

1. Bank is a class that provides method to get the rate of interest. But, rate of interest may differ according to banks. For example, SBI, ICICI and AXIS banks are providing 8.4%, 7.3% and 9.7% rate of interest. Write a Java program for above scenario.
2. Develop a Java application to generate Electricity bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading. Compute the bill amount using the following tariff.
 - First 100 units – Rs. 1 per unit
 - 101-200 units – Rs. 2.50 per unit
 - 201 -500 units – Rs. 4 per unit
 - > 501 units – Rs. 6 per unit
3. Create a java program to construct the volume of Box using default constructor method.
4. Write a Java program to create a class Student and create constructor which assigns the values for the student details such as student name, register number, and five subject marks. Calculate the total and average of five subject marks and display the marks and average.
5. Create a class name 'Overload'. Write a program to assign the values by different number of arguments using a single function.
6. Bring out the situation in which member names of a subclass hide members by the same name in the super class. How it can be resolved? Write Suitable code in Java and Implement above scenario with the Parametrized Constructor (accept int type parameter) of the Super Class can be called from Sub Class Using super () and display the input values provided.
7. Define an Employee class with suitable attributes having getSalary() method, which returns salary withdrawn by a particular employee. Write a class Manager which extends a class Employee, **override the getsalary()** method, which will return salary of manager by adding traveling _allowance, house rent allowance etc.
8. Develop a program that uses Multilevel Inheritance concepts to compute a student's grades in six subjects. The total and aggregate are then calculated, and the student's grade is displayed. If the student achieves an aggregate of more than 75%, the grade is Distinction. If the aggregate is between 60 and 75, the grade is First Division. If the aggregate is between 50 and 60, the grade is Second Division. If the aggregate is between 40 and 50, the grade is Third Division. Otherwise, the grade is FAIL.
 - Sample Input & Output:
 - Enter the marks in python: 90
 - Enter the marks in c programming: 91

Enter the marks in Mathematics: 92

Enter the marks in Physics: 93

Enter the marks in Chemistry: 92

Enter the marks in Professional Ethics: 93

Total= 551

Aggregate = 91.83

Class: DISTINCTION

9. Generate a Java code to perform simple arithmetic operations and to throw Arithmetic Exception for Division-by-Zero.
10. Bank is a class that provides method to get the rate of interest. But, rate of interest may differ according to banks. For example, SBI, ICICI and AXIS banks are providing 8.4%, 7.3% and 9.7% rate of interest. Write a Java program for above scenario.

```
class Bank
{
    double p,n,r,Sintrest,Aintrest,lintrest;
    void get(double pr,double no)
    {
        p=pr;
        n=no;
    }
    void SBI(double r)
    {
        Sintrest=p*n*r/100;
    }
    void ICICI(double r)
    {
        lintrest=p*n*r/100;
    }
    void Axis(double r)
    {
        Aintrest=p*n*r/100;
    }
    void display()
    {
        System.out.println("SBI intrest: "+Sintrest);
        System.out.println("ICICI intrest: "+lintrest);
        System.out.println("Axis intrest: "+Aintrest);
    }
    public static void main(String[] args)
    {
        Bank b=new Bank();
        b.get(100000,5);
        b.SBI(8.4);
        b.ICICI(7.3);
        b.Axis(9.7);
    }
}
```

```

        b.display();
    }
}

```

11. Develop a Java application to generate Electricity bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading. Compute the bill amount using the following tariff.

First 100 units – Rs. 1 per unit
 101-200 units – Rs. 2.50 per unit
 201 -500 units – Rs. 4 per unit
 > 501 units – Rs. 6 per unit

```

class EB_Bill
{
    int cno;
    String cname;
    int pmr;
    int cmr;
    int u;
    double amt=0.0;
    void get(int consumerno, String consumername, int previousmonthreading, int
currentmonthreading)
    {
        cno=consumerno;
        cname=consumername;
        pmr=previousmonthreading;
        cmr=currentmonthreading;
    }
    void cal()
    {
        u=cmr-pmr;
        System.out.println("Consumer Number = "+cno);
        System.out.println("Consumer Name = "+ cname);
        System.out.println("Previous Month Reading = "+pmr);
        System.out.println("Current Month Reading = "+cmr);
        System.out.println("Unit used this month = "+u);
    }
    void tariff()
    {
        if(u<=100)
        {
            amt=u*1.0;
            System.out.println("Amount="+amt);
        }
        else if(u>=101&&u<=200)

```

```

        {
            amt=u*2.50;
            System.out.println("Amount="+amt);
        }
        else if(u>=201&&u<=500)
        {
            amt=u*4.0;
            System.out.println("Amount="+amt);
        }
        else if(u>=501)
        {
            amt=u*6.0;
            System.out.println("Amount="+amt);
        }
        else
        {
            System.out.println("invalid input");
        }
    }
    public static void main(String[]args)
    {
        EB_Bill obj=new EB_Bill();
        obj.get(100201,"Hema",1850,2550);
        obj.cal();
        obj.tariff();
    }
}

```

12. Create a java program to construct the volume of Box using default constructor method.

```

class Box
{
    double width;
    double height;
    double depth;
    Box()
    {
        width = 10.5;
        height = 20.5;
        depth = 5.5;
    }
    void volume()
    {
        double v= width * height * depth;
        System.out.print("Volume of Box = "+v);
    }
}

```

```

    public static void main(String[] args)
    {
        Box myBox = new Box();
        myBox.volume();
    }
}

```

Argument Constructor

```

class Box
{
    double width;
    double height;
    double depth;
    Box(double w, double h, double d)
    {
        width = w;
        height = h;
        depth = d;
    }
    void volume()
    {
        double v= width * height * depth;
        System.out.print("Volume of Box = "+v);
    }

    public static void main(String[] args)
    {
        Box myBox = new Box(10.5, 15.5, 20.5);
        myBox.volume();
    }
}

```

13. Write a Java program to create a class Student and create constructor which assigns the values for the student details such as student name, register number, and five subject marks. Calculate the total and average of five subject marks and display the marks and average.

```

class Student
{
    String name;
    int regno;
    int total;
    double average;
    int m1,m2,m3,m4,m5;
    Student(String student_name,int number,int a,int b,int c,int d,int e)

```

```

    {
        name=student_name;
        regno=number;
        m1=a;
        m2=b;
        m3=c;
        m4=d;
        m5=e;
    }
    void calculate()
    {
        total=(m1+m2+m3+m4+m5);
        average=total/5;
    }
    void display()
    {
        System.out.println("Name:"+name);
        System.out.println("Reg.No:"+regno);
        System.out.println("Mark1:"+m1);
        System.out.println("Mark2:"+m2);
        System.out.println("Mark3:"+m3);
        System.out.println("Mark4:"+m4);
        System.out.println("Mark5:"+m5);
        System.out.println("Average of above marks:"+average);
    }
    public static void main(String[] args)
    {
        Student obj=new Student("Vijaya Lakshmi",19210200,90,80,85,90,99);
        obj.calculate();
        obj.display();
    }
}

```

14. Create a class name 'overload'. Write a program to assign the values by different number of arguments using a single function.

```

class Overload
{
    void add(int a, int b)
    {
        int sum1 = a+b;
        System.out.println("Sum of two numbers: " + sum1);
    }

    void add(int a, int b, int c)
    {
        int sum1 = a+b+c;
    }
}

```

```

        System.out.println("Sum of three numbers: " + sum1);
    }
    public static void main(String[] args)
    {
        Overload obj = new Overload();
        obj.add(10, 20);
        obj.add(10, 20, 30);
    }
}

```

15. Bring out the situation in which member names of a subclass hide members by the same name in the super class. How it can be resolved? Write Suitable code in Java and Implement above scenario with the Parametrized Constructor (accept int type parameter) of the Super Class can be called from Sub Class Using super () and display the input values provided.

```

class A
{
    int i,j;
    A(int a,int b)
    {
        i=a;
        j=b;
        System.out.println("baseclass constructor");
    }
    void show()
    {
        System.out.println("i = "+i+" j = "+j);
    }
}
class B extends A
{
    int k;
    B(int a,int b,int c)
    {
        super(a,b);
        k=c;
        System.out.println("subclass constructor");
    }
    void show()
    {
        super.show();
        System.out.println("i = "+i+" j = "+j+" k = "+k);
    }
}
class Override
{

```

```

        public static void main(String arg[])
        {
            B b=new B(100,200,300);
            b.show();
        }
    }

```

16. Define an Employee class with suitable attributes having getSalary() method, which returns salary withdrawn by a particular employee. Write a class Manager which extends a class Employee, override the getsalary() method, which will return salary of manager by adding traveling_allowance, house rent allowance etc.

```

class Employee
{
    int sal;
    void getSalary()
    {
        sal=50000;
    }
}
class manager extends Employee
{
    int ta,hra,totalsal;
    void getSalary()
    {
        super.getSalary();
        ta=10000;
        hra=1500;
        totalsal=sal+ta+hra;
    }
    void display()
    {
        System.out.println("Employee salary : "+sal);
        System.out.println("Manager salary: "+totalsal);
    }
    public static void main(String a[])
    {
        manager m=new manager();
        m.getSalary();
        m.display();
    }
}

```

17. Develop a program that uses Multilevel Inheritance concepts to compute a student's grades in six subjects. The total and aggregate are then calculated, and the student's grade is displayed. If the student achieves an aggregate of more than 75%, the grade is Distinction. If the aggregate is between 60 and 75, the grade is First Division. If the

aggregate is between 50 and 60, the grade is Second Division. If the aggregate is between 40 and 50, the grade is Third Division. Otherwise, the grade is FAIL.

Sample Input & Output:

Enter the marks in python: 90

Enter the marks in c programming: 91

Enter the marks in Mathematics: 92

Enter the marks in Physics: 93

Enter the marks in Chemistry: 92

Enter the marks in Professional Ethics: 93

Total= 551

Aggregate = 91.83

Class: DISTINCTION

```
class Marks
{
    int pythonMarks;
    int CMarks;
    int CPPMarks;
    int JavaMarks;
    void input(int m1, int m2, int m3, int m4)
    {
        pythonMarks = m1;
        CMarks = m2;
        CPPMarks = m3;
        JavaMarks = m4;
    }
}
class Average extends Marks
{
    int total;
    double aggregate;
    void calculate()
    {
        total = pythonMarks + CMarks + CPPMarks + JavaMarks;
        aggregate = total / 4.0;
    }
    void display()
    {
        System.out.println("Total = " + total);
        System.out.println("Aggregate = " + aggregate);
    }
}
class Grade extends Average
{
    void grade_display()
```

```

{
    if (aggregate > 75)
    {
        System.out.println("DISTINCTION");
    }
    else if (aggregate >= 60 && aggregate < 75)
    {
        System.out.println("First Division");
    }
    else if (aggregate >= 50 && aggregate < 60)
    {
        System.out.println("Second Division");
    }
    else if (aggregate >= 40 && aggregate < 50)
    {
        System.out.println("Third Division");
    }
    else
    {
        System.out.println("Fail");
    }
}
public static void main(String[] args)
{
    Grade obj=new Grade();
    obj.input(95,90,95,100);
    obj.calculate();
    obj.display();
    obj.grade_display();
}
}

```

18. Generate a Java code to perform simple arithmetic operations and to throw Arithmetic Exception for Division-by-Zero.

```

import java.util.*;
class ArithExce
{
    public static void main(String args[])
    {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter a and b values");
        int a=s.nextInt();
        int b=s.nextInt();
        try

```

```

        {
            int x=a/b;
            System.out.println("output = "+x);
        }
        catch(Exception ex)
        {
            System.out.println("Division by zero error");
            System.out.println("Change the b value");
        }
        finally
        {
            System.out.println("Always executed");
        }
    }
}

```

19. Generate a Java code to find the sum of N numbers using array and throw `ArrayIndexOutOfBoundsException` when the loop variable beyond the size N.

Sample Input : 5

1 2 3 4 5

Sample Output : 15

```

import java.util.*;
class Sumofarray
{
    public static void main(String arg[])
    {
        Scanner sc=new Scanner(System.in);
        int sum=0,i,n;
        System.out.println("enter no.of values:");
        n=sc.nextInt();
        System.out.println("enter array of elements:");
        int a[]=new int[n];
        try
        {
            for(i=0;i<5;i++)
            {
                a[i]=sc.nextInt();
                sum=sum+a[i];
            }
            System.out.println("sum:"+sum);
        }
        catch(Exception ex)
        {
            System.out.println("more than n values");
            System.out.println("change n value");
        }
    }
}

```

```
    }  
    finally  
    {  
        System.out.println("always executed");  
    }  
}  
}
```