

Assignment: 2

Name: Shrenik Mehar

Batch: C3 `Roll No: 86

Course: OOPS Date: 20-7-21

Aim: Method Overloading.

1. In the below Class X, is 'method' properly overloaded? Explain with reason.

```
class X {  
    int method(int i, int d) {  
        return i+d;  
    }  
    static int method(int i, double d) {  
        return (int)(i+d);  
    }  
    double method(double i, int d) {  
        return i+d;  
    }  
    static double method(double i, double d) {  
        return i+d;  
    }  
}
```

If a class has multiple methods having same name but different in parameters, it is known as Method Overloading.

Here are 4 methods named method they are not duplicated, they are overloaded because according to the definition they are overloaded because they are of different data types.

2. What will be the outcome of the below program?

```
public class MainClass {  
    double overloadedMethod(double d) {  
        return d *= d;  
    }  
}
```

```

    }
    int overloadedMethod(int i) {
        return overloadedMethod(i *= i);
    }
    float overloadedMethod(float f) {
        return overloadedMethod(f *= f);
    }
    public static void main(String[] args) {
        MainClass main = new MainClass();
        System.out.println(main.overloadedMethod(100));
    }
}

```

Output will be: It will throw java.lang.StackOverflowError at run time. Because, overloadedMethod(int) keeps calling itself.

3. What will be the output of the following program?

```

public class MainClass {
    static void method(Integer i) {
        System.out.println(1);
    }
    static void method(Double d) {
        System.out.println(2);
    }
    static void method(Number n) {
        System.out.println(4);
    }

    static void method(Object o) {
        System.out.println(5);
    }
    public static void main(String[] args) {
        method((short) 12);
    }
}

```

Output will be: 4

Because methods are overloaded and method of data type short is called which is also known as number

4. Create a class to print the area of a square and a rectangle. The class has two methods with the same name but different number of parameters. The method for printing the area of the rectangle has two parameters

which are length and breadth respectively while the other method for printing area of square has one parameter which is side of square.

```
class Area

{

    double a, s, l, b;

    void area(double s)

    {

        this.a=s*s;

        System.out.println("Area of Square is "+a+" units sqr");

    }

    void area(double l, double b)

    {

        this.a=l*b;

        System.out.println("Area of Rectangle is "+a+" units sqr");

    }

}

public class MainArea

{

    public static void main(String args[])

    {

        Area c1=new Area();

        Area c2=new Area();

        c1.area(7.8);
```

```
c2.area(6.9,4.20);  
  
}  
  
}
```

```
Area of Square is 60.839999999999996 units sqr  
Area of Rectangle is 28.980000000000004 units sqr
```

5. Create a class 'Degree' having a method 'getDegree' that prints "I got a degree". It has two subclasses namely 'Undergraduate' and 'Postgraduate' each having a method with the same name that prints "I am an Undergraduate" and "I am a Postgraduate" respectively. Call the method by creating an object of each of the three classes.

```
class Degree  
{  
    void getDegree()  
    {  
        System.out.println("I got Degree");  
    }  
}  
  
class Undergraduate extends Degree  
{  
    void getDegree()  
    {  
        System.out.println("I am Undergraduate");  
    }  
}
```

```
    }  
}  
  
class Postgraduate extends Degree  
{  
    void getDegree()  
    {  
        System.out.println("I am Postgraduate");  
    }  
}  
  
public class MainDegree  
{  
    public static void main(String[] args)  
    {  
        Degree d = new Degree();  
        Undergraduate ug = new Undergraduate();  
        Postgraduate pg = new Postgraduate();  
        d.getDegree();  
        ug.getDegree();  
        pg.getDegree();  
    }  
}
```

```
I got Degree
I am Undergraduate
I am Postgraduate
|
```

6. A boy has his money deposited \$1000, \$1500 and \$2000 in banks-Bank A, Bank B and Bank C respectively. We have to print the money deposited by him in a particular bank. Create a class 'Bank' with a method 'getBalance' which returns 0. Make its three subclasses named 'BankA', 'BankB' and 'BankC' with a method with the same name 'getBalance' which returns the amount deposited in that particular bank. Call the method 'getBalance' by the object of each of the three banks.

```
class Bank
```

```
{
    int getBalance()
    {
        return 0;
    }
}
```

```
class BankA extends Bank
```

```
{
    int getBalance(int bal)
    {
        return bal;
    }
}
```

```
    }  
}
```

```
class BankB extends Bank
```

```
{  
    int getBalance(int bal)  
    {  
        return bal;  
    }  
}
```

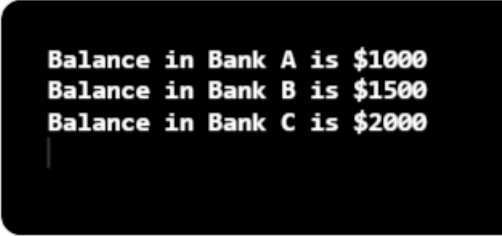
```
class BankC extends Bank
```

```
{  
    int getBalance(int bal)  
    {  
        return bal;  
    }  
}
```

```
public class ShrenikBal
```

```
{  
    public static void main(String args[])
```

```
{  
    BankA a = new BankA();  
    BankB b = new BankB();  
    BankC c = new BankC();  
    System.out.println("Balance in Bank A is $" + a.getBalance(1000));  
    System.out.println("Balance in Bank B is $" + b.getBalance(1500));  
    System.out.println("Balance in Bank C is $" + c.getBalance(2000));  
}  
}
```

A terminal window with a black background and white text. It displays three lines of output: "Balance in Bank A is \$1000", "Balance in Bank B is \$1500", and "Balance in Bank C is \$2000". A vertical cursor is visible on the line "Balance in Bank C is \$2000".

```
Balance in Bank A is $1000  
Balance in Bank B is $1500  
Balance in Bank C is $2000
```