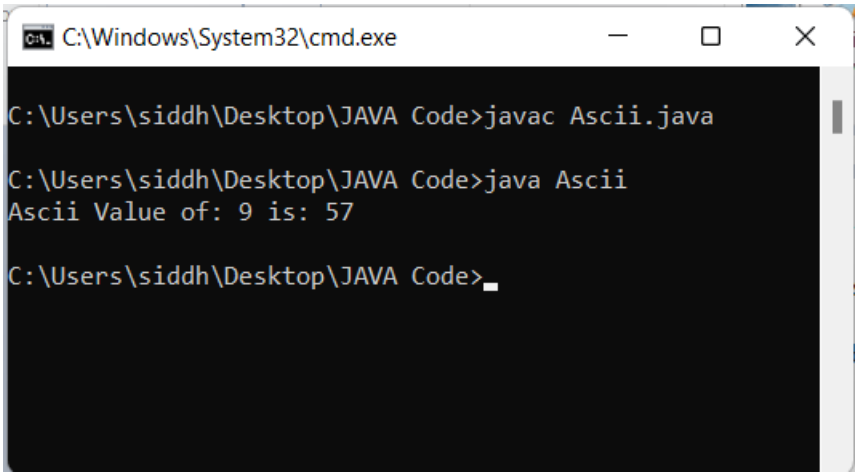
Q1. Write a program lo display ASCII value of a number 9.

Ans: Code:

```
public class Ascii{
        public static void main(String...args){
                char ch='9';
                int ascii=ch;
                System.out.println("Ascii Value of: " +ch+ " is: " +ascii);


        }
}
```

Output:



Q2. Write a program which displays functioning of ATM machine, (Hint: Withdraw, Deposit, Check Balance and Exit).

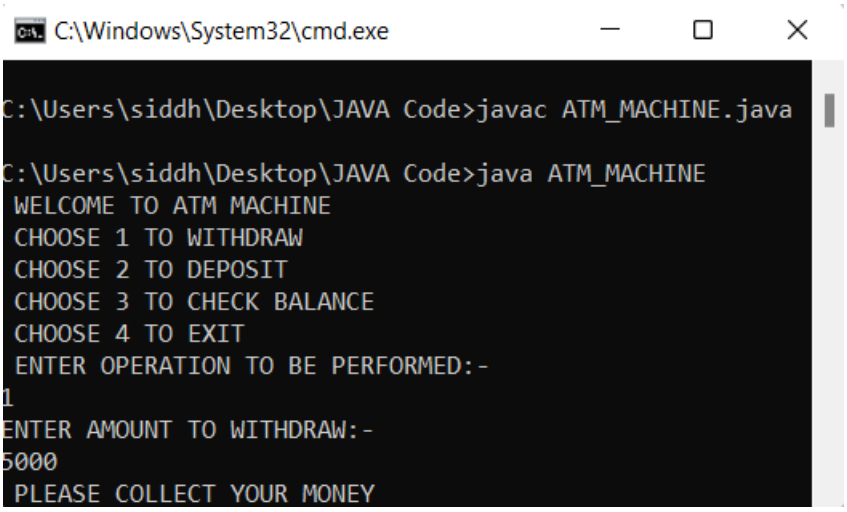
Ans: Code:

```
import java.util.*;
public class ATM_MACHINE{
        public static void main(String...args){
                int balance=10000,withdraw,deposit;
```

```
Scanner sc=new Scanner(System.in);
while(true){
	System.out.println(" WELCOME TO ATM MACHINE ");
	System.out.println(" CHOOSE 1 TO WITHDRAW ");
	System.out.println(" CHOOSE 2 TO DEPOSIT ");
	System.out.println(" CHOOSE 3 TO CHECK BALANCE ");
	System.out.println(" CHOOSE 4 TO EXIT ");
	System.out.println(" ENTER OPERATION TO BE PERFORMED:- ");
	int n=sc.nextInt();
	switch(n){
		case 1:
		System.out.println("ENTER AMOUNT TO WITHDRAW:- ");
		withdraw=sc.nextInt();
		if(balance>withdraw){
			balance=balance-withdraw;
			System.out.println(" PLEASE COLLECT YOUR MONEY ");
		}else{
			System.out.println(" INSUFFICIENT BALANCE ");
		}
		System.out.println("");
		break;
		case 2:
		System.out.println("ENTER AMOUNT TO DEPOSIT:- ");
		deposit=sc.nextInt();
		balance=balance+deposit;
		System.out.println(" AMOUNT DEPOSITED SUCCESSFULLY ");
		System.out.println("");
		break;
		case 3:
```

```
                    System.out.println("YOUR CURRENT BALANCE IS:- "+balance);
                    System.out.println("");
                    break;
                    case 4:
                    System.exit(0);
                }
            }
        }
}
```

Output:



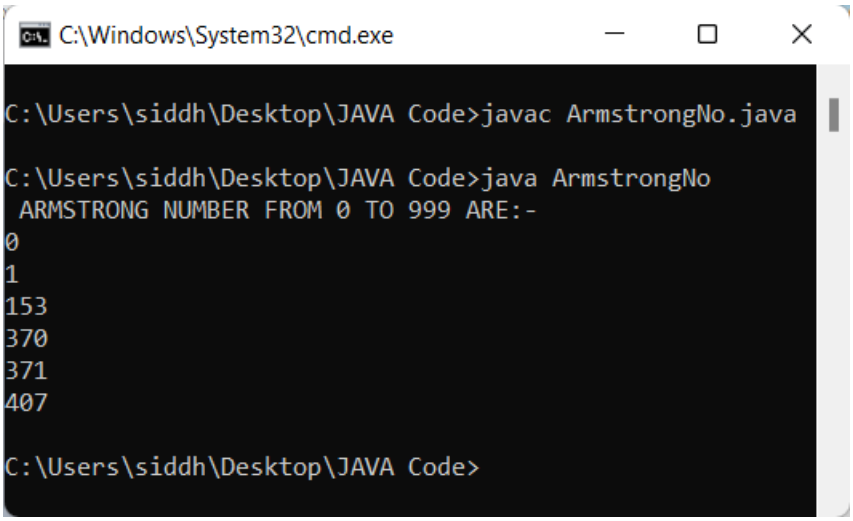
Q3. Write a program to print all the Armstrong numbers from 0 to 999.

Ans: Code:

```
public class ArmstrongNo{
    public static void main(String...args){
        int i=0,arm,a,n;
        System.out.println(" ARMSTRONG NUMBER FROM 0 TO 999 ARE:- ");
        while(i<1000){
            n=i;
            arm=0;
```

```
                while(n>0){
                        a=n%10;
                        arm=arm+(a*a*a);
                        n=n/10;
                }
                if(arm==i)
                System.out.println(i);
                i++;
        }
    }
}
```

Output:



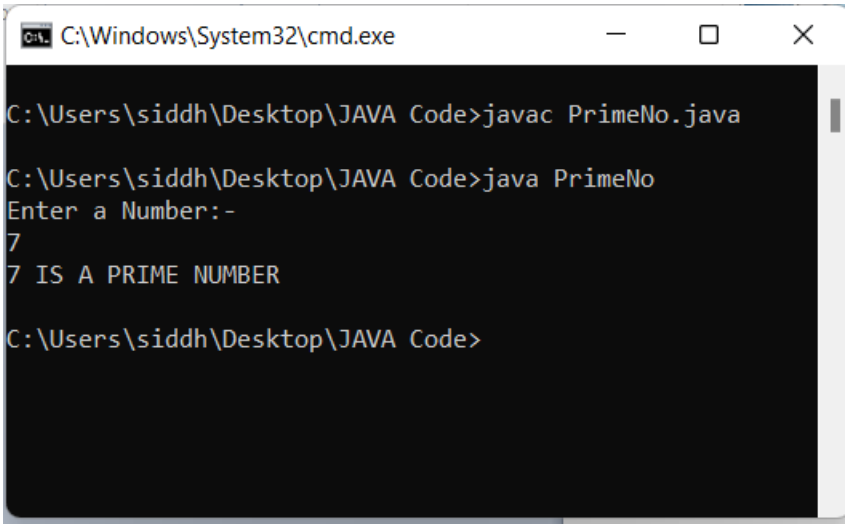
Q4. Write a program to check whether the given number is prime or not.

Ans: Code:

```
import java.util.*;
public class PrimeNo{
        public static void main(String...args){
                int i,flag=0;
```

```
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter a Number:- ");
                int num=sc.nextInt();
                for(i=2;i<num;i++){
                        if(num%i==0){
                                flag=1;
                                break;
                        }
                }
                if(flag==0){
                        System.out.println(num+ " IS A PRIME NUMBER ");
                }else{
                        System.out.println(num+ " IS NOT A PRIME NUMBER ");
                }
        }
}
```

Output:

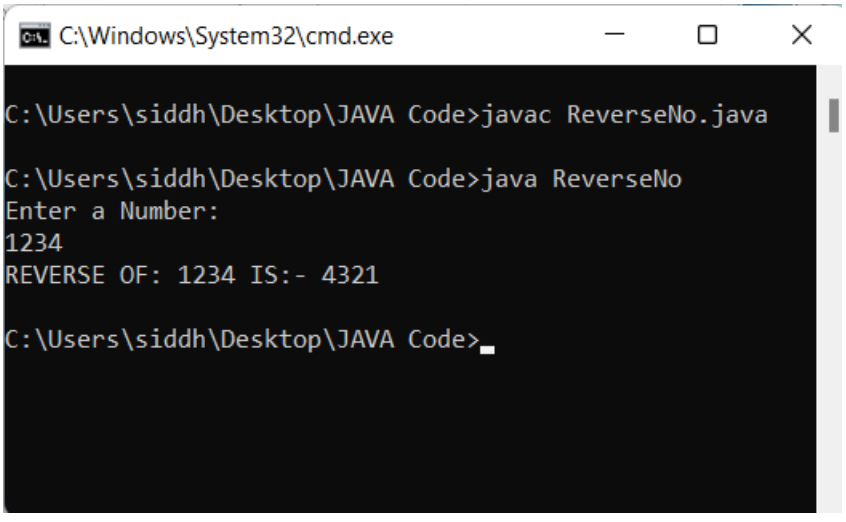


Q5. Write a program to find reverse of a number.

Ans: Code:

```
import java.util.*;
public class ReverseNo{
	public static void main(String...args){
		int rev=0,rem,num,d;
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter a Number: ");
		num=sc.nextInt();
		d=num;
		while(num!=0){
			rem=num%10;
			rev=(rev*10)+rem;
			num=num/10;
		}
		System.out.println("REVERSE OF: " +d+ " IS:- "+rev);
	}
}
```

Output:

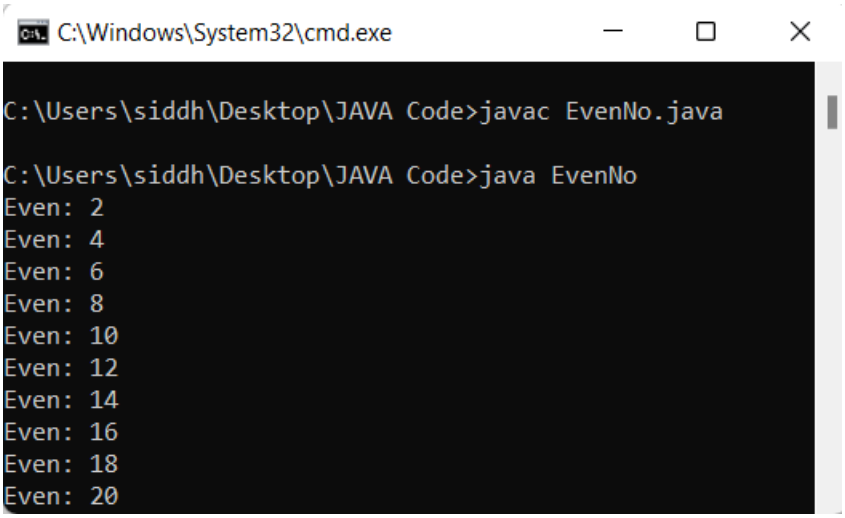


Q6. Write a Java Program to find out the even numbers from 1 to 100 using for loop.

Ans: Code:

```
public class EvenNo{
	public static void main(String...args){
		for(int i=1;i<100;i++){
			if(i%2==0){
				System.out.println("Even: "+i);
			}
		}
	}
}
```

Output:



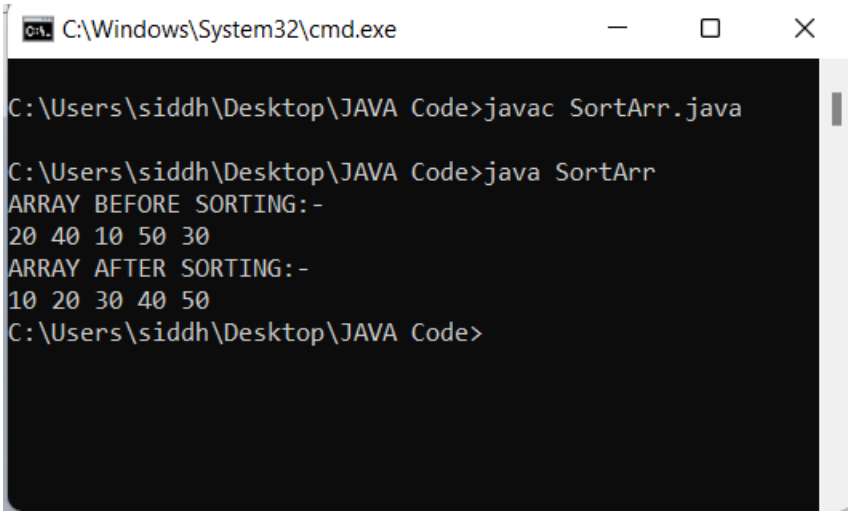
Q7. Write a program to sort the elements of an array in ascending order.

Ans: Code:

```
public class SortArr{
	public static void main(String...args){
		int arr[]={20,40,10,50,30};
		System.out.println("ARRAY BEFORE SORTING:- ");
```

```
                for(int i=0;i<arr.length;i++){
                        System.out.print(arr[i]+ " ");
                }
                System.out.println("");
                int n=arr.length;
                int temp;
                for(int i=0;i<n;i++){
                        for(int j=1;j<n-i;j++){
                                if(arr[j-1]>arr[j]){
                                        temp=arr[j-1];
                                        arr[j-1]=arr[j];
                                        arr[j]=temp;
                                }
                        }
                }
                System.out.println("ARRAY AFTER SORTING:- ");
                for(int i=0;i<arr.length;i++){
                        System.out.print(arr[i]+ " ");
                }
        }
}
```

Output:



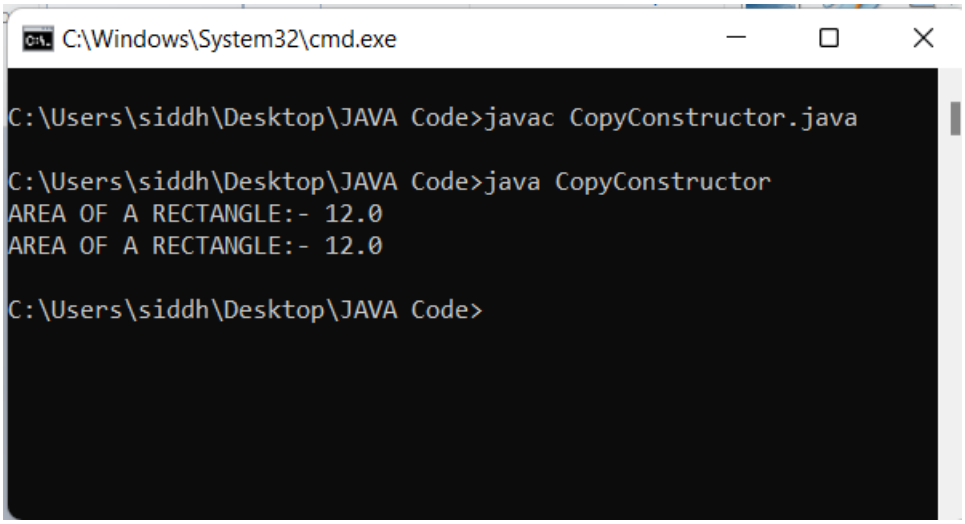
Q8. Write a program to show the use of copy constructor.

Ans: Code:

```
public class CopyConstructor{
        float length,breadth;
        CopyConstructor(float len,float brt){
                length=len;
                breadth=brt;
        }
        CopyConstructor(CopyConstructor cc){
                length=cc.length;
                breadth=cc.breadth;
        }
        void display(){
                float area;
                area=length*breadth;
                System.out.println("AREA OF A RECTANGLE:- "+area);
        }
        public static void main(String...args){
```

```
        CopyConstructor c1=new CopyConstructor(4,3);
        c1.display();
        CopyConstructor c2=new CopyConstructor(c1);
        c2.display();
    }
}
```

Output:



Q9. Write a program to print the sum, difference and product of two complex numbers by creating a class named "Complex" with separate methods for each operation whose real and imaginary parts are entered by user.

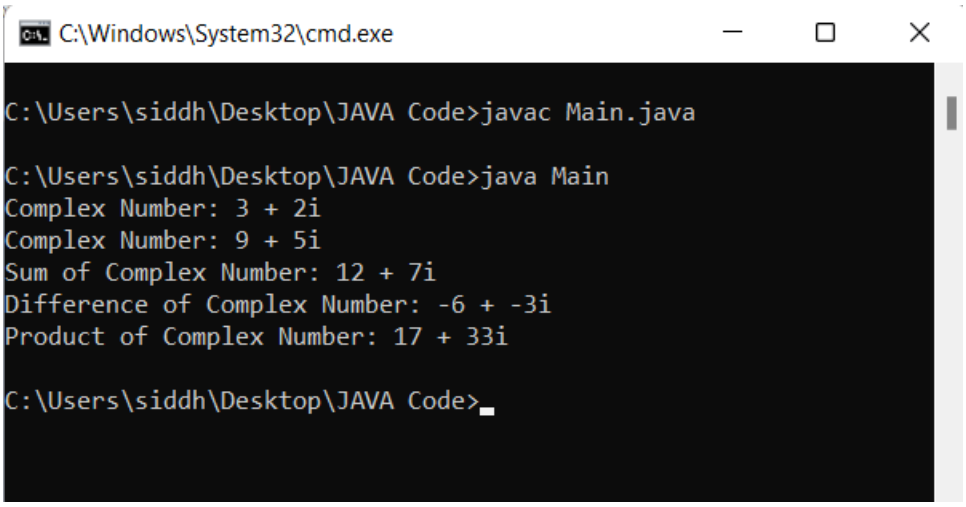
Ans: Code:

```
import java.util.*;
class Complex{
    int real,imaginary;
    Complex(){
    }
    Complex(int r,int i){
        real=r;
        imaginary=i;
```

```
	}
	Complex addComp(Complex c1,Complex c2){
		Complex temp=new Complex();
		temp.real=c1.real+c2.real;
		temp.imaginary=c1.imaginary+c2.imaginary;
		return temp;
	}
	Complex subComp(Complex c1,Complex c2){
		Complex temp=new Complex();
		temp.real=c1.real-c2.real;
		temp.imaginary=c1.imaginary-c2.imaginary;
		return temp;
	}
	Complex prodComp(Complex c1,Complex c2){
		Complex temp=new Complex();
		temp.real=((c1.real*c2.real)-(c1.imaginary*c2.imaginary));
		temp.imaginary=((c1.real*c2.imaginary)+(c1.imaginary*c2.real));
		return temp;
	}
	void printComplexNumber(){
		System.out.println("Complex Number: " +real+ " + " +imaginary+ "i");
	}
}
public class Main{
	public static void main(String...args){
		Complex c1=new Complex(3,2);
		c1.printComplexNumber();
		Complex c2=new Complex(9,5);
```

```
        c2.printComplexNumber();
        Complex c3=new Complex();
        c3=c3.addComp(c1,c2);
        System.out.print("Sum of ");
        c3.printComplexNumber();
        c3=c3.subComp(c1,c2);
        System.out.print("Difference of ");
        c3.printComplexNumber();
        c3=c3.prodComp(c1,c2);
        System.out.print("Product of ");
        c3.printComplexNumber();
    }

}
```

Output:



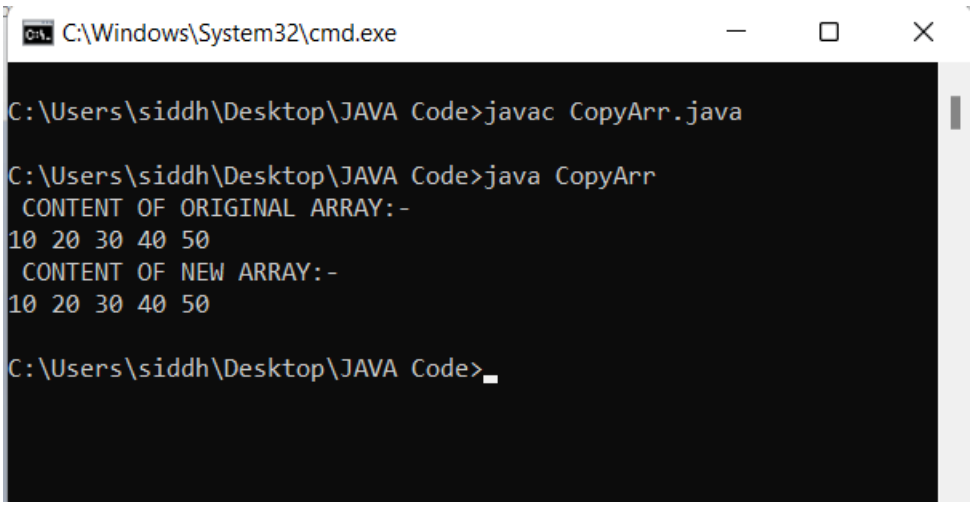
Q10. Write a program to copy all elements of one array into another array.

Ans: Code:

```
public class CopyArr{
```

```
    public static void main(String...args){
        int arr1[]={10,20,30,40,50};
        int arr2[]=new int[arr1.length];
        for(int i=0;i<arr1.length;i++){
            arr2[i]=arr1[i];
        }
        System.out.println(" CONTENT OF ORIGINAL ARRAY:- ");
        for(int i=0;i<arr1.length;i++){
            System.out.print(arr1[i]+ " ");
        }
        System.out.println("");
        System.out.println(" CONTENT OF NEW ARRAY:- ");
        for(int i=0;i<arr2.length;i++){
            System.out.print(arr2[i]+ " ");
        }
        System.out.println("");
    }
}
```

Output:

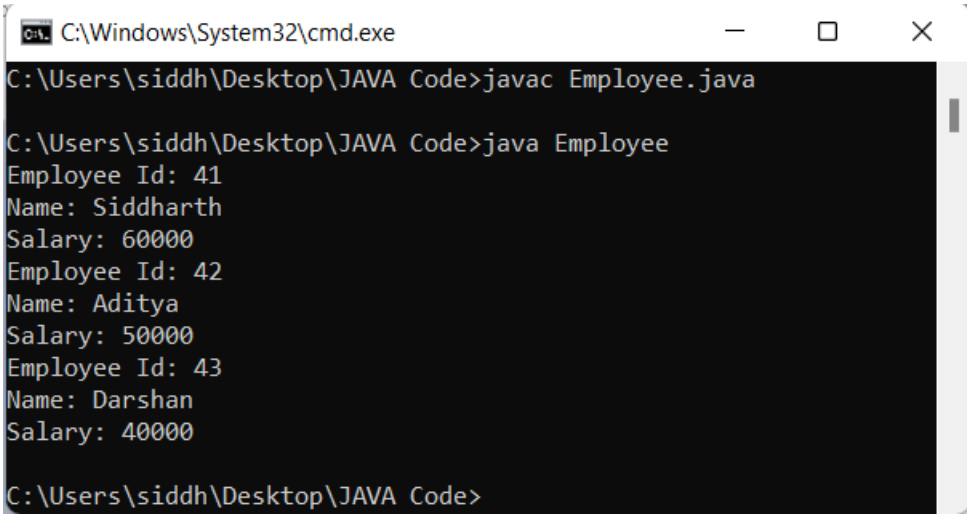


Q11. Define a class employee with data members 'empid, name and salary'. Accept data for three objects and display it.

Ans: Code:

```
public class Employee{
	int empid,salary;
	String name;
	Employee(int empid,String name,int salary){
		this.empid=empid;
		this.name=name;
		this.salary=salary;
	}
	public static void main(String...args){
		Employee e[]=new Employee[3];
		e[0]=new Employee(41,"Siddharth",60000);
		e[1]=new Employee(42,"Aditya",50000);
		e[2]=new Employee(43,"Darshan",40000);
		for(int i=0;i<e.length;i++){
			System.out.println("Employee Id: "+e[i].empid);
			System.out.println("Name: "+e[i].name);
			System.out.println("Salary: "+e[i].salary);
		}
	}
}
```

Output:



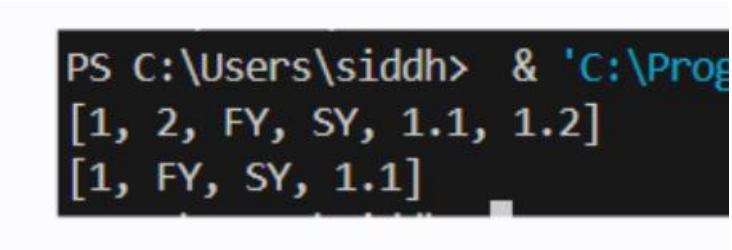
Q12. Write a program to add 2 integers, 2 string and 2 float values in a vector. Remove the element specified by the user and display the list.

Ans: Code:

```
import java.util.*;

public class Vector2 {

   public static void main(String[] args) {

      Vector v=new Vector();

      Integer s1=new Integer(1);

      Integer s2=new Integer(2);

      String s3=new String("FY");

      String s4=new String("SY");

      Float s5=new Float(1.1f);

      Float s6=new Float(1.2f);

      v.addElement(s1);

      v.addElement(s2);

      v.addElement(s3);

      v.addElement(s4);

      v.addElement(s5);
```

```
        v.addElement(s6);
        System.out.println(v);
        v.removeElement(s2);
        v.removeElementAt(4);
        System.out.println(v);
    }
}
```

Output:



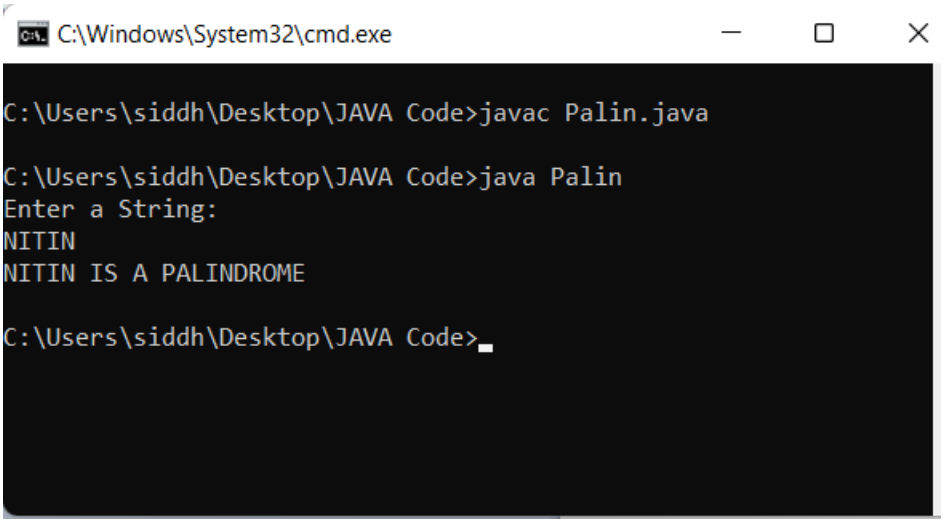
Q13. Write a program to check whether the string provided by the user is palindrome or not.

Ans: Code:

```
import java.util.*;
public class Palin{
        public static void main(String...args){
                String str,rev="";
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter a String: ");
                str=sc.nextLine();
                int length=str.length();
                for(int i=length-1;i>=0;i--){
                        rev=rev+str.charAt(i);
                }
                if(str.equals(rev)){
```

```
                System.out.println(str+ " IS A PALINDROME ");
            }else{
                System.out.println(str+ " IS NOT A PALINDROME ");
            }
        }
}
```

Output:



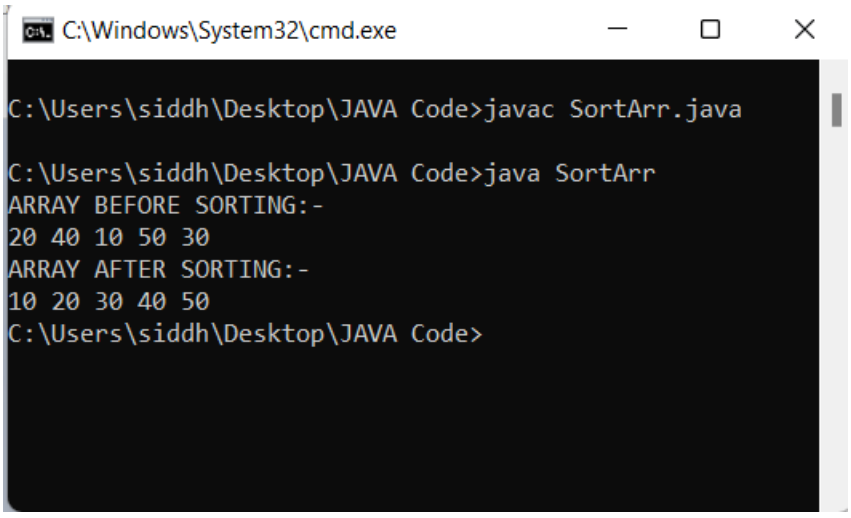
Q14. Write a java program to sort a 1-d array in ascending order using bubble-sort

Ans: Code:

```
public class SortArr{
    public static void main(String...args){
        int arr[]={20,40,10,50,30};
        System.out.println("ARRAY BEFORE SORTING:- ");
        for(int i=0;i<arr.length;i++){
            System.out.print(arr[i]+ " ");
        }
        System.out.println("");
        int n=arr.length;
```

```
                int temp;
                for(int i=0;i<n;i++){
                        for(int j=1;j<n-i;j++){
                                if(arr[j-1]>arr[j]){
                                        temp=arr[j-1];
                                        arr[j-1]=arr[j];
                                        arr[j]=temp;
                                }
                        }
                }
                System.out.println("ARRAY AFTER SORTING:- ");
                for(int i=0;i<arr.length;i++){
                        System.out.print(arr[i]+ " ");
                }
        }
}
```

Output:



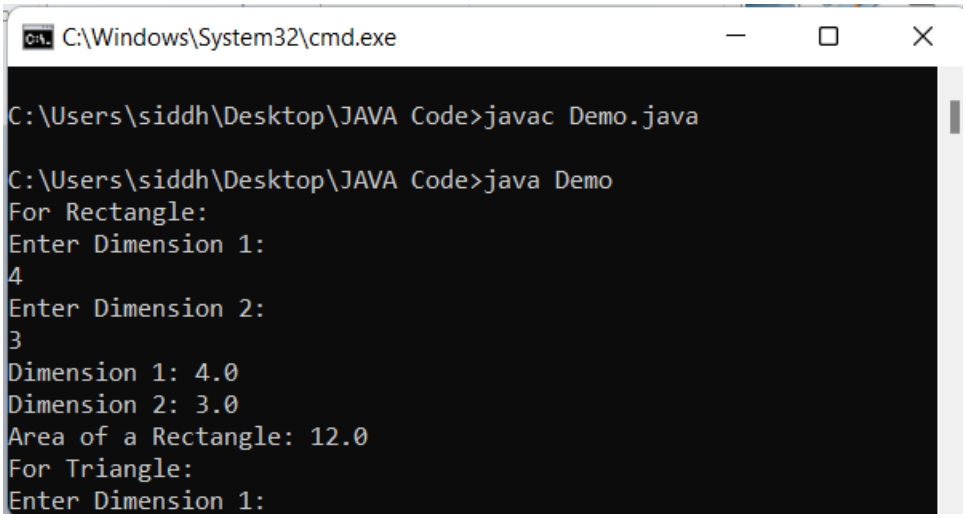
Q15. Write a program to show the hierarchical Inheritance.

Ans: Code:

```
import java.util.*;
abstract class Shape{
	float dim1,dim2;
	void getdata(){
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter Dimension 1: ");
		dim1=sc.nextInt();
		System.out.println("Enter Dimension 2: ");
		dim2=sc.nextInt();
	}
	void display(){
		System.out.println("Dimension 1: "+dim1);
		System.out.println("Dimension 2: "+dim2);
	}
	abstract void area();
}
class Rectangle extends Shape{
	void getdata(){
		super.getdata();
	}
	void area(){
		float area1=dim1*dim2;
		System.out.println("Area of a Rectangle: "+area1);
	}
}
class Triangle extends Shape{
	void getdata(){
```

```
        super.getdata();
    }
    void area(){
        float area2=0.5f*dim1*dim2;
        System.out.println("Area of a Triangle: "+area2);
    }
}
class Demo{
    public static void main(String...args){
        Rectangle r=new Rectangle();
        System.out.println("For Rectangle: ");
        r.getdata();
        r.display();
        r.area();
        Triangle t=new Triangle();
        System.out.println("For Triangle: ");
        t.getdata();
        t.display();
        t.area();
    }
}
```

Output:



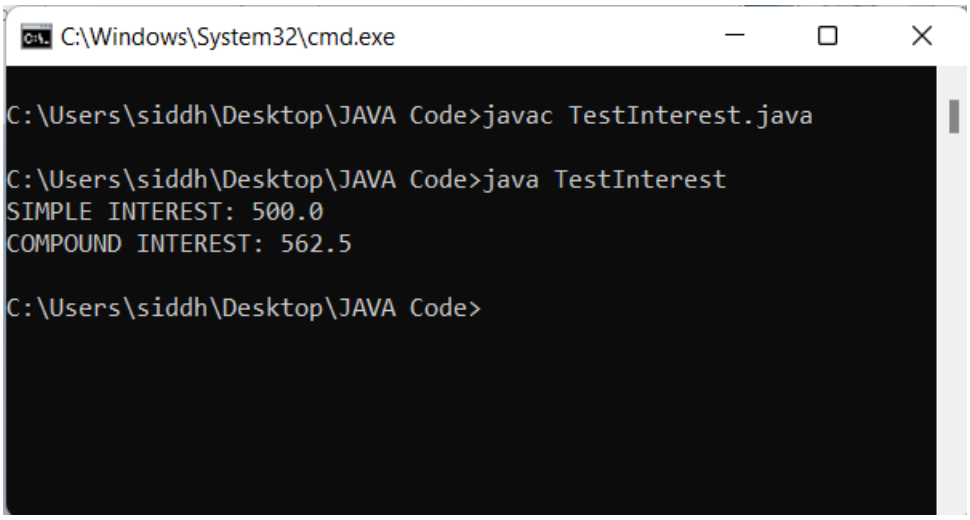
Q16. Develop an “Interest” interface, which contains Simple Interest and Compound Interest methods and static final field of rate 25%. Write a class to implement those methods.

Ans: Code:

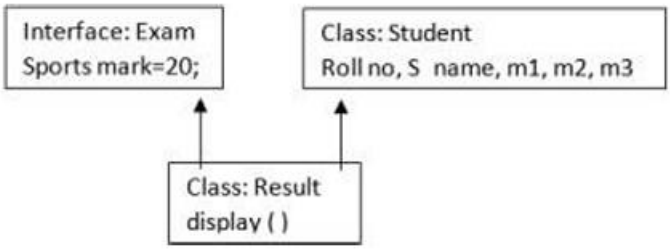
```
import java.util.*;

import static java.lang.Math.pow;

interface Interest{

        float roi=25;

        public void simpleInterest(float principal,float time);

        public void compoundInterest(float principal,float time);

}

public class TestInterest implements Interest{

        public void simpleInterest(float principal,float time){

                float si=((principal*roi*time)/100);

                System.out.println("SIMPLE INTEREST: "+si);

        }

        public void compoundInterest(float principal,float time){

                double ci=principal*(Math.pow((1.0+(roi/100)),time))-principal;

                System.out.println("COMPOUND INTEREST: "+ci);
```

```
    }
    public static void main(String...args){
        TestInterest t=new TestInterest();
        t.simpleInterest(1000,2);
        t.compoundInterest(1000,2);
    }
}
```

Output:



Q17. Write a program to implement the following inheritance.



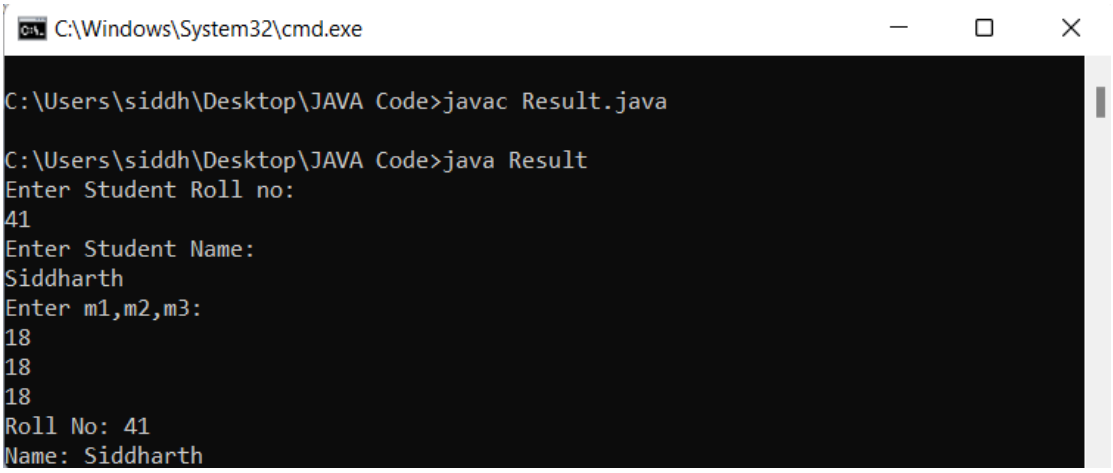
Ans: Code:

```
import java.util.*;
interface Exam{
    int sportsmarks=20;
```

```
	void display();
}
class Student{
	int rollno,m1,m2,m3;
	String name;
	void getd(){
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter Student Roll no: ");
		rollno=sc.nextInt();
		System.out.println("Enter Student Name: ");
		name=sc.next();
		System.out.println("Enter m1,m2,m3: ");
		m1=sc.nextInt();
		m2=sc.nextInt();
		m3=sc.nextInt();
	}
	void display(){
		System.out.println("Roll No: "+rollno);
		System.out.println("Name: "+name);
		System.out.println("Marks 1: "+m1);
		System.out.println("Marks 2: "+m2);
		System.out.println("Marks 3: "+m3);
	}
}
class Result extends Student implements Exam{
	public void display(){
		super.display();
		System.out.println("Sports Marks: "+sportsmarks);
```

```
        float result=m1+m2+m3+sportsmarks;
        System.out.println("Result: "+result);
    }
    public static void main(String...args){
        Result r=new Result();
        r.getd();
        r.display();
    }
}
```

Output:



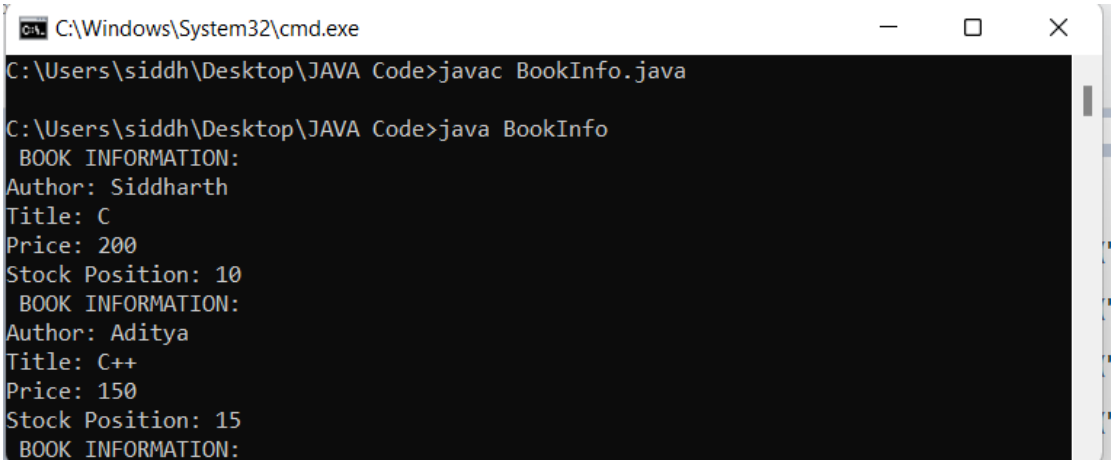
Q18. Develop a program to create a class “Book” having data members “author”, “title” and “price”. Derive a class “BookInfo” having data member “stockposition” and “method to initialize and display the information for three objects.

Ans: code:

```
class Book{
    String author,title;
    int price;
    Book(String a,String t,int p){
        author=a;
        title=t;
```

```
                price=p;
        }
}
class BookInfo extends Book{
        int stockposition;
        BookInfo(String a,String t,int p,int sp){
                super(a,t,p);
                stockposition=sp;
        }
        void display(){
                System.out.println(" BOOK INFORMATION: ");
                System.out.println("Author: "+author);
                System.out.println("Title: "+title);
                System.out.println("Price: "+price);
                System.out.println("Stock Position: "+stockposition);
        }
        public static void main(String...args){
                BookInfo bi[]=new BookInfo[3];
                bi[0]=new BookInfo("Siddharth","C",200,10);
                bi[1]=new BookInfo("Aditya","C++",150,15);
                bi[2]=new BookInfo("Darshan","JAVA",100,20);
                for(int i=0;i<bi.length;i++){
                        bi[i].display();
                }
        }
}
```

Output:



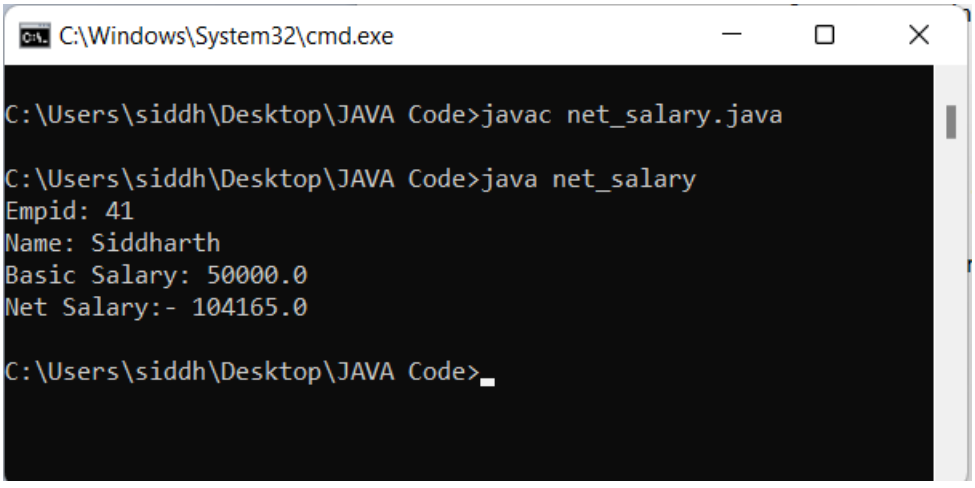
Q19. Write a program, to create a class “Salary” with data members “empid”, “name” and “basicsalary”. Write an interface “Allowance” which stores rates of calculation for da 90% of basic salary, hra as 10% of basic salary and pf as 8.33% of basic salary. Include a method to calculate net salary and display it

Ans: Code:

```
interface Allowance{
        float da=0.9f;
        float hra=0.1f;
        float pf=0.0833f;
        void display();
}
class Salary{
        int empid;
        String name;
        float bsalary;
        Salary(int empid,String name,float bsalary){
                this.empid=empid;
                this.name=name;
                this.bsalary=bsalary;
        }
        void display(){
```

```
                System.out.println("Empid: "+empid);
                System.out.println("Name: "+name);
                System.out.println("Basic Salary: "+bsalary);
        }
}
class net_salary extends Salary implements Allowance{
        net_salary(int empid,String name,float bsalary){
                super(empid,name,bsalary);
        }
        public void display(){
                super.display();
                float ns=bsalary+(bsalary*da)+(bsalary*hra)+(bsalary*pf);
                System.out.println("Net Salary:- "+ns);
        }
        public static void main(String...args){
                net_salary n=new net_salary(41,"Siddharth",50000f);
                n.display();
        }
}
```

Output:

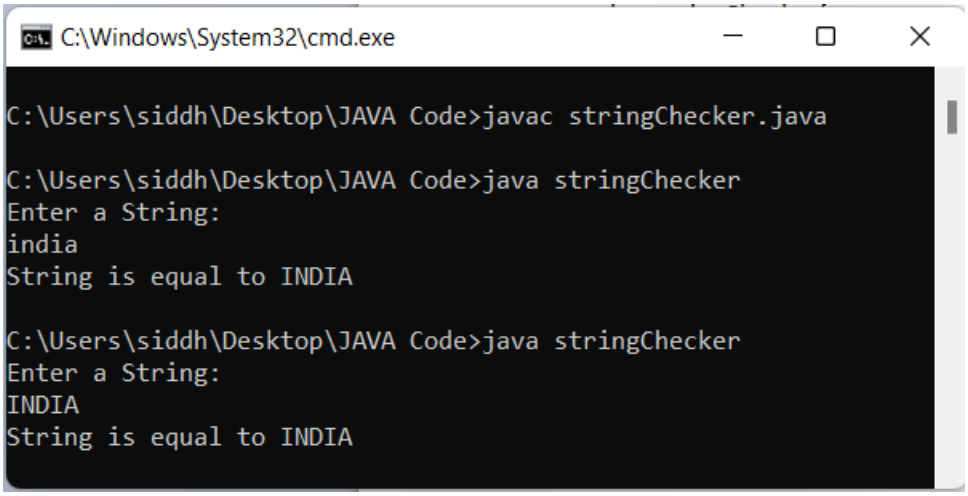


Q20. Write a program that throws an exception called “NoMatchException” when a string is not equal to “India”.

Ans: Code:

```
import java.io.*;

import java.lang.*;

import java.util.*;

class NoMatchException extends Exception{

        NoMatchException(String msg){

                super(msg);

        }

}

class stringChecker{

        public static void main(String...args){

                Scanner sc=new Scanner(System.in);

                String s1;

                try{

                        System.out.println("Enter a String: ");

                        s1=sc.nextLine();

                        if(s1.equalsIgnoreCase("INDIA")){

                                System.out.println("String is equal to INDIA ");

                        }else{

                                throw new NoMatchException("String not equal to INDIA");

                        }

                }catch(Exception e){

                        System.out.println(e);

                }

        }

}
```

Output:



Q21. Write a program to create a user defined exception in java.

Ans: Code:

```
import java.util.*;
import java.lang.*;
import java.io.*;
class MyException extends Exception{
        MyException(String msg){
                super(msg);
        }
}
class ageTest{
        public static void main(String...args){
                Scanner sc=new Scanner(System.in);
                int age;
                try{
                        System.out.println("Enter Age: ");
                        age=sc.nextInt();
```

```
                    if(age<18){
                        throw new MyException("Invalid Age ");
                    }else{
                        System.out.println("Valid Age ");
                    }
                }catch(Exception e){
                    System.out.println(e);
                }
        }
}
```

Output:

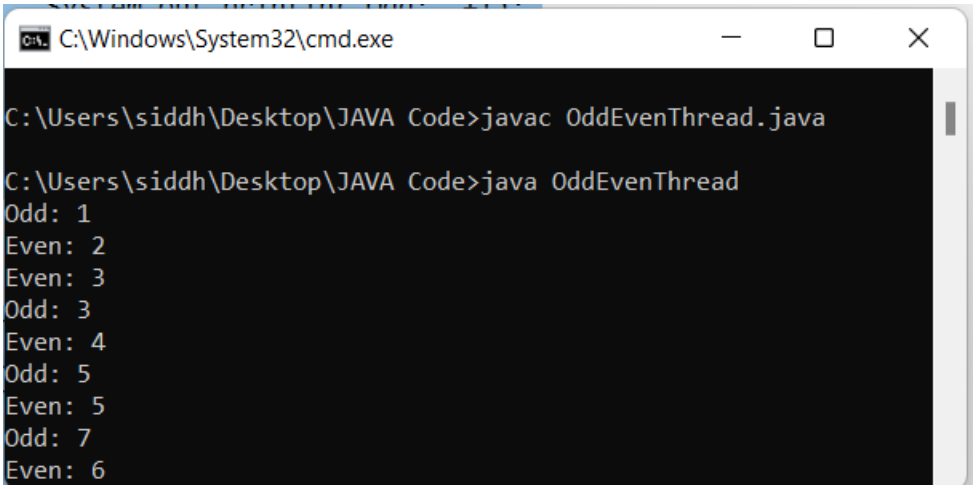
Q22. Write a program to print even and odd number using two threads with delay of 1000ms after each number.

Ans: Code:

```
import java.io.*;

import java.lang.*;

class OddThread extends Thread{
        public void run(){
                for(int i=1;i<=20;i+=2){
                        System.out.println("Odd: "+i);
```

```
                try{
                        sleep(1000);
                }catch(Exception e){
                        System.out.println("Error");
                }
        }
    }
}
class EvenThread extends Thread{
    public void run(){
        for(int i=2;i<=20;i++){
                System.out.println("Even: "+i);
                try{
                        sleep(1000);
                }catch(Exception e){
                        System.out.println("Error");
                }
        }
    }
}
class OddEvenThread{
    public static void main(String...args){
        OddThread ot=new OddThread();
        EvenThread et=new EvenThread();
        ot.start();
        et.start();
    }
}
```

Output:



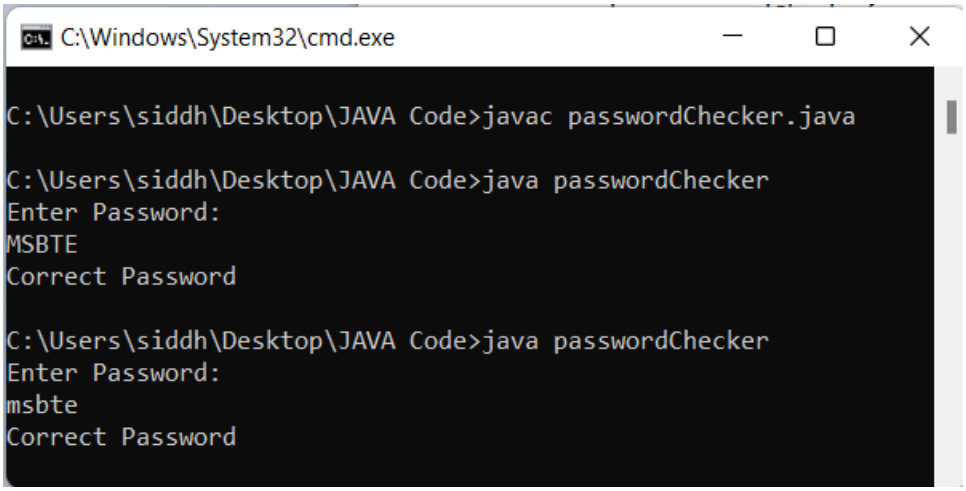
Q23. Define an exception called “NoMatchException” that is thrown when the password accepted is not equal to “MSBTE”. Write the program.

Ans: Code:

```
import java.util.*;

import java.lang.*;

import java.io.*;

class NoMatchException extends Exception{

        NoMatchException(String msg){

                super(msg);

        }

}

class passwordChecker{

        public static void main(String...args){

                Scanner sc=new Scanner(System.in);

                String s1;

                try{

                        System.out.println("Enter Password: ");

                        s1=sc.nextLine();
```

```
                    if(s1.equalsIgnoreCase("MSBTE")){
                        System.out.println("Correct Password ");
                    }else{
                        throw new NoMatchException("Wrong Password ");
                    }
                }catch(Exception e){
                    System.out.println(e);
                }
        }
}
```

Output:



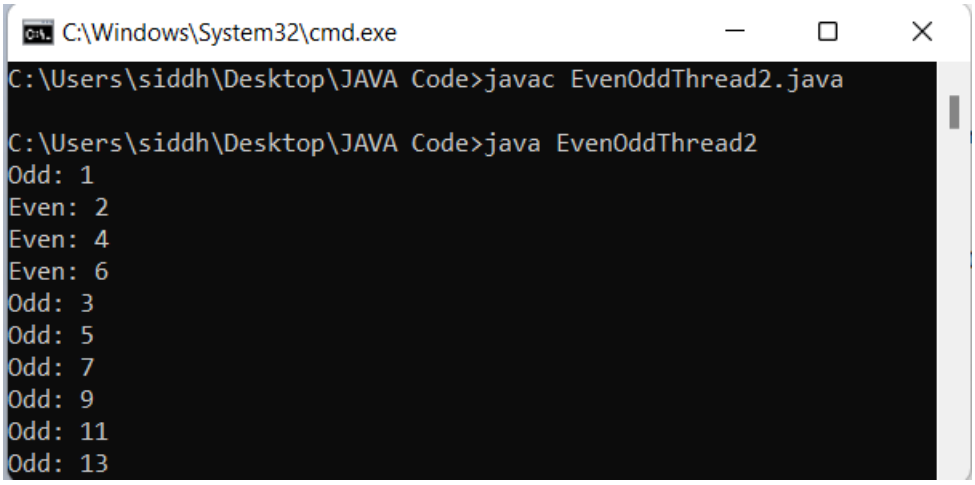
Q24. Write a Java program in which thread A will display the even numbers between1 to 50 and thread B will display the odd numbers between1 to 50. After 3 iterations thread A should go to sleep for 50ms.

Ans: Code:

```
import java.io.*;

import java.lang.*;

class EvenNoThread extends Thread{
        public void run(){
                for(int i=2;i<=50;i+=2){
```

```
                System.out.println("Even: "+i);
                try{
                        if(i%6==0){
                                sleep(50);
                        }
                }catch(Exception e){
                        System.out.println(e);
                }
        }
    }
}
class OddNoThread extends Thread{
    public void run(){
        for(int i=1;i<=50;i+=2){
                System.out.println("Odd: "+i);
        }
    }
}
class EvenOddThread2{
    public static void main(String...args){
        EvenNoThread et=new EvenNoThread();
        OddNoThread ot=new OddNoThread();
        et.start();
        ot.start();
    }
}
```

Output:



Q25. Write an applet program for following graphics method. i) Drawoval () ii) Drawline ().

Ans: Code:

```
import java.awt.*;
import java.applet.*;
public class shape extends Applet{
        public void paint(Graphics g){
                g.drawOval(100,100,190,190);
                g.drawLine(100,300,290,300);
        }
}
/*<applet code="shape.class" height=600 width=600></applet>*/
```

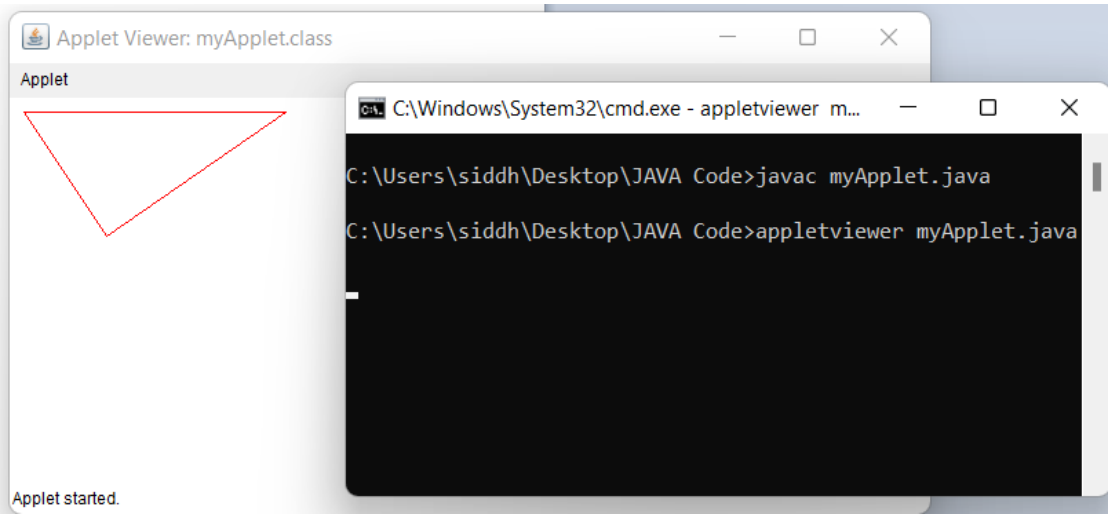
Output:

Q26. Write a java applet to display the following output in Red.

Ans: Code:

```
import java.awt.*;
import java.applet.*;
public class myApplet extends Applet{
	public void paint(Graphics g){
		int x[]={10,200,70};
		int y[]={10,10,100};
		g.setColor(Color.red);
		g.drawPolygon(x,y,3);
	}
}
/* <applet code="myApplet.class" height=600 width=600 > </applet> */
```

Output:



Q27. Write a program to read a file and then count number of words.

Ans: Code:

```
import java.io.*;


public class Demo1 {
   public static void main(String... args) throws IOException {
      File file = new File("hello.java");
      FileInputStream fileInputStream = new FileInputStream(file);
      InputStreamReader inputStreamReader = new InputStreamReader(fileInputStream);
      BufferedReader bufferedReader = new BufferedReader(inputStreamReader);
      String line;
      int wordCount = 0;
      int paraCount = 0;
      while ((line = bufferedReader.readLine()) != null) {
         if (line.equals("")) {
            paraCount++;
         } else {
            String[] words = line.split("\\s+");
```

```
            wordCount += words.length;
          }
        }
        // Close the BufferedReader to release resources
        bufferedReader.close();

        // Print the word count and paragraph count
        System.out.println("Total word count: " + wordCount);
        System.out.println("Total paragraph count: " + paraCount);
    }
}
```

Output:

```
PS C:\Users\siddh\Desktop\JAVA Code>  & 'C:\Program Fil
h\AppData\Roaming\Code\User\workspaceStorage\ec82ee310e
Total word count: 14
Total paragraph count: 0
PS C:\Users\siddh\Desktop\JAVA Code>
```

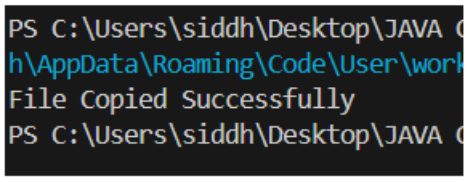
Q28. Write a program to append content of one file into another file.

Ans: Code:

```
import java.io.*;
public class FileCopy {
    public static void main(String...args) throws IOException{
        FileReader fr=new FileReader("hello.java");
        FileWriter fw=new FileWriter("hello1.java");
        int ch;
        try{
            while((ch=fr.read())!=-1){
```

```
            fw.write(ch);
        }
        System.out.println("File Copied Successfully ");
        fr.close();
        fw.close();
    }
    finally{
        if(fr!=null)
        fr.close();
        if(fw!=null)
        fw.close();
    }
  }
}
```

Output:



Q29. Write a program for reading and writing character to and from the given files using character stream classes.

Ans: Code:

```
import java.io.*;

public class Demo3 {

   public static void main(String...args) throws IOException{

      File file=new File("hello.java");

      FileReader fr=new FileReader(file);

      char chars[]=new char[(int)file.length()];
```

```
        fr.read(chars);
        FileWriter fw=new FileWriter("Demo1.java");
        fw.write(chars);
        fw.flush();
        System.out.println("Data successfully written in specified file");
    }
}
```

Output:

```
PS C:\Users\siddh\Desktop\JAVA Code>  & 'C:\Program
h\AppData\Roaming\Code\User\workspaceStorage\ec82ee3
Data successfully written in specified file
PS C:\Users\siddh\Desktop\JAVA Code>
```

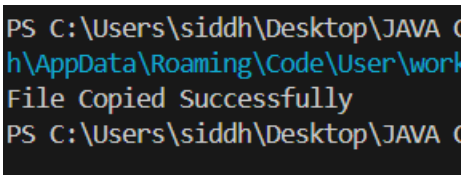
Q30. Write a Java program to copy the content of one file into another.

Ans: Code:

```
import java.io.*;
public class FileCopy {
    public static void main(String...args) throws IOException{
        FileReader fr=new FileReader("hello.java");
        FileWriter fw=new FileWriter("hello1.java");
        int ch;
        try{
            while((ch=fr.read())!=-1){
                fw.write(ch);
            }
            System.out.println("File Copied Successfully ");
            fr.close();
```

```
            fw.close();
        }
        finally{
            if(fr!=null)
            fr.close();
            if(fw!=null)
            fw.close();
        }
    }
}
```

Output:



Q31. Write a Java program to count the number of words from a text file using stream classes.

Ans: Code:

```
import java.io.*;
public class FileWordCount {
    public static void main(String...args) throws IOException{
        FileReader fr=new FileReader("hello.java");
        int ch,c=0;
        try{
            while((ch=fr.read())!=-1){
                c++;
            }
            System.out.println(c);
```

```
            fr.close();
        }
        finally{
            if(fr!=null)
            fr.close();
        }
    }
}
```

Output: