifier which is used for verify byte code generated property or not if byte code not generated properly it will throw error at run time java.lang.VerifyError

as well as linking section JVM allocates memory for class static variable and initialization the default in it means JVM allocate memory static variable before creating and give some default value to them means if static variable is integer then 0 default if static variable is string then give null value etc.

Q. what is initialization?

after verify byte code and allocate memory to static variable after that JVM try to initialize code for execution purpose from top to bottom.

There are three types of initialize in JVM

1) Bootstrap class loader 2) extension class loader 3) System class loader.

1) Bootstrap class loader: bootstrap class loader is used for load the core API classes present in JAVA_HOME/lib folder.

2) Extension class loader : it is used for load classes present under JAVA_HOME/jre/lib/ext folder

3)System class loader or Application class loader : it is child of extension class loader it is responsible load all classes from classpath.

JVM Memory?

JVM memory is distribution of memory management in ram means when your program running in memory then JVM memory decides which content program store in which section of memory.
We have different types of section in memory

Method Area: in this method area all class level information like as class name, its immediate parent class name , methods and variables and access specifier used with variable and method etc stored in method area as well as store static variable and its default value stored in method area.
Heap Section: in this section memory all objects stored in heap section of memory as well as all shared resources stored in heap section means if we perform any thing using new keyword in java it is stored in heap section.
Stack Area: so JVM create separate stack memory for every thread in java.
PC Register: store address of current execution instruction of thread

Q. what JRE and how it works?

Q. what is diff between JDK JRE and JVM?

JDK	**JRE**	**JVM**
JDK stands for java development kit	JRE stands for java run time enviourment	JVM stands java virtual machine
JDK is tool which provide enviourment to us to developer ,compile and run java program	JRE only provide environment to us to execute java program not compile	JVM is logical existence in JRE which is used for manage the all activities of java program execution
JDK use at dev environment because in dev environment we need to write code , compile it and execute it means here we need compiler and JVM etc Means in JDK contain compiler, JVM and JRE	JRE refer at test environment or prod environment because in test enviourment not need to write code and compile code just need to execute code and test it as well as JRE also use in production environment because here only program execute not write code	