**POLYMORPHISM**

**WE CAN EXIBIT THE NATURE OF POLYMORPHISM IN TWO WAYS**

**1.METHOD OVER LOADING**

**2.METHOD OVERRIDING**

**1.METHOD OVERLOADING**

 **Assigning additional work to an existing method other than the original work is known as method overloading…**

**We CAN overload a Method in Three cases…**

**\* When number of parameters/arguments are differ…**

**\* When data types of the arguments are differ…**

* **When the order of the parameters are differ**

Ex:1

**Class ABC**

**{**

**Public void add(int a,int b)**

**{**

**……**

**….**

**}**

**Public void add(int a,int b,int c)**

**{**

**……..**

**…….**

**}**

**}**

Ex2:

**Class ABC**

**{**

**Public void add(int a,int b)**

**{**

**……**

**….**

**}**

**Public void add(float a,float b)**

**{**

**……..**

**…….**

**}**

**}**

Ex3:

**Class ABC**

**{**

**Public void add(int a,float b)**

**{**

**……**

**….**

**}**

**Public void add(float a,int b)**

**{**

**……..**

**…….**

**}**

**}**

Ex4: Invalid

**Class ABC**

**{**

**Public int add(int a,int b)**

**{**

**……**

**….**

**}**

**Public void add(int a,int b)**

**{**

**……..**

**…….**

**}**

**}**

**Example:**

**class Demo**

**{**

**public void add(int a,int b)**

**{**

**System.out.println("Addition is "+(a+b));**

**}**

**public void add(int a,int b,int c)**

**{**

**System.out.println("Addition is "+(a+b+c));**

**}**

**public void add(int a,float b)**

**{**

**System.out.println("Addition is "+(a+b));**

**}**

**}**

**class OverLoading**

**{**

**public static void main(String args[])**

**{**

**Demo obj=new Demo();**

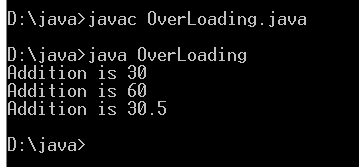
**obj.add(10,20);**

**obj.add(10,20,30);**

**obj.add(10,20.5f);**

**}**

**}**

****

**Method overriding:**

class Demo

{

public void Show()

{

System.out.println("This is from show in Demo");

}

}

class Demo1 extends Demo

{

public void Show()

{

System.out.println("This is from show in demo1");

}

}

class OverLoading

{

public static void main(String args[])

{

Demo1 obj1=new Demo1();

Demo obj=new Demo();

obj1.Show();

obj.Show();

}

}

