Experiment No. 1 (a)

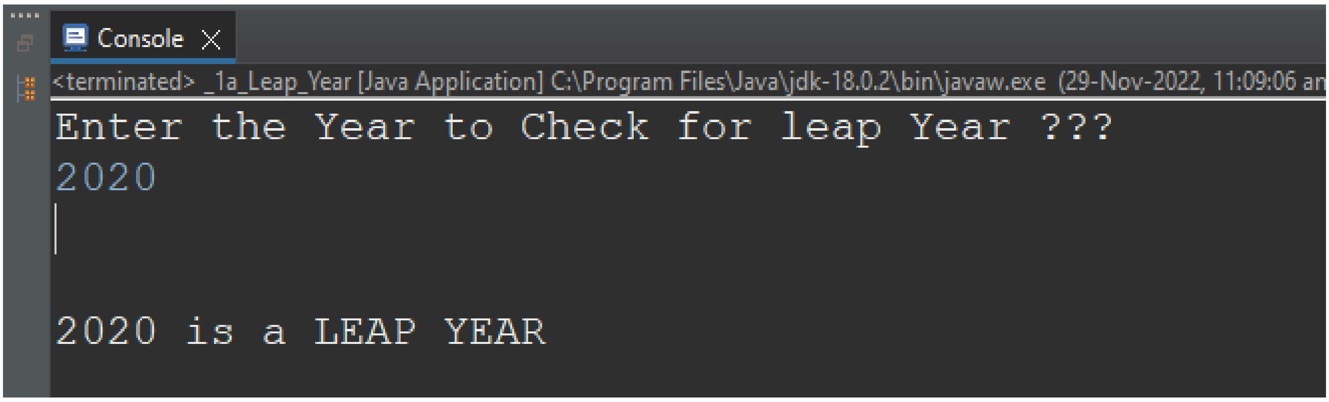
Aim

Java program to print whether the year entered by the user is Leap Year or Not.

Source code

```
package java_file;

import java.util.Scanner;

public class _1a_Leap_Year {

	public static void main(String[] args) {

		Scanner input=new Scanner(System.in);
		System.out.println("Enter the Year to Check for leap Year  ???");
		int year=input.nextInt();
		if(((year%4==0)&&(year%100!=0))||(year%400==0))
		{
			System.out.println("\n\n"+year+" is a LEAP YEAR");
		}
		else
		{
			System.out.println("\n\n"+year+" is a COMMON YEAR");
		}
	}

}
```

Output



```
<terminated> _1a_Leap_Year [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (29-Nov-2022, 11:09:06 an
Enter the Year to Check for leap Year ???
2020


2020 is a LEAP YEAR
```

Experiment No. 1 (b)

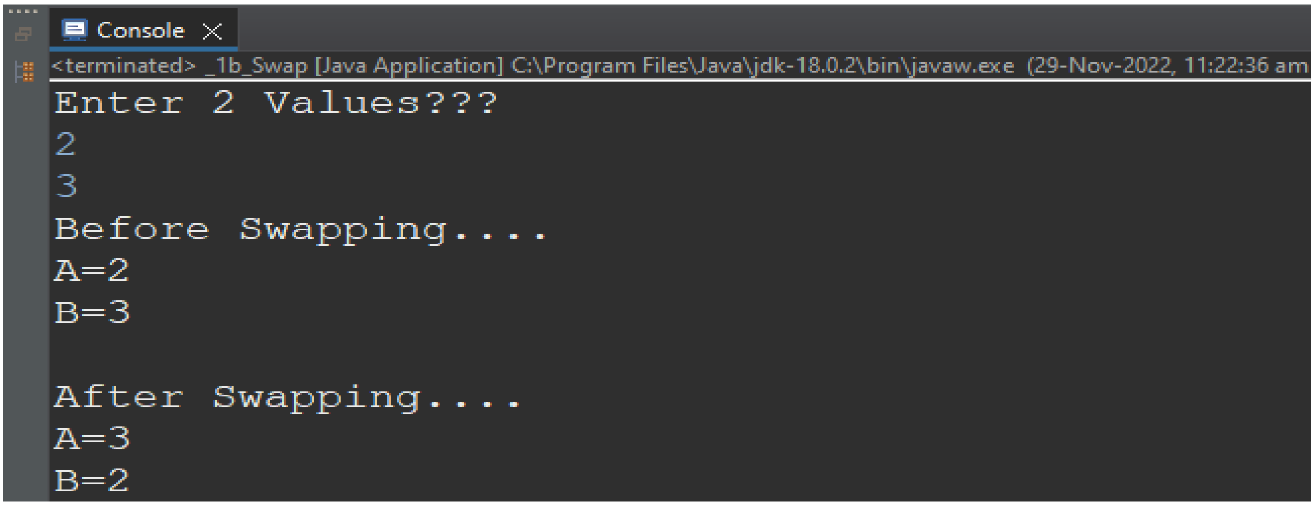
Aim

Java program to Swap 2 Numbers without using Third Variable.

Source code

```
package java_file;

import java.util.Scanner;

public class _1b_Swap {

    public static void main(String[] args) {

            Scanner input=new Scanner(System.in);

            System.out.println("Enter 2 Values???");
            int a=input.nextInt();
            int b=input.nextInt();
            System.out.print("Before Swapping....\n"+"A="+a+"\nB="+b);
            a=a+b;
            b=a-b;
            a=a-b;
            System.out.print("\n\nAfter Swapping....\n"+"A="+a+"\nB="+b);
    }

}
```

Output



```
<terminated> _1b_Swap [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (29-Nov-2022, 11:22:36 am
Enter 2 Values???
2
3
Before Swapping....
A=2
B=3

After Swapping....
A=3
B=2
```

Experiment No. 2 (a)

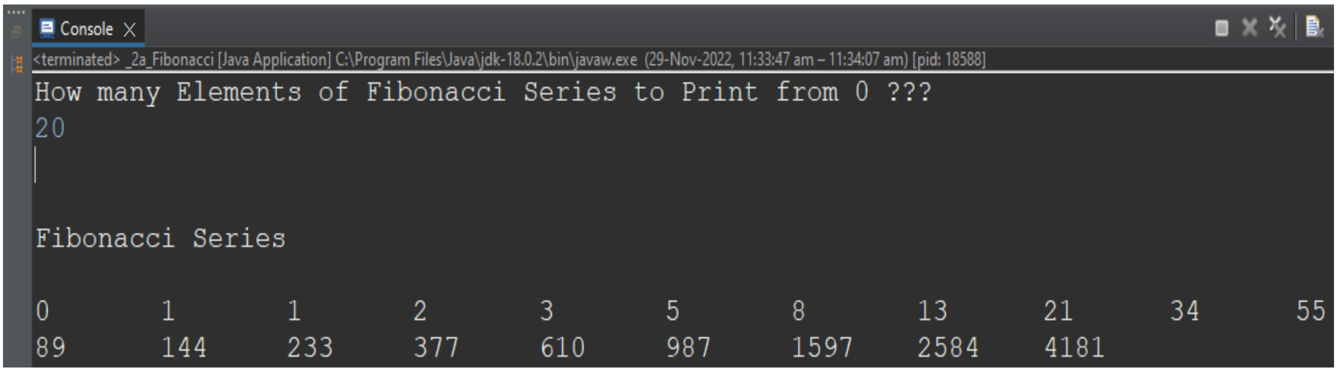
Aim

Java program to Print the Fibonacci Series.

Source code

```
package java_file;

import java.util.Scanner;

public class _2a_Fibonacci {

	public static void main(String[] args) {
		Scanner input=new Scanner(System.in);
		System.out.println("How many Elements of Fibonacci Series to Print from 0 ???");
		int a=input.nextInt();
		int full=10;
		int fib[]=new int[a];
		fib[0]=0;
		fib[1]=1;
		System.out.println("\n\nFibonacci Series\n");
		System.out.print("0\t1\t");

		for(int i=2;i<a;i++)
		{
			fib[i]=fib[i-1]+fib[i-2];
			System.out.print(fib[i]+"\t");
			if(i==full)
			{
				System.out.println();
				full+=10;
			}
		}
	}
}
```

Output



```
<terminated> _2a_Fibonacci [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (29-Nov-2022, 11:33:47 am – 11:34:07 am) [pid: 18588]
How many Elements of Fibonacci Series to Print from 0 ???
20



Fibonacci Series

0       1       1       2       3       5       8       13      21      34      55
89      144     233     377     610     987     1597    2584    4181
```

Experiment No. 2 (b)

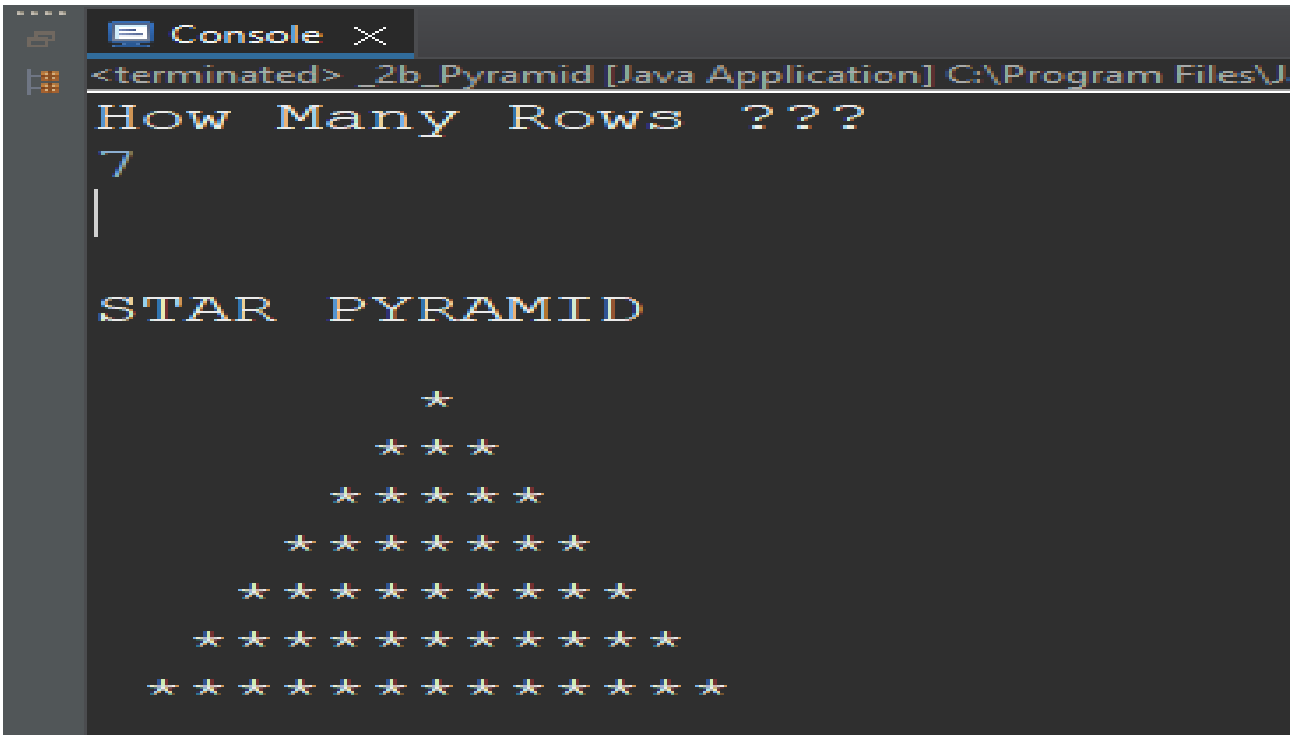
Aim

Java program to Print the Pyramid Star Pattern (Ask the user the Number of Rows using the Scanner class).

Source code

```
package java_file;

import java.util.Scanner;


public class _2b_Pyramid {

    public static void main(String[] args) {

            Scanner input=new Scanner(System.in);

            System.out.println("How Many Rows ???");

            int z=input.nextInt();

            int y=1;


            System.out.println("\n\nSTAR PYRAMID\n");

            for(int i=z;i>=1;i--)

            {

                    for(int j=1;j<=i;j++)

                            System.out.print(" ");

                    int k=1;

                    while(k<=y)

                    {

                            System.out.print("*");

                            k++;

                    }

                    y+=2;

                    System.out.println("");
            }
        }
    }
```

Output



```
<terminated> _2b_Pyramid [Java Application] C:\Program Files\J
How Many Rows ???
7


STAR PYRAMID

       *
      ***
     *****
    *******
   *********
  ***********
 *************
```

Experiment No. 3

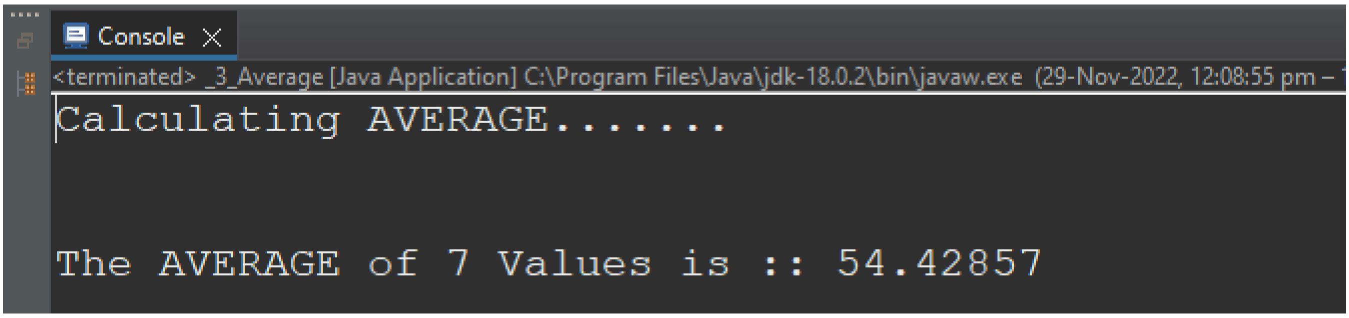
Aim

Java program to print Calculate Average of numbers using Arrays.

Source code

```
package java_file;


public class _3_Average {


    public static void main(String[] args) {

            float arr[]= {27,71,39,85,13,69,77};

            float temp=0;

            System.out.println("Calculating AVERAGE.......");

            for(int i=0;i<7;i++)

            {

                    temp=temp+arr[i];

            }

            float avg=temp/7f;

            System.out.println("\n\nThe AVERAGE of 7 Values is :: "+avg);

    }

}
```

Output



<terminated> _3_Average [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (29-Nov-2022, 12:08:55 pm –

```
Calculating AVERAGE.......


The AVERAGE of 7 Values is :: 54.42857
```

Experiment No. 4

Aim

Create a class with 2 methods input() and sum()

Input(): take user input, and call the method sum() then print the sum

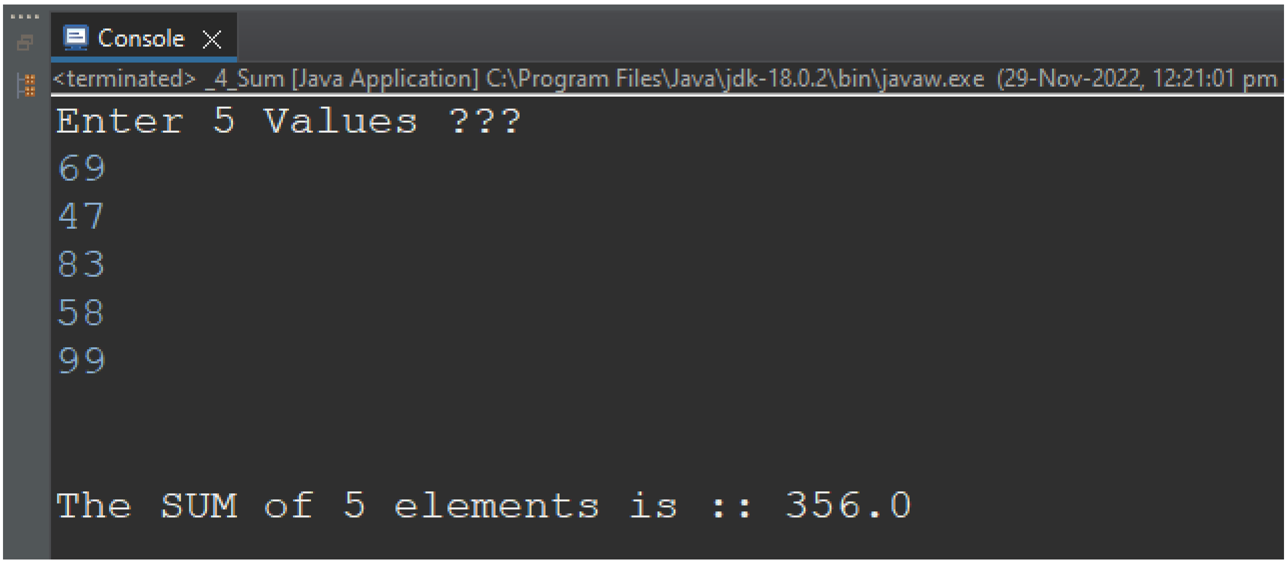
Sum(); calculate the sum and return the value to the input method.

Source code

```
package java_file;

import java.util.Scanner;


public class _4_Sum {

    float sum[]=new float[5];


    public static void main(String[] args) {

        _4_Sum obj=new _4_Sum();

        obj.input();

    }


    void input() {

        Scanner read=new Scanner(System.in);

        System.out.println("Enter 5 Values ???");

        for(int i=0;i<sum.length;i++) {

            sum[i]=read.nextFloat();

        }

        System.out.println("\n\nThe SUM of 5 elements is :: "+sum());

    }


    float sum() {
```

```
                float temp=0;

                for(int i=0;i<sum.length;i++) {

                        temp+=sum[i];

                }

                return temp;

        }

}
```

Output



```
<terminated> _4_Sum [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (29-Nov-2022, 12:21:01 pm
Enter 5 Values ???
69
47
83
58
99


The SUM of 5 elements is :: 356.0
```

Experiment No. 5

Aim

Define a class Employee with the following specifications:

Data Member:

empno, ename, basic, hra, da, netpay

Member Methods:

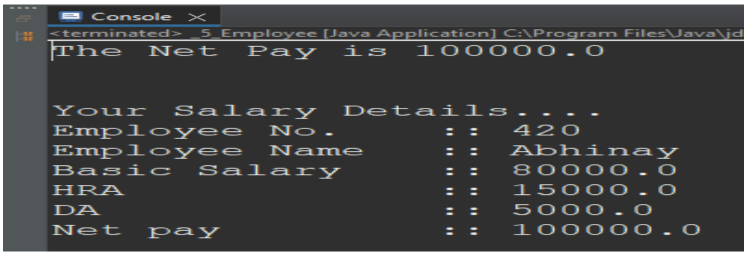
haveData() method to accept values for empno, ename, basic, hra, da & invoke the method calculate() for netpay.

dispData() method to display all the data members on the screen.

Source code

```
package java_file;


public class _5_Employee {


    int empno;

    String ename;

    float basic;

    float hra;

    float da;

    float netpay;


    public static void main(String[] args) {

            _5_Employee obj=new _5_Employee();

            obj.haveData(420,"Abhinay",80000,15000,5000);

            obj.dispData();

    }
    void haveData(int a, String b, float c, float d, float e) {
        empno=a;
        ename=b;
        basic=c;
        hra=d;
        da=e;
        calculate();
    }

    void calculate() {
        netpay=basic+hra+da;
        System.out.println("The Net Pay is "+netpay);
    }

    void dispData() {
        System.out.println("\n\nYour Salary Details....");
        System.out.println("Employee No. \t:: "+empno);
        System.out.println("Employee Name\t:: "+ename);
        System.out.println("Basic Salary \t:: "+basic);
        System.out.println("HRA          \t:: "+hra);
        System.out.println("DA           \t:: "+da);
        System.out.println("Net pay      \t:: "+netpay);
    }
}
```

Output



```
<terminated> _5_Employee [Java Application] C:\Program Files\Java\jd
The Net Pay is 100000.0


Your Salary Details....
Employee No.        ::  420
Employee Name       ::  Abhinay
Basic Salary        ::  80000.0
HRA                 ::  15000.0
DA                  ::  5000.0
Net pay             ::  100000.0
```

Experiment No. 6

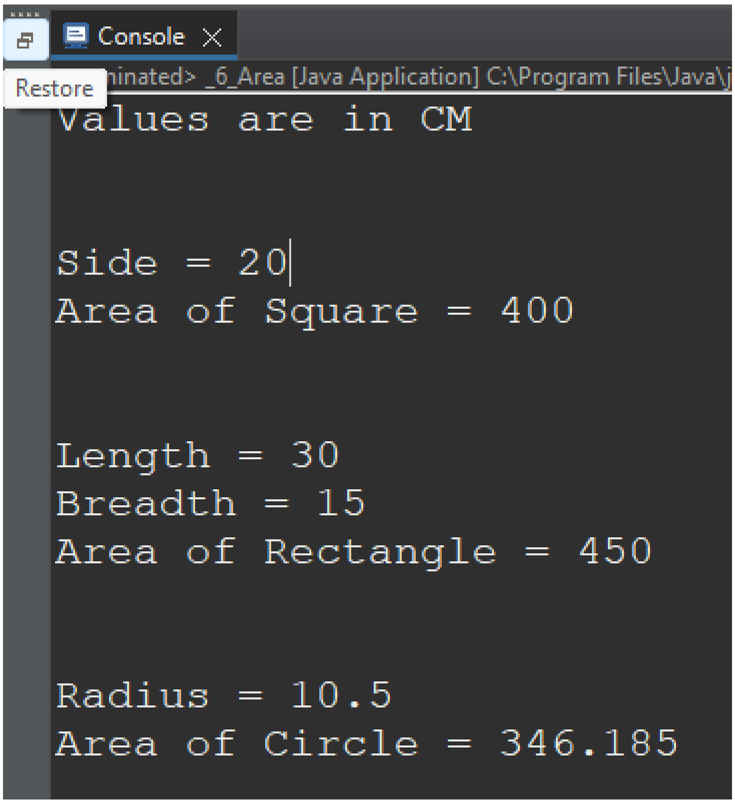
Aim

Program to find area of Square, Rectangle and Circle using Method Overloading.

Source code

```
package java_file;

public class _6_Area {

    float pi=3.14f;

    public static void main(String args[]) {
        _6_Area myobj=new _6_Area();
        System.out.println("Values are in CM");
        System.out.println("\n\nSide = "+20);
        myobj.findArea(20);
        System.out.println("\n\nLength = 30\nBreadth = 15");
        myobj.findArea(30, 15);
        System.out.println("\n\nRadius = "+10.5f);
        myobj.findArea(10.5f);
    }

    public void findArea(int a) {
        System.out.println("Area of Square = "+(a*a));
    }

    public void findArea(int len, int wid) {
        System.out.println("Area of Rectangle = "+(len*wid));
    }

    public void findArea(float rad) {
        System.out.println("Area of Circle = "+(pi*rad*rad));
    }
}
```

Output



```
Values are in CM


Side = 20
Area of Square = 400


Length = 30
Breadth = 15
Area of Rectangle = 450


Radius = 10.5
Area of Circle = 346.185
```

Experiment No. 7

Aim

Construct a class named product with data members PNumber, PName, PPrice and PQuantity, TotalPrice. Write three functions:

product(): to initialize the value of the data members PNumber, PName, PPrice and PQuantity (using constructor concept).

calculate Total(): to calculate the TotalPrice (Total = price* quantity).

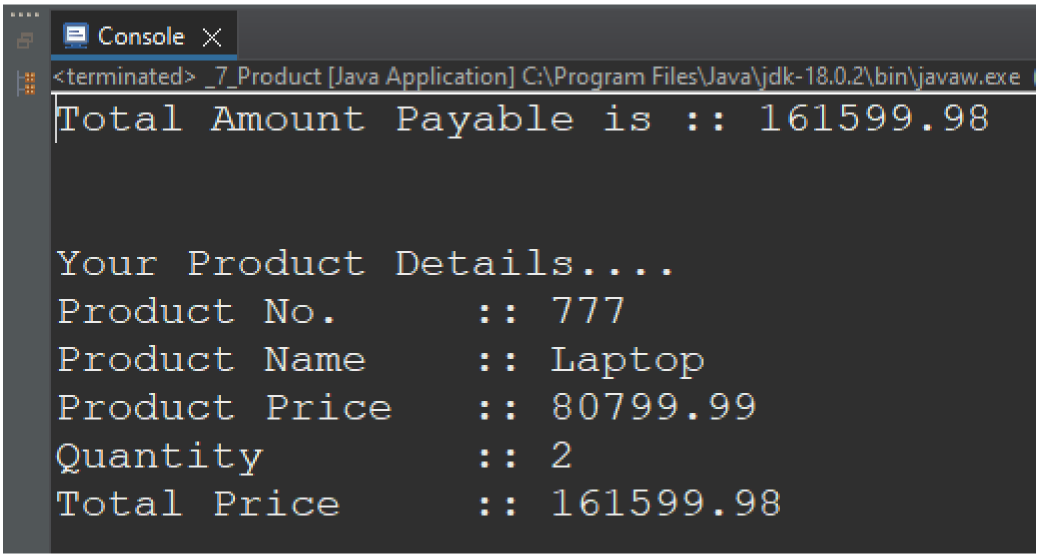
display(): to print the values of the data members on the screen.

Source code

```
package java_file;


public class _7_Product {


    int PNumber;

    String PName;

    float PPrice;

    int PQuantity;

    float TotalPrice;


    public static void main(String[] args) {

            _7_Product obj=new _7_Product(777,"Laptop",80799.99f,2);

            obj.calculateTotal();

            obj.display();

    }


    _7_Product(int a, String b, float c, int d){

            PNumber=a;

            PName=b;
```

```
			PPrice=c;
			PQuantity=d;
		}

		void calculateTotal(){
			TotalPrice=PPrice*PQuantity;
			System.out.println("Total Amount Payable is :: "+TotalPrice);
		}

		void display() {
			System.out.println("\n\nYour Product Details....");
			System.out.println("Product No.  \t:: "+PNumber);
			System.out.println("Product Name \t:: "+PName);
			System.out.println("Product Price \t:: "+PPrice);
			System.out.println("Quantity \t:: "+PQuantity);
			System.out.println("Total Price \t:: "+TotalPrice);
		}
}
```

Output



```
<terminated> _7_Product [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe
Total Amount Payable is :: 161599.98


Your Product Details....
Product No.     :: 777
Product Name    :: Laptop
Product Price   :: 80799.99
Quantity        :: 2
Total Price     :: 161599.98
```

Experiment No. 8 (a)

Aim

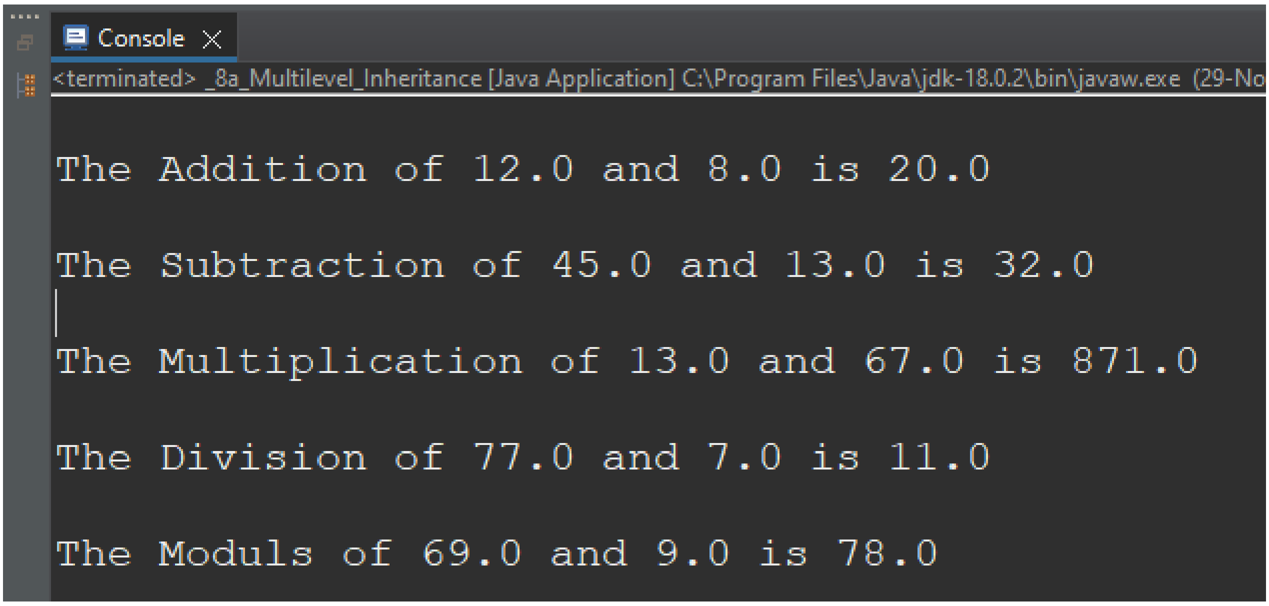
Demonstrate the Concept of Multilevel Inheritance in Java.

Source code

```
package java_file;


class Calculation {


    void add(float x, float y) {

            System.out.println("\nThe Addition of "+x+" and "+y+" is "+(x+y));

    }


    void sub(float x, float y) {

            System.out.println("\nThe Subtraction of "+x+" and "+y+" is "+(x-y));

    }

}


class NewCalculation1 extends Calculation{


    void mul(float x, float y) {

            System.out.println("\nThe Multiplication of "+x+" and "+y+" is "+(x*y));

    }


    void div(float x, float y) {

            System.out.println("\nThe Division of "+x+" and "+y+" is "+(x/y));

    }

}
```

```
class NewCalculation2 extends NewCalculation1{

    void mod(float x, float y) {
        System.out.println("\nThe Moduls of "+x+" and "+y+" is "+(x+y));
    }
}

public class _8a_Multilevel_Inheritance {

    public static void main(String[] args) {
        NewCalculation2 obj=new NewCalculation2();
        obj.add(12, 8);
        obj.sub(45, 13);
        obj.mul(13, 67);
        obj.div(77, 7);
        obj.mod(69, 9);
    }
}
```

Output



<terminated> _8a_Multilevel_Inheritance [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (29-No

```
The Addition of 12.0 and 8.0 is 20.0

The Subtraction of 45.0 and 13.0 is 32.0

The Multiplication of 13.0 and 67.0 is 871.0

The Division of 77.0 and 7.0 is 11.0

The Moduls of 69.0 and 9.0 is 78.0
```

Experiment No. 8 (b)

Aim

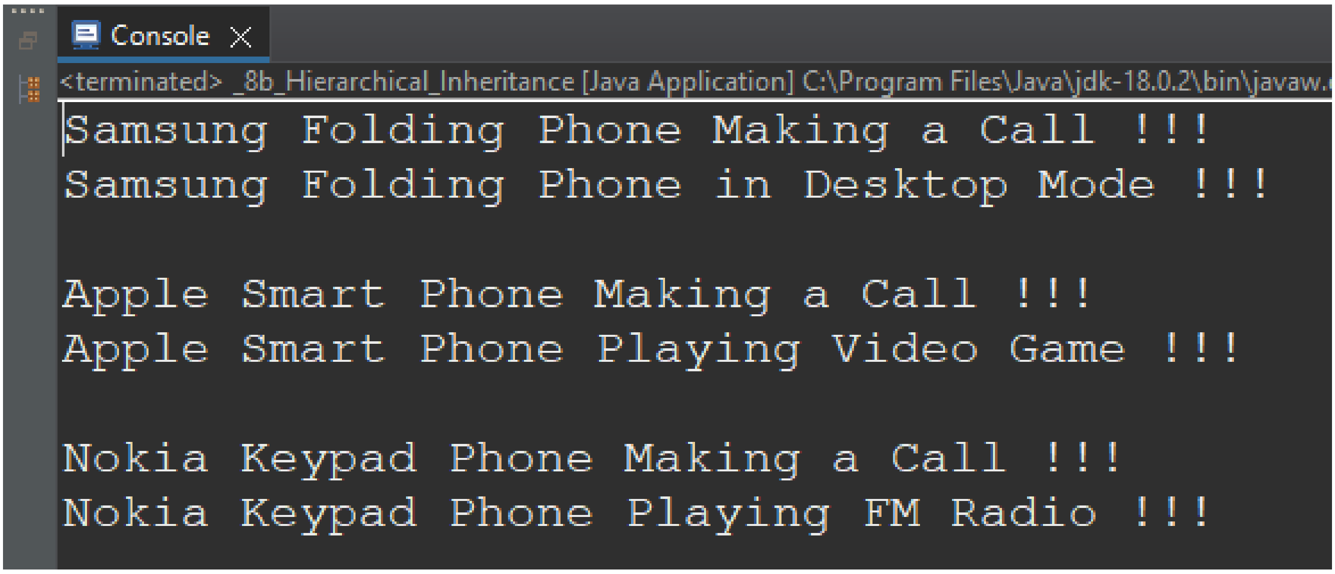
Demonstrate the Concept of Hierarchical Inheritance in java.

Source code

```
package java_file;


class Mobile_Phone {

    int model_no;

    String brand;

    String colour;

    float price;

    int battery;

    String type;




    void call() {

            System.out.println(brand+" "+type+" Making a Call !!!");

    }


    void message() {

            System.out.println(brand+" "+type+" Sending Message !!!");

    }


    void camera() {

            System.out.println(brand+" "+type+" Clicked Photo !!!");

    }
    void speaker() {
            System.out.println(brand+" "+type+" Playing Sound !!!");
    }

    void radio() {
            System.out.println(brand+" "+type+" Playing FM Radio !!!");
    }
}

class KeypadPhone extends Mobile_Phone {
    float keypad_size;

    KeypadPhone(){
            type="Keypad Phone";
            brand="Nokia";
    }
}

class SmartPhone extends Mobile_Phone {
    float display_size;
    String Stylus;

    SmartPhone() {
            type="Smart Phone";
            brand="Apple";
    }

    void finger_print_scanner() {
            System.out.println(brand+" "+type+" Scanning Finger !!!");
    }
    void face_recognition() {
        System.out.println(brand+" "+type+" Scanning Face !!!");
    }

    void games () {
        System.out.println(brand+" "+type+" Playing Video Game !!!");
    }
}

class FoldingPhone extends SmartPhone {

    FoldingPhone() {
        type="Folding Phone";
        brand="Samsung";
    }

    void desktop_mode() {
        System.out.println(brand+" "+type+" in Desktop Mode !!!");
    }

    void Multiwindow() {
        System.out.println(brand+" "+type+" using Multi-Window Function !!!");
    }
}

public class _8b_Hierarchical_Inheritance {

    public static void main(String[] args) {
        FoldingPhone samsung=new FoldingPhone();
```

```
                SmartPhone apple=new SmartPhone();
                KeypadPhone nokia=new KeypadPhone();


                samsung.call();
                samsung.desktop_mode();
                System.out.println();
                apple.call();
                apple.games();
                System.out.println();
                nokia.call();
                nokia.radio();
        }
}
```

Output



```
<terminated> _8b_Hierarchical_Inheritance [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.
Samsung Folding Phone Making a Call !!!
Samsung Folding Phone in Desktop Mode !!!

Apple Smart Phone Making a Call !!!
Apple Smart Phone Playing Video Game !!!

Nokia Keypad Phone Making a Call !!!
Nokia Keypad Phone Playing FM Radio !!!
```

Experiment No. 9

Aim

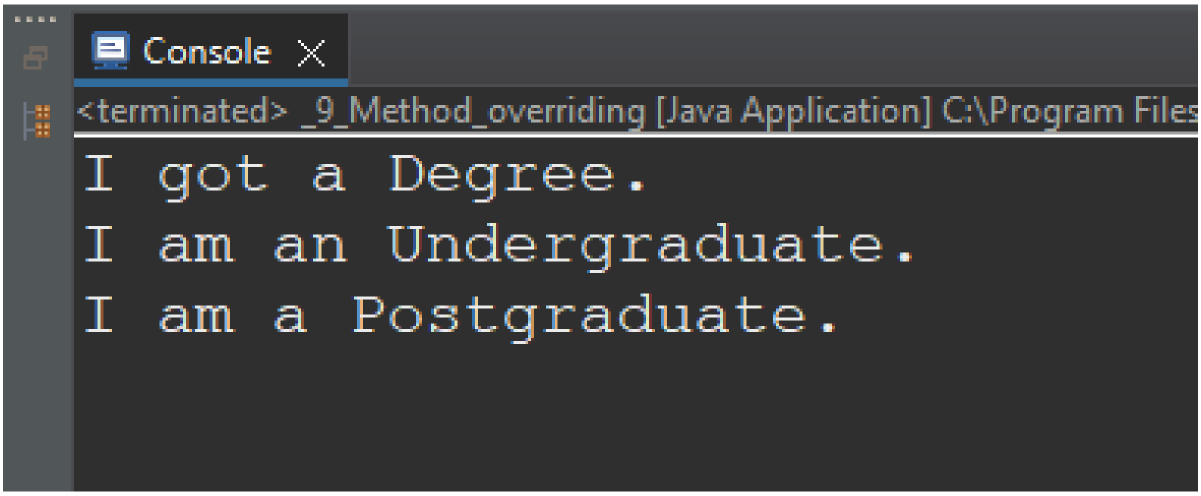
Create a class 'Degree' having a method 'getDegree' that prints "I got a degree". It has two subclasses namely 'Undergraduate' and 'Postgraduate each having a method with the same name that prints "I am an Undergraduate" and "I am a Postgraduate" respectively. Call the 3 method by creating an object of the two child classes.

Source code

```
package java_file;


class Degree {

    void getDegree() {

            System.out.println("I got a Degree.");

    }

}


class Undergraduate extends Degree {

    @Override

    void getDegree() {

            super.getDegree();

            System.out.println("I am an Undergraduate.");

    }

}


class Postgraduate extends Degree {

    void getDegree() {

            System.out.println("I am a Postgraduate.");

    }

}
```

```
public class _9_Method_overriding {

    public static void main(String[] args) {
            Undergraduate student=new Undergraduate();
            Postgraduate student2=new Postgraduate();
            student.getDegree();
            student2.getDegree();
    }
}
```

Output



```
<terminated> _9_Method_overriding [Java Application] C:\Program Files
I got a Degree.
I am an Undergraduate.
I am a Postgraduate.
```

Experiment No. 10

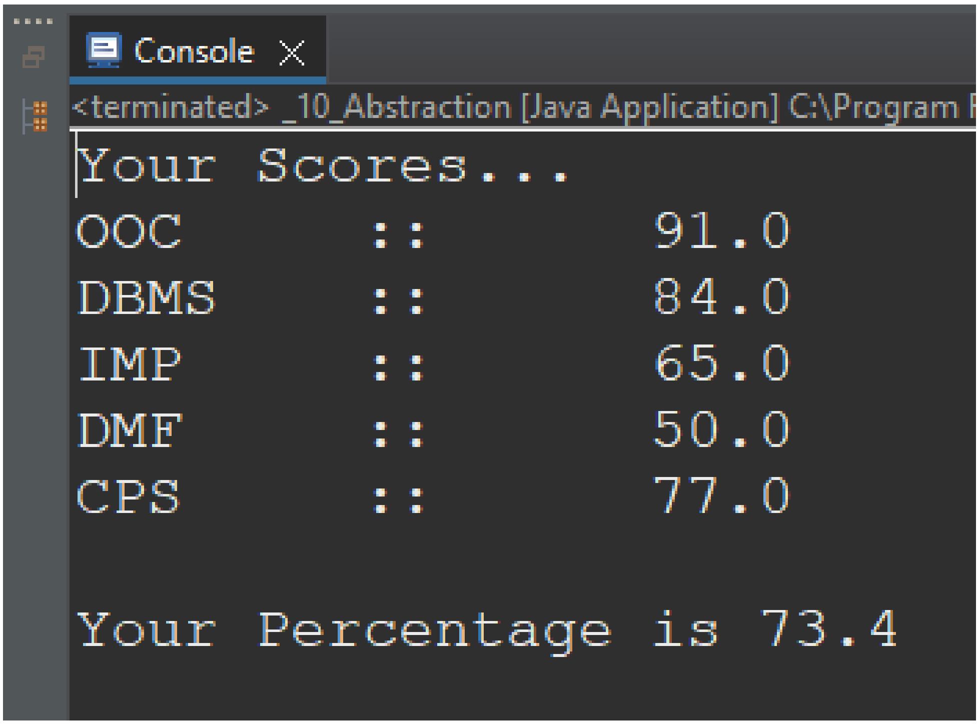
Aim

Demonstrate the Concept of Abstraction in Java.

Source code

```
package java_file;

abstract class Result {
    abstract float percentage(float x[]);
    abstract float sum(float x[]);

    void display(float a[]) {
        System.out.println("Your Scores...");
        System.out.println("OOC \t::\t"+a[0]);
        System.out.println("DBMS \t::\t"+a[1]);
        System.out.println("IMP \t::\t"+a[2]);
        System.out.println("DMF \t::\t"+a[3]);
        System.out.println("CPS \t::\t"+a[4]);
        System.out.println("\nYour Percentage is "+percentage(a));
    }
}

class Working extends Result {
    float percentage(float x[]) {
        float per=(sum(x)/500)*100;
        return per;
    }
    float sum(float x[]) {
        float temp=0;
        for(int i=0;i<x.length;i++)
            temp+=x[i];
        return temp;
    }
}

public class _10_Abstraction {

    public static void main(String[] args) {
        float arr[]= {91,84,65,50,77};
        Working obj=new Working();
        obj.display(arr);
    }

}
```

Output



```
<terminated> _10_Abstraction [Java Application] C:\Program F
Your Scores...
OOC       ::       91.0
DBMS      ::       84.0
IMP       ::       65.0
DMF       ::       50.0
CPS       ::       77.0

Your Percentage is 73.4
```

Experiment No. 11

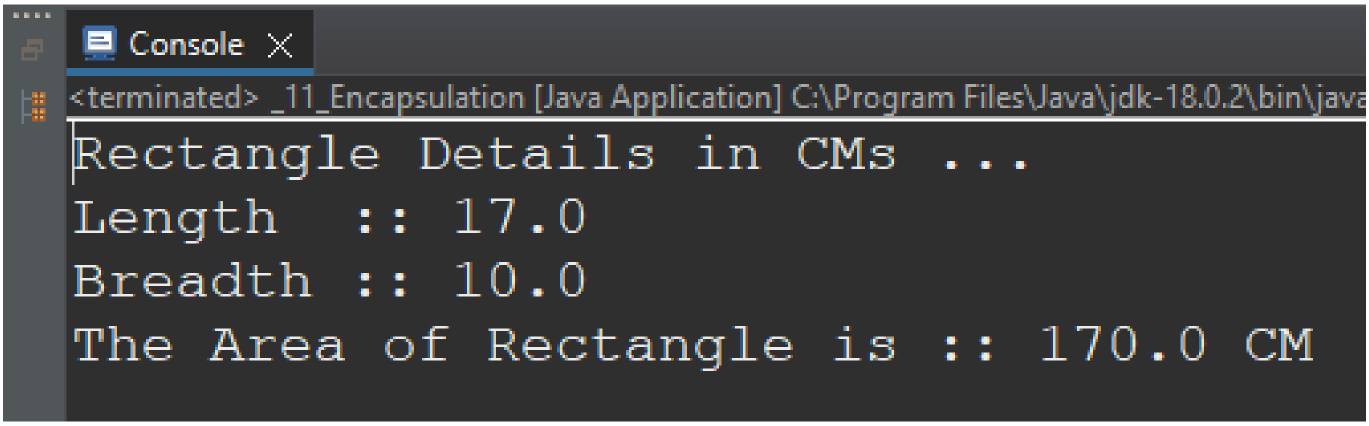
Aim

Java program to calculate the area of rectangle using the concept of Encapsulation.

Source code

```
package java_file;


class Area_Rectangle {

    private float length;

    private float breadth;

    private float Area;


    private void Cal_Area() {

        Area=length*breadth;

        System.out.println("The Area of Rectangle is :: "+Area+" CM");

    }


    public void setLength_Breadth(float length, float breadth) {

        this.length = length;

        this.breadth = breadth;

    }


    public float getLength() {

        return length;

    }


    public float getBreadth() {

        return breadth;

    }
    public void show_Area() {
            Cal_Area();
    }
}

public class _11_Encapsulation {

    public static void main(String[] args) {
            Area_Rectangle obj=new Area_Rectangle();
            obj.setLength_Breadth(17, 10);
            System.out.println("Rectangle Details in CMs ...");
            System.out.println("Length  :: "+obj.getLength());
            System.out.println("Breadth :: "+obj.getBreadth());
            obj.show_Area();
    }
}
```

Output



```
<terminated> _11_Encapsulation [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\java
Rectangle Details in CMs ...
Length  :: 17.0
Breadth :: 10.0
The Area of Rectangle is :: 170.0 CM
```

Experiment No. 12

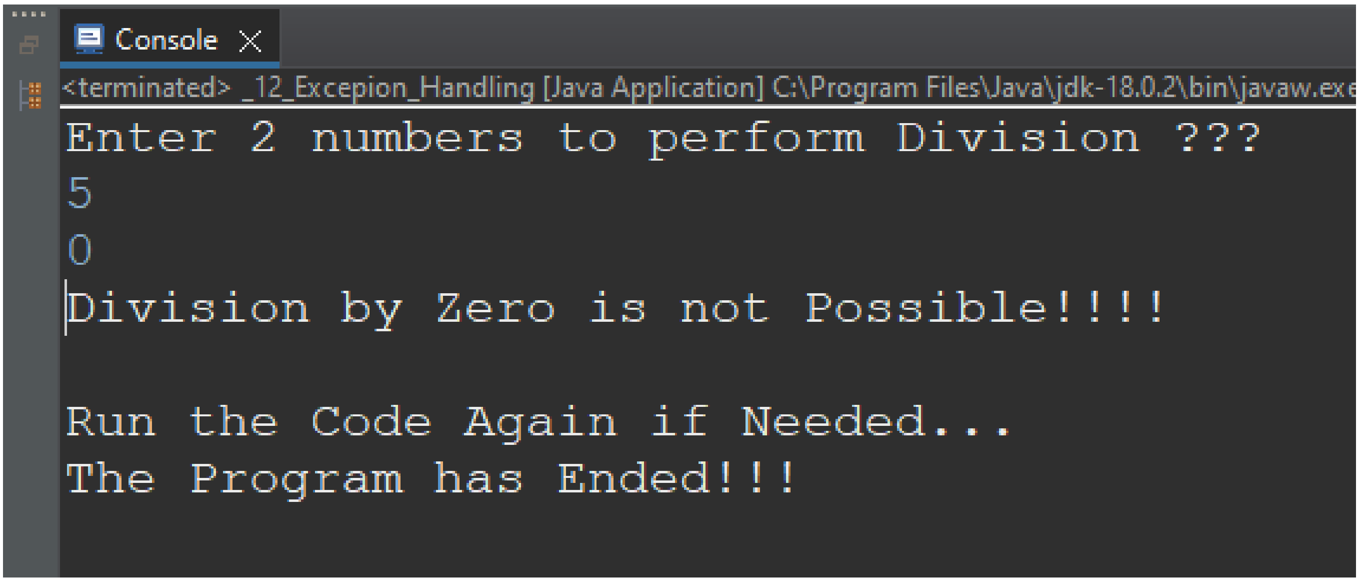
Aim

Java Program to demonstrate the concept of try, catch, and finally block in Exception Handling.

Source code

```
package java_file;

import java.util.Scanner;


public class _12_Excepion_Handling {


    public static void main (String[] args) {

        Scanner input=new Scanner (System.in);

        System.out.println("Enter 2 numbers to perform Division ???");

        int a=input.nextInt();

        int b=input.nextInt();

        try

        {

            int c=a/b;

            System.out.println("The Quotient of division is :: "+c);

        }

        catch (ArithmeticException e)

        {

            System.out.println("Division by Zero is not Possible!!!!");

        }

        finally

        {

            System.out.println("\nRun the Code Again if Needed...");

            System.out.println("The Program has Ended!!!");
            }
        }
    }
```

Output



Console

<terminated> _12_Excepion_Handling [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe

```
Enter 2 numbers to perform Division ???
5
0
Division by Zero is not Possible!!!!

Run the Code Again if Needed...
The Program has Ended!!!
```