2. Write a program in java to implement a calculator with simple arithmetic operations

```
package constructoroverloading;

import java.util.Scanner;

public class Calculator {
        public static void main(String[] args) {
                Double a, b, result;
                Scanner input = new Scanner(System.in);

               System.out.println("Enter an operator: +, -, *, or /");
               String operator = input.next();

                System.out.println("Enter first number");
                a = input.nextDouble();

                System.out.println("Enter second number");
                b = input.nextDouble();

                switch (operator) {

                   case "+":
                       result = a + b;
```

```
                System.out.println(a + " + " + b + " = " + result);
                break;

            case "-":
                result = a - b;
                System.out.println(a + " - " + b + " = " + result);
                break;

            case "*":
                result = a * b;
                System.out.println(a + " * " + b + " = " + result);
                break;
            case "/":
                result = a / b;
                System.out.println(a + " / " + b + " = " + result);
                break;

            default:
                System.out.println("Invalid operator!");
                break;
        }

    }

}
```

3. Write a program in Java with class Rectangle with the data fields width, length, area and colour. The length, width and area are of double type and colour is of string type. The methods are get_length (), get_width (), get_colour() and find_area(). Create two objects of Rectangle and compare their area and colour. If the area and colour both are the same for the objects then display “Matching Rectangles”, otherwise display “Non-matching Rectangle”

```
package com.descending;


import java.util.Scanner;


  public class Rectangle {
Double length=0.0;
Double width=0.0;
String colour="";
  Double area=0.0;
  Scanner scanner=new Scanner(System.in);
  public Double getLength() {
  System.out.println("Enter the Length");
  length=scanner.nextDouble();
  return length;
  }
  public Double getWidth() {
  System.out.println("Enter the width");
  width=scanner.nextDouble();
  return width;
  }
public String getColour() {
```

```
  System.out.println("Enter the colour");
  colour=scanner.next();
  return colour;
  }
  public Double findArea() {
  area=length*width;
System.out.println("Area is"+area);
  return area;
  }
  public static void main(String[] args) {
  System.out.println("First Rectangle");
  Rectangle rect1=new Rectangle();
  rect1.getLength();
  rect1.getWidth();
  rect1.getColour();
  rect1.findArea();
  System.out.println("Second Rectangle");
Rectangle rect2=new Rectangle();
rect2.getLength();
  rect2.getWidth();
  rect2.getColour();
  rect2.findArea();
  if(rect1.area==rect2.area || rect1.colour.equals(rect2.colour))
System.out.println("Matching");
  else
```

```
        System.out.println("Not matching");


}

}
```

5.Write a program in java to sort the names in alphabetical order.

```
import java.util.Scanner;
public class GFC
{
    public static void main(String[] args)
    {
        int count;
        String temp;
        Scanner scan = new Scanner(System.in);

        //User will be asked to enter the count of strings
        System.out.print("Enter number of strings you would like to enter:");
        count = scan.nextInt();
        String str[] = new String[count];
        Scanner scan2 = new Scanner(System.in);
       //User is entering the strings and they are stored in an array
        System.out.println("Enter the Strings one by one:");
        for(int i = 0; i < count; i++)
        {
            str[i] = scan2.nextLine();
        }
        scan.close();
        scan2.close();
       //Sorting the strings
        for (int i = 0; i < count; i++)
        {
            for (int j = i + 1; j < count; j++) {
                if (str[i].compareTo(str[j])>0)
```

```
                {
                    temp = str[i];
                    str[i] = str[j];
                    str[j] = temp;
                }
            }
        }
//Displaying the strings after sorting them based on alphabetical order
        System.out.print("Strings in Sorted Order:");
        for (int i = 0; i <= count - 1; i++)
        {
            System.out.print(str[i] + ", ");
        }
    }
}
```

8. Write a java program which implements INTERFACE

1. Teacher

```
package java_interface;


public interface Teacher {
	void assignment();
	void practical();
	int marks();


}
```

2. Student

```
package java_interface;
import java.util.Scanner;
public class Student implements Teacher {
	public void assignment() {
		System.out.println("I am doing Assignment");


	}
	public void practical() {
		System.out.println("I am doing Practical ");


	}
	public int marks() {
		return 9;
	}


}
```

3. Test

```
package java_interface;


public class Test {

	public static void main(String[] args) {

		Student s1= new Student();

		s1.assignment();

		s1.practical();

		System.out.println("Marks obtained by Student is: " + s1.marks());

	}


}
```

9. Write a java program for exception handling using try catch and finally block?

```
public class Main {
    public static void main(String[] args) {
        try {
            int[] myNumbers = {1, 2, 3};
            System.out.println(myNumbers[10]);
        } catch (Exception e) {
            System.out.println("Something went wrong.");
        } finally {
            System.out.println("The 'try catch' is finished.");
        }
    }
}
```

10.Write a java program to draw oval,rectangle,line textusing graphics class.

```
package com.test;

import java.awt.*;
import java.applet.*;
public class Demo extends Applet
{
  public void paint(Graphics g)
{

//Draw a Oval
      g.drawOval(20,20,200,120);
      g.setColor(Color.green);
      g.fillOval(70,30,100,100);
      //Draw a rectangle
      g.setColor(Color.black);
      g.drawRect(120, 50, 100, 100);
  //Draw a line
      g.drawLine(100,10,250, 150);
      g.drawLine(100,150,150,10);

      g.drawString("Hello World!", 200, 200);
      showStatus("showing the Status Message");
            //Draw a text
```

```
        g.drawString("Hello World!", 400, 400);
        showStatus("showing the Status Message");


}
}
```

11. Write a java program in which data is read from one file and should be written in another line.

```
package com.test;

import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class Applet {
       public static void main(String[] args)
          {


              try {

                  FileReader fr = new
FileReader("C:\\Users\\Sudershan\\Desktop\\abc.txt");

                  FileWriter fw = new
FileWriter("C:\\Users\\Sudershan\\Desktop\\abcoutput.txt");


                  String str = "";

                  int i;

                  while ((i = fr.read()) != -1) {

                                      str += (char)i;
                  }

                  System.out.println(str);

                  fw.write(str);

                  fr.close();
                  fw.close();

                  System.out.println(
                      "File reading and writing both done");
              }

              catch (IOException e) {


                  System.out.println(
                      "There are some IOException");
              }
          }

}
```

12. Write a Java Program to sort the elements of an array in ascending order

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

		//Initialize array
		int [] arr = new int [] {5, 2, 8, 7, 1};
		int temp = 0;

		//Displaying elements of original array
		System.out.println("Elements of original array: ");
		for (int i = 0; i < arr.length; i++) {
			System.out.print(arr[i] + " ");
		}

		//Sort the array in ascending order
		for (int i = 0; i < arr.length; i++) {
			for (int j = i+1; j < arr.length; j++) {
				if(arr[i] > arr[j]) {
					temp = arr[i];
					arr[i] = arr[j];
					arr[j] = temp;
				}
			}
		}

		System.out.println();
```

```
        //Displaying elements of array after sorting
        System.out.println("Elements of array sorted in ascending order: ");
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
    }
}
```

Output:

Elements of original array:

5 2 8 7 1

Elements of array sorted in ascending order:

1 2 5 7 8

13 To calculate and display area of circle

```
public class Circle {
	public static void main(String[] args)
	{
		int radius;
		double pi = 3.142, area;
		radius = 5;
		// calculating the area of the circle
		area = pi * radius * radius;
		// printing the area of the circle
		System.out.println("Area of circle is :" + area);
	}
}
```

14	Write a java program for “Array out of bound” exception handling using try catch and finally block.

```
class Main {
    public static void main(String[] args) {
        //array of subjects. There are 5 elements.
        String[] subjects = {"Maths","Science","French","Sanskrit", "English"};

        //for loop iterates from 0 to 5 (length of array)
        for(int i=0;i<=subjects.length;i++) {
            //when ‘i’ reaches 5, it becomes invalid index and exception will be thrown
            System.out.print(subjects[i] + " ");
        }
    }
```

15. Write a java program for method overloading and overriding?

Overloading:-

```
class Adder{
static int add(int a, int b){return a+b;}
static double add(double a, double b){return a+b;}
}
class TestOverloading2{
public static void main(String[] args){
System.out.println(Adder.add(11,11));
System.out.println(Adder.add(12.3,12.6));
}}
```

overriding:-

```
class Animal {
    public void displayInfo() {
        System.out.println("I am an animal.");
    }
}

class Dog extends Animal {
    @Override
    public void displayInfo() {
        System.out.println("I am a dog.");
    }
}
```

```
class Main {
    public static void main(String[] args) {
        Dog d1 = new Dog();
        d1.displayInfo();
    }
}
```

1. To find factorial of a number

```
package java_factorial;


class Factorial {
	public static void main(String args[]) {
		int i;
		int fact=1;
		int number=5;
		for(i=1;i<=number;i++) {
			fact=fact*i;
		}
		System.out.println("Factorial of    "+number+" is: "+fact);
	}

}
```

17. Write a Java program to display first 50 prime numbers.

```
package com.demo;


  import java.util.Scanner;
class PrimeNumberDemo
{
      public static void main(String args[])
      {
            int n;
            int status = 1;
            int num = 3;
            //For capturing the value of n
            Scanner scanner = new Scanner(System.in);
            System.out.println("Enter the value of n:");
            //The entered value is stored in the var n
            n = scanner.nextInt();
            if (n >= 1)
            {
                  System.out.println("First "+n+" prime numbers are:");
                  //2 is a known prime number
                  System.out.println(2);
            }


            for ( int i = 2 ; i <=n ;    )
```

```
        {
            for ( int j = 2 ; j <= Math.sqrt(num) ; j++ )
            {
                if ( num%j == 0 )
                {
                    status = 0;
                    break;
                }
            }
            if ( status != 0 )
            {
                System.out.println(num);
                i++;
            }
            status = 1;
            num++;
        }
    }
}
```

18. write a program in JAVA to demonstrate the constructor overloading

```
package constructoroverloading;


public class ConstructorOverloading{
	int id;
	String name;
	public ConstructorOverloading() {
		System.out.println("Default constructor");


	}
	public ConstructorOverloading(int a, String b) {
		id=a;
		name=b;


	}
	public static void main(String[]args) {
		ConstructorOverloading con=new ConstructorOverloading();
		System.out.println("Student Id: "+con.id + "\nStudent Name : "+con.name);
		ConstructorOverloading con2=new ConstructorOverloading(10, "Ram");
		System.out.println("Student Id: "+con2.id + "\nStudent Name : "+con2.name);



				}
}
```

19. Write a java program to sort the element of an array in descending array.

```
public class SortDsc {
	public static void main(String[] args) {
		//Initialize array
		int [] arr = new int [] {5, 2, 8, 7, 1};
		int temp = 0;

		//Displaying elements of original array
		System.out.println("Elements of original array: ");
		for (int i = 0; i < arr.length; i++) {
			System.out.print(arr[i] + " ");
		}

		//Sort the array in descending order
		for (int i = 0; i < arr.length; i++) {
			for (int j = i+1; j < arr.length; j++) {
				if(arr[i] < arr[j]) {
					temp = arr[i];
					arr[i] = arr[j];
					arr[j] = temp;
				}
			}
		}

		System.out.println();
```

```
        //Displaying elements of array after sorting
        System.out.println("Elements of array sorted in descending order: ");
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
    }
}
```