**Java Programming Notes**

* **Author** : James Gosling
* **Vendor** : Sun microsystem
* **Old** **Name** : OAK
* **Present** **Name** : Java
* **Symbol** : Coffee Cup with Saucer
* **Slogan** : WORA (Write once run any where)
* **Language** **Type** : Open Source
* **Operating** **System** : Any Operating System
* **Developed** **From** : C and C++
* **Extensions** : .java, .class, .jar

**Execution Scenario of Java**

JVM

Byte code

Source Code

Compile

.java .class

Machine Code

Output

**Features of Java:-**

1. Simple:- Java is Simple for C or C++ learner , because all complexities of C and C++ are reduced in java.
2. Open Source:- Java is open source that means source code of java is available for user and user can modify source code.
3. Platform Independent:- Java is platform independent that means you can run java program on any operating system.
4. Object Oriented:- Java programming language is an object oriented programming language.
5. High Performance:- The performance of java programming is better than C and c++.
6. Write once run anywhere
7. Architecture Neutral:- Java program is not executed under operating system. Java program is executed under java runtime environment (JRE).
8. Multithreaded:- Java programming language supports concept of multithreading.
9. Web Application Development :- You can develop web application by using java programming language.

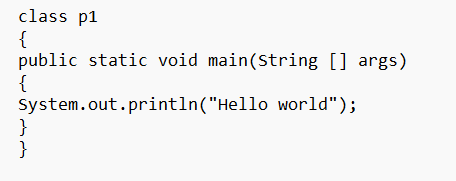
**Applications Developed using Java :-**

1. Desktop Application e.g. media player, anti-virus .. etc.
2. Web application e.g. irctc.co.in.
3. ERP (Enterprise Resource planning) like banking solution e.g. Finacle.
4. Mobile application development.
5. Embedded system and robotics
6. Game development.

Class:- Class is blueprint of object. Class is a collection of variables and method.

Object:- object is a physical property which have

It is used to take input from command line.



Access Modifier Modifier Return Type Method Name

Class Object

Method

**How to take input from user in java:-**

scanner Is a class in java for input in java we use the Scanner class object. Scanner is a class available in java.util packages.

**package**:- package is a collection of classes, interfaces and sub-package.

**Code** :- import java.util.Scanner;

Classname objectname = new Classname();

Scanner sc= new Scanner(System.in);

for int input:-

int a;

a =sc.nextInt();

for float input:-

float b;

b= sc.nextFloat();

for double input:-

double b;

b=sc.nextDouble();

for String input :-

String name ;

name =sc.nextLine();

**Decision Controls in java** :- Decision control are used to decision making.In java programming language there are following types of decision controls:-

1. if statement

2.if-else statement

3.ladder if else statement

4.switch statement

1.**if statement**:- it is keyword which work as decision control.We attach a condition with if block code will executed otherwise no code will be executed.

if(condition){

//code

}

2. if-else statement:- if - else is the variration of if statement.we attach a condition with if statement,if given condition is true then if block code will executed and if given condition is false then else block code will executed .

if (condition){

//code

}else{

//code

}

Ternary Operator (?):-Ternary operator is the alternate of if-else,(expression1)?(expression2):(expression3);

if expression1 is true then expression2 will executed and if expression1 is false then expression3 will executed.

08/02/2023

ladder if else:- If you have many conditions and you want to execute code based on those conditions,them you can use ladder if-else.

if (condition1){

//code 1 executed

}else if (condition2){

//code2 executed

}else{

//code3 executed

}

(expression1)?(kuch karenge):(kuch aur karenge);

**Switch statement** :- switch is keyword,which work as a case control.It is used to menu based program.

switch (expression) //int or char or String

{

case1:

//code

break;

case2:

//code

break;

case3:

//code

break;

default:

//code

}

**About the loop control** :- If you have a block of code which you want to execute repeatedly up to given condition is true ,then you can use a loop comtrol .in java programming language there are four types of loops controls:-

1.while loop

2.for loop

3. do-while loop

4. for each loop

**1.while loop:-** while is a keyword which works as loop control.

Initialization of loop counter;

while(condition){

//code

updation of loop counter

}

**For Loop**:- for is a keyword which works as loop control. Working of for loop is same as while but syntax is different.



For(Initialization; Condition; updation)

{

//



}

Do – while loop:-

Initialization of loop counter ;

Do

{

//code updation of loop counter;

}

While(Condition);

**Task:-**

1. Write a program to find compound interest.
2. Write a program to find area and perimeter of circle.
3. Write a program to check the given Armstrong number or not.

**For each :-** for each loop in java is used to traverse elements of a collection of like array.

**Array in java:-** array is a collectionof similar data types, that means an array can store multiple values of similar datatypes .

**How to create array in java :-**

Datatype [] arrayname= new datatype[size];

E.g.

int [] list=new int[10];

**Initialization of array:-**

int [] x={10,20,30,40,50};

x[0]=10;

x[1]=20;

x[2]=30;

x[3]=40;

x[4]=50; 10 20 30 40 50

**Memory Representation of array**

**How to take input from a user for an array?**

**Code segment:-**

int [] x=new int [5];

int i;

Scanner sc=new Scanner(System.in);

System.out.println(“Enter five number : ”);

For (i=0; i<5; i++)

{

x[i]=s.nextInt();

}

Program:-

// how to take array form user

// Write a program to take 10 numbers as input. store these numbers in array and find sum and average.

import java.util.\*;

class Arrays

{

public static void main(String[] args)

{

int[] list= new int[10];

int i,sum=0;

double avg;

Scanner sc=new Scanner(System.in);

System.out.println("Enter ten number to the list");

for (i=0; i<10; i++)

{

list[i]=sc.nextInt();

sum=sum+list[i];

}

avg=(float)sum/10;

System.out.println("Total array list : "+sum);

System.out.println("Total array list : "+avg);

}

}

**String in java :-** String is a class in java. The object of string class is used to store sequence of characters.

String str=”Softpro India”;

toUpperCase():- toUpperCase() method of String class is used to convert String into upper case (Capital letters).

toLowerCase():- toLowerCase() method of string class is used to convert String into lower case (small letter).

Length():- length() method of string class is used to find length of string.

Equals():- equals() method of string class is used

**EqualsIgnoreCase():-** this method of string class compare two Strings for equality and this method

Split():- split() method of String class split String into substrings. This method return array of string.

e.g. String str=”He is a good boy. ”;

String [] words=str.split(“ ”);

Words[0]=”He”;

Words[1]=”is”;

Words[2]=”a”;

Words[3]=”good”;

Words[4]=”boy”;

**Replace():-** replace() method of string class replace one String with another string in given String.

Replace(fw, rw);

**Concept of method in java:-** method is a named block of code, which perform specific task.

If you have a block of code which required at different locations of program, then you can create a method of that code and call it from desired locations.

By using method you can avoid to write same code over and over. Method is used to achieve modularity.

**How to create a method in java?**

<modifier> <return type> method name(parameters)

{

//

}

Public int add( int x, int y)

{

Return (x+y);

}

**Types of methods in Java:-**

**In java programming language there are two types of methods:-**

1. **Static methods:-** static are created by using static modifier. These methods are also called class methods. There is no need of object to call static methods.
2. **No Static methods:-**  non-static methods are created without using static modifiers. There is a need objects to call non-static methods.

**Notes:-** Methods are created within class and outside of main() method.

**OOPS:-** OOPS stands for object-oriented programming system. It is a mechanism of software development. OOPS has four Pillars:-

1. **Abstraction:-** Abstraction is a mechanism to hide the functionalities of an object.
2. **Encapsulation:-** Encapsulation in a mechanism to wrap properties and functionalities in a single unit. That single unit is called an object.
3. **Inheritance:-** Inheritance is a mechanism to create new product by using existing product.
4. **Polymorphism:-** term polymorphism means one thing many forms.

Note:- Any Programming language which follows these four pillars of OOPS concepts.

**Class:-** Class is blue print of object. Class is a container of variable and method. class is created using “Class” keyword followed by classname. The body of class is enclosed within braces.

Class Employee

{

//Variables and method

}

// This program demonstrate concept of class

class MyClass

{

void sayHello()

{

System.out.println("Hello World");

}

}

class CLassDemo1

{

public static void main(String [] args)

{

MyClass m=new MyClass();

m.sayHello();

}

}

**Second Program:-**

// This program demonstrate concept of class

class MyClass

{

void sayHello(String name)

{

System.out.println("Hello" +name);

}

}

class ClassDemo1

{

public static void main(String [] args)

{

MyClass m=new MyClass();

m.sayHello("Harsh Pandey");

}

}

**Third program :-**

class Employee

{

int empid; //Instance variable

String empname; //Instance variable

double salary; //Instance variable

void setValue(int eid, String ename, double sal)

{

empid=eid;

empname=ename;

salary=sal;

}

void display()

{

System.out.println("Employee id="+empid);

System.out.println("Employee name="+empname);

System.out.println("Employee Salary="+salary);

}

}

class ClassDemo2

{

public static void main(String [] args)

{

Employee e1=new Employee();

e1.setValue(1001,"Brijesh Mishra",40000.0);

e1.display();

Employee e2=new Employee();

e2.setValue(1002,"Harsh Pandey",60000.0);

e2.display();

}

}

**Concept of Constructor:-**Constructor is a special method, which is used to initialize variable.

1. Constructor name is same as class name.
2. Constructor has no return type.
3. Constructor call automatically as soon as object is

**Program 1.**

class Student

{

int rollno;

String name;

double fee;

Student(int rno, String nm, double f) //constructor

{

rollno =rno;

name=nm;

fee=f;

}

void display()

{

System.out.println("Roll no = "+rollno);

System.out.println("Name = "+name);

System.out.println("Fee = "+fee);

}

}

class ClassDemo3

{

public static void main(String [] args)

{

Student s=new Student(1001,"Harsh",10000.00);

s.display();

}

}

**Inheritance In Java:-** In Inheritance you can create a new class by using existing class. Existing class is called base class and new created class is called derived class.

Base Class A class A {

//Variable and methods

}

Class B extends A { //Variable and methods

Derived Class B }

The concept of inheritance is also called ‘Reusability’

**Types of Inheritance in Java:-**

In java programming language there are three types of inheritance are available:-

1. **Single Inheritance:-** In single inheritance there is a single base class and single derived class.

Base Class A class A {

//Variable and methods

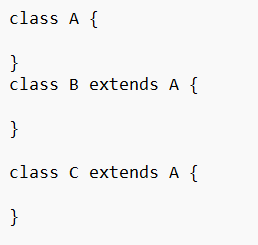
}

Class B extends A { //Variable and methods

Derived Class B }

1. **Hierarchical Inheritance:-** In hierarchical Inheritance there is a single base and multiple derived classes.

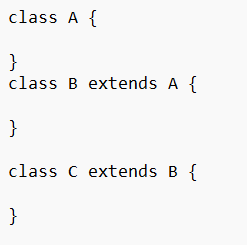
Class A

****

Class B Class C

**3. Multi-level Inheritance:-**

Class A

 Class C

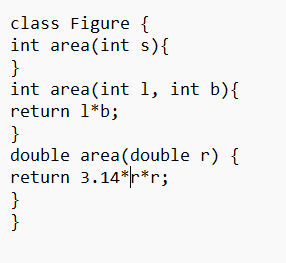
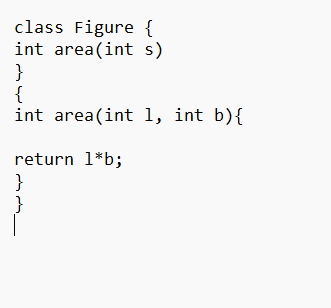
Polymorphism:- the term ‘Polymorphism’ means one thing many form. In java there are two types of polymorphism

1. **Compile time polymorphism (method overloading)**
2. **Run time polymorphism (method overriding)**

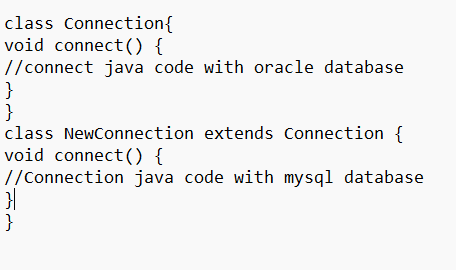
**Method overloading :-** in java you can create multiple methods with same name in same class, but their parameters should be different. Basse on method parameters it is decided at compilation time that which method call form where. This is called method overloading .

Method parameters can be different in two types:-

1. **Number of parameters can be different.**
2. **Types of parameters can be different.**
3. **Method overloading.**

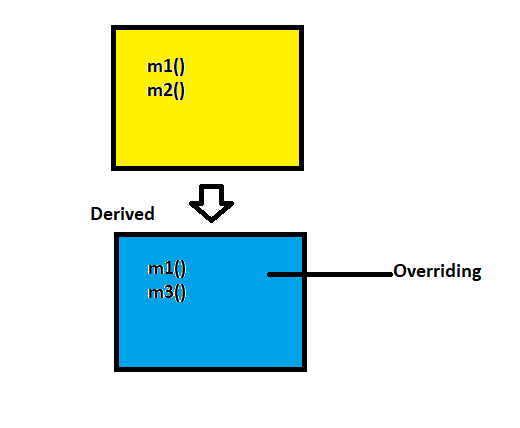
** **

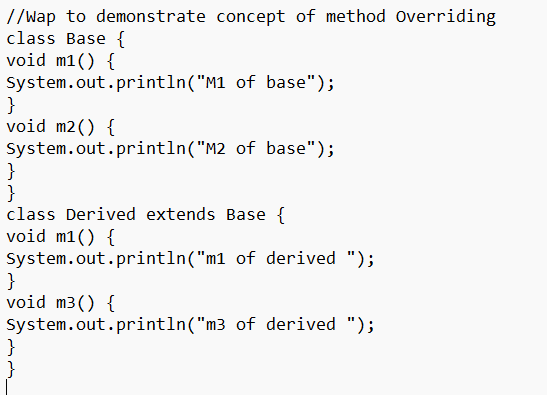
1. **Method overriding :-** Re writing of base class method into derived class is called method overriding.



**There are following rules to perform method overriding:-**

1. Class must be inherited.
2. Base class method name and derived class method name must be same.
3. Base class method parameters and derived class method parameters must be same.
4. Base class method return type and derived class method return type can be.





class Overriding {

public static void main(String [] args){

Base b=new Base();

b.m1();

b.m2();

Derived d=new Derived();

d.m1();

d.m2();

d.m3();

}

}

**Exception Handling In Java:-** The Dictionary meaning of exception is abnormal termination. When exception is occurred then program is terminated abnormally and rest of code is not executed.

**In Java programming language there are three types of exception:-**

1. **Checked Exceptions**
2. **Unchecked Exceptions**
3. **Errors**

* **Checked Exceptions :-** Checked Exception are those exception which are identified by compiler at compilation time. E.g. CLassNotFoundException, IOException, SQLException, FileNotFoundException, InterruptedException… etc.
* **Unchecked Exceptions:**- Unchecked Exception are those Exception which are identified at run time. E.g. NullPointerException, ArithmeticException, InputMisMatchException, ArrayIndexOutOfBoundsException.. etc.
* **Errors:**- Errors are occurred due to lack of system resources. E.g. AwtError, FileNotFoundError, IOError,…. Etc..