INDEX

PROGRAM NO. 1

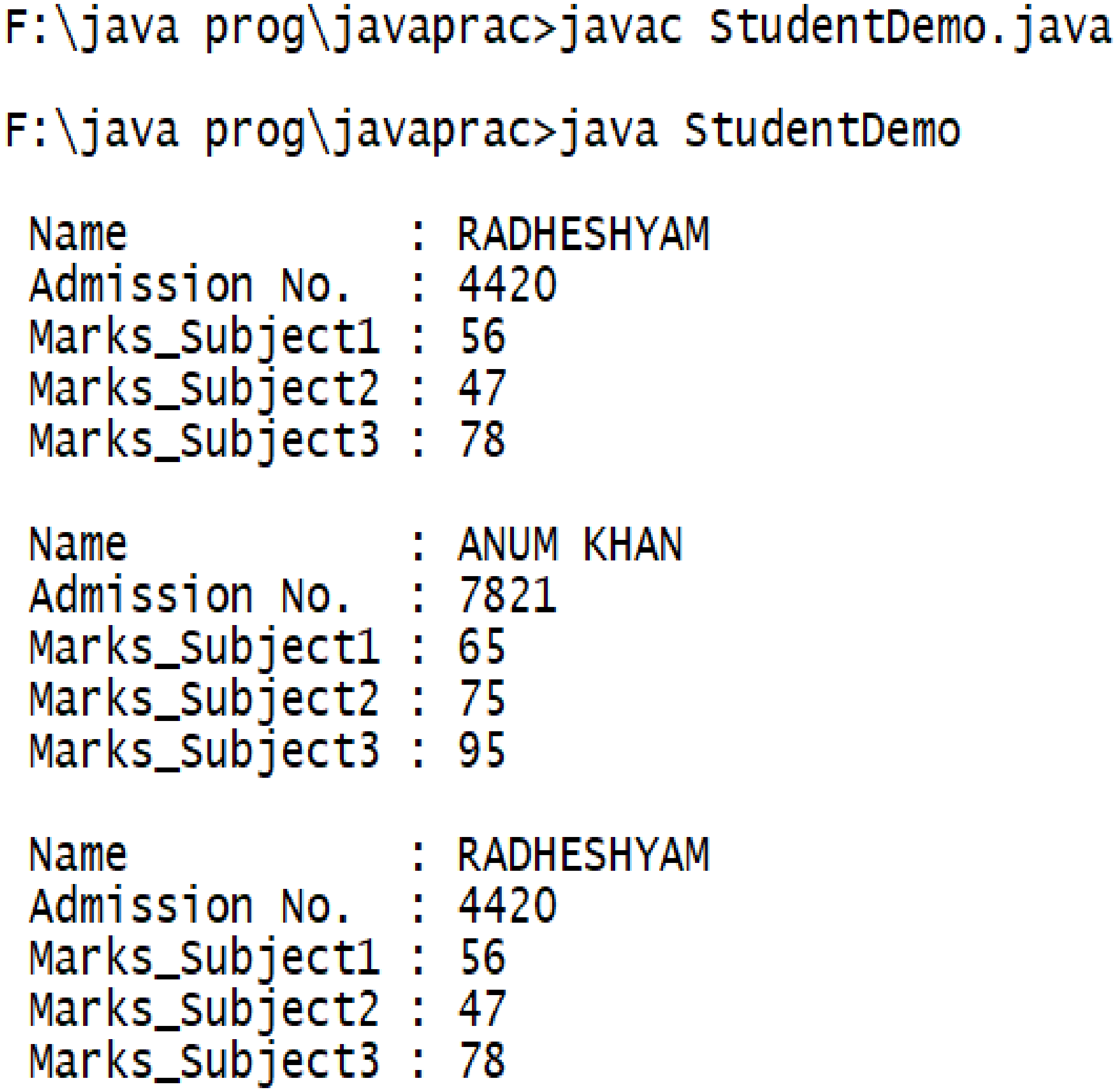
Program to define a CLASS, describe its constructor, overload the Constructors and instantiate its OBJECT

Code:

```
class Student
{
	String Name;
	int Admno;
	int Marks1,Marks2,Marks3;
	Student()					// null constructor
	{
		Name="RADHESHYAM";
		Admno=4420;
		Marks1=56;
		Marks2=47;
		Marks3=78;
	}
	Student(String STR,int Anum,int m1,int m2,int m3)
	{						// parameterized constructor
		Name=STR;
		Admno=Anum;
		Marks1=m1;
		Marks2=m2;
		Marks3=m3;
	}
	Student(Student STUD)		// copy constructor
	{
		Name=STUD.Name;
		Admno=STUD.Admno;
		Marks1=STUD.Marks1;
		Marks2=STUD.Marks2;
		Marks3=STUD.Marks3;
	}
	void display()
	{
```

```
                System.out.println("\n Name           : "+Name);
                System.out.println(" Admission No.  : "+Admno);
                System.out.println(" Marks_Subject1         : "+Marks1);
                System.out.println(" Marks_Subject2         : "+Marks2);
                System.out.println(" Marks_Subject3         : "+Marks3);
        }
}

class StudentDemo
{
        public static void main(String arg[])
        {
                Student STUD1=new Student();
                Student STUD2=new Student("ANUM KHAN",7821,65,75,95);
                Student STUD3=new Student(STUD1);
                STUD1.display();
                STUD2.display();
                STUD3.display();
        }
}
```

Output:



```
F:\java prog\javaprac>javac StudentDemo.java

F:\java prog\javaprac>java StudentDemo

 Name            : RADHESHYAM
 Admission No.   : 4420
 Marks_Subject1 : 56
 Marks_Subject2 : 47
 Marks_Subject3 : 78

 Name            : ANUM KHAN
 Admission No.   : 7821
 Marks_Subject1 : 65
 Marks_Subject2 : 75
 Marks_Subject3 : 95

 Name            : RADHESHYAM
 Admission No.   : 4420
 Marks_Subject1 : 56
 Marks_Subject2 : 47
 Marks_Subject3 : 78
```

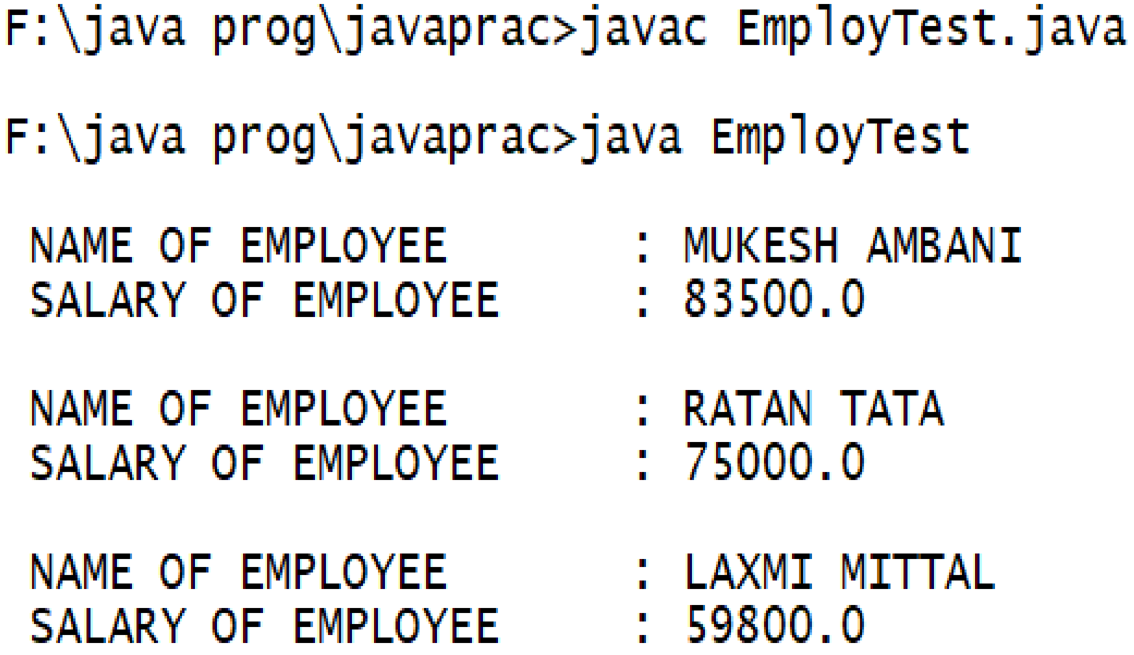
PROGRAM NO. 2

Program to implement ARRAY of objects.

Code:

```
class Employee
{
        private String name;
        private double salary;
        public Employee(String str, double sal)
        {
                name = str;
                salary = sal;
        }
        public void print()
        {
                System.out.println("\n NAME OF EMPLOYEE      : "+name);
                System.out.println(" SALARY OF EMPLOYEE      : "+salary);
        }
}
public class EmployTest
{
        public static void main(String[] args)
        {
                Employee[] staffs = new Employee[3];
                staffs[0] = new Employee("MUKESH AMBANI", 83500);
                staffs[1] = new Employee("RATAN TATA", 75000);
                staffs[2] = new Employee("LAXMI MITTAL", 59800);
                for (int i=0;i<3;i++)
                        staffs[i].print();
        }
}
```

Output:



```
F:\java prog\javaprac>javac EmployTest.java

F:\java prog\javaprac>java EmployTest

 NAME OF EMPLOYEE          : MUKESH AMBANI
 SALARY OF EMPLOYEE        : 83500.0

 NAME OF EMPLOYEE          : RATAN TATA
 SALARY OF EMPLOYEE        : 75000.0

 NAME OF EMPLOYEE          : LAXMI MITTAL
 SALARY OF EMPLOYEE        : 59800.0
```

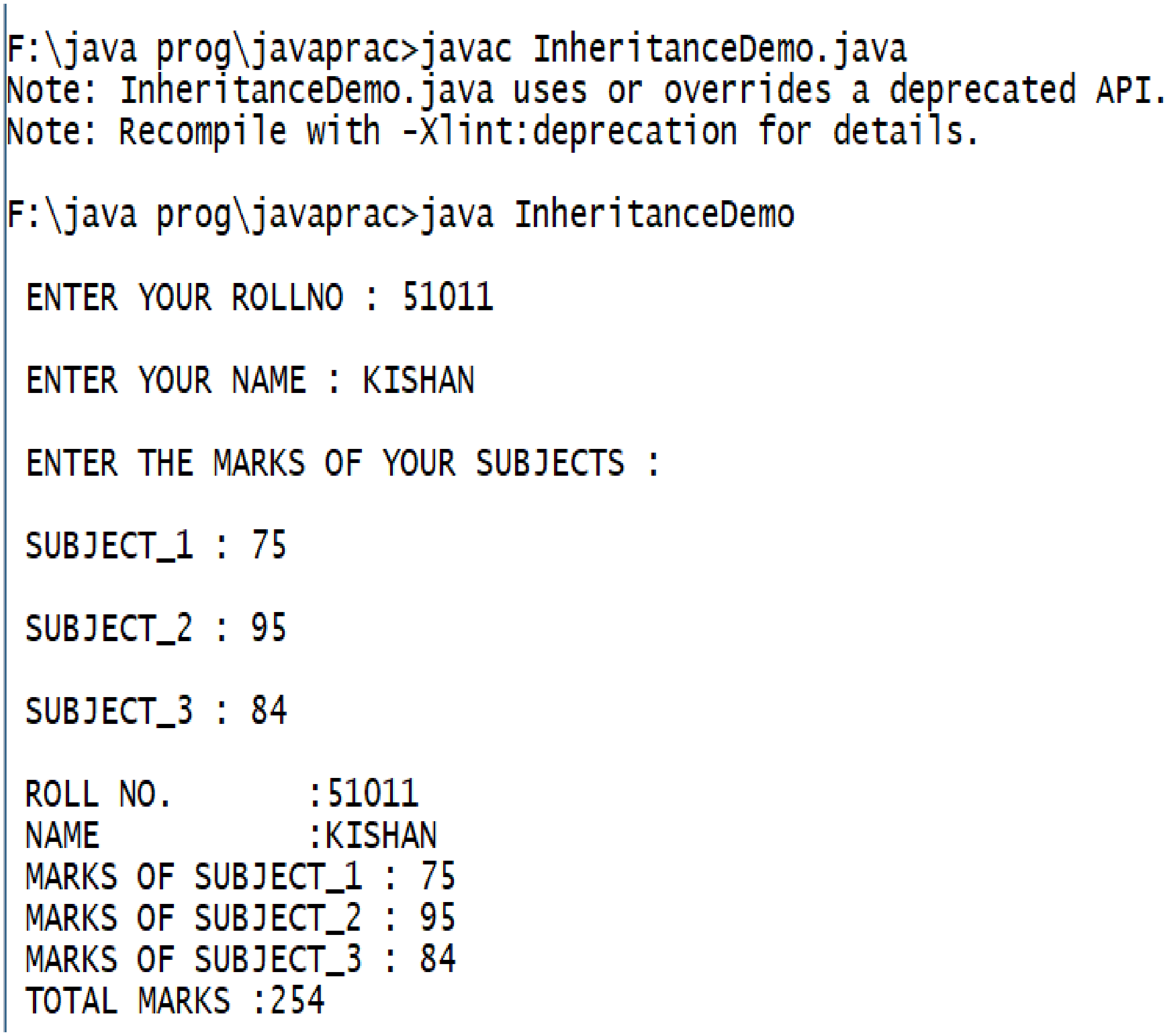
PROGRAM NO. 3

Program to implement multilevel INHERITANCE by applying various access controls to its data members and methods.

Code:

```
import java.io.DataInputStream;

class Student
{
        private int rollno;
        private String name;
        DataInputStream dis=new DataInputStream(System.in);
        public void getRollno()
        {
                try
                {
                        System.out.print("\n ENTER YOUR ROLLNO : ");
                        rollno=Integer.parseInt(dis.readLine());
                        System.out.print("\n ENTER YOUR NAME : ");
                        name=dis.readLine();
                        System.out.println("");
                }
                catch(Exception e){ }
        }
        void putRollno()
        {
                System.out.println(" ROLL NO.         :"+rollno);
                System.out.println(" NAME             :"+name);
        }
}
class Marks extends Student
{
        protected int Marks1,Marks2,Marks3;
        void getMarks()
        {
                try
                {
                        System.out.println(" ENTER THE MARKS OF YOUR SUBJECTS :");
                        System.out.print("\n SUBJECT_1 : ");
                        Marks1=Integer.parseInt(dis.readLine());
                        System.out.print("\n SUBJECT_2 : ");
                        Marks2=Integer.parseInt(dis.readLine());
                        System.out.print("\n SUBJECT_3 : ");
                        Marks3=Integer.parseInt(dis.readLine());
                        System.out.print("\n");
                }
                catch(Exception e) { }
        }
        void putMarks()
        {
                System.out.println(" MARKS OF SUBJECT_1 : "+ Marks1);
                System.out.println(" MARKS OF SUBJECT_2 : "+ Marks2);
                System.out.println(" MARKS OF SUBJECT_3 : "+ Marks3);
        }
}
class Result extends Marks
{
        private int total;
        void computeDisplay()
        {
                total=Marks1+Marks2+Marks3;
                System.out.println(" TOTAL MARKS :" +total);
        }
}
class InheritanceDemo
{
        public static void main(String arg[])
        {
                Result reslt=new Result();
                reslt.getRollno();
                reslt.getMarks();
                reslt.putRollno();
                reslt.putMarks();
                reslt.computeDisplay();
        }
}
```

Output:



```
F:\java prog\javaprac>javac InheritanceDemo.java
Note: InheritanceDemo.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

F:\java prog\javaprac>java InheritanceDemo

 ENTER YOUR ROLLNO : 51011

 ENTER YOUR NAME : KISHAN

 ENTER THE MARKS OF YOUR SUBJECTS :

 SUBJECT_1 : 75

 SUBJECT_2 : 95

 SUBJECT_3 : 84

 ROLL NO.       :51011
 NAME           :KISHAN
 MARKS OF SUBJECT_1 : 75
 MARKS OF SUBJECT_2 : 95
 MARKS OF SUBJECT_3 : 84
 TOTAL MARKS :254
```

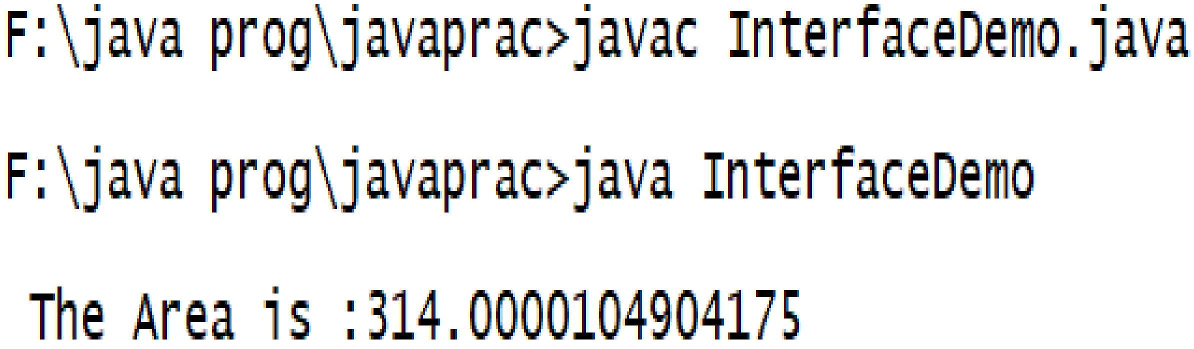
PROGRAM NO. 4

Program to demonstrate use of implementing INTERFACES.

Code:

```
interface Area
{
        float PI=3.14f;
        double compute(double x);
}

interface Display extends Area
{
        void displayResult(double result);
}

class Circle implements Display
{
        public double compute(double radius)
        {
                return(PI*radius*radius);
        }
        public void displayResult(double result)
        {
                System.out.println("\n The Area is :" +result);
        }
}

class InterfaceDemo
{
        public static void main(String a[])
        {
                Circle crcl=new Circle();
                double result=crcl.compute(10);
                crcl.displayResult(result);
        }
}
```

Output:



```
F:\java prog\javaprac>javac InterfaceDemo.java

F:\java prog\javaprac>java InterfaceDemo

 The Area is :314.0000104904175
```

PROGRAM NO. 5

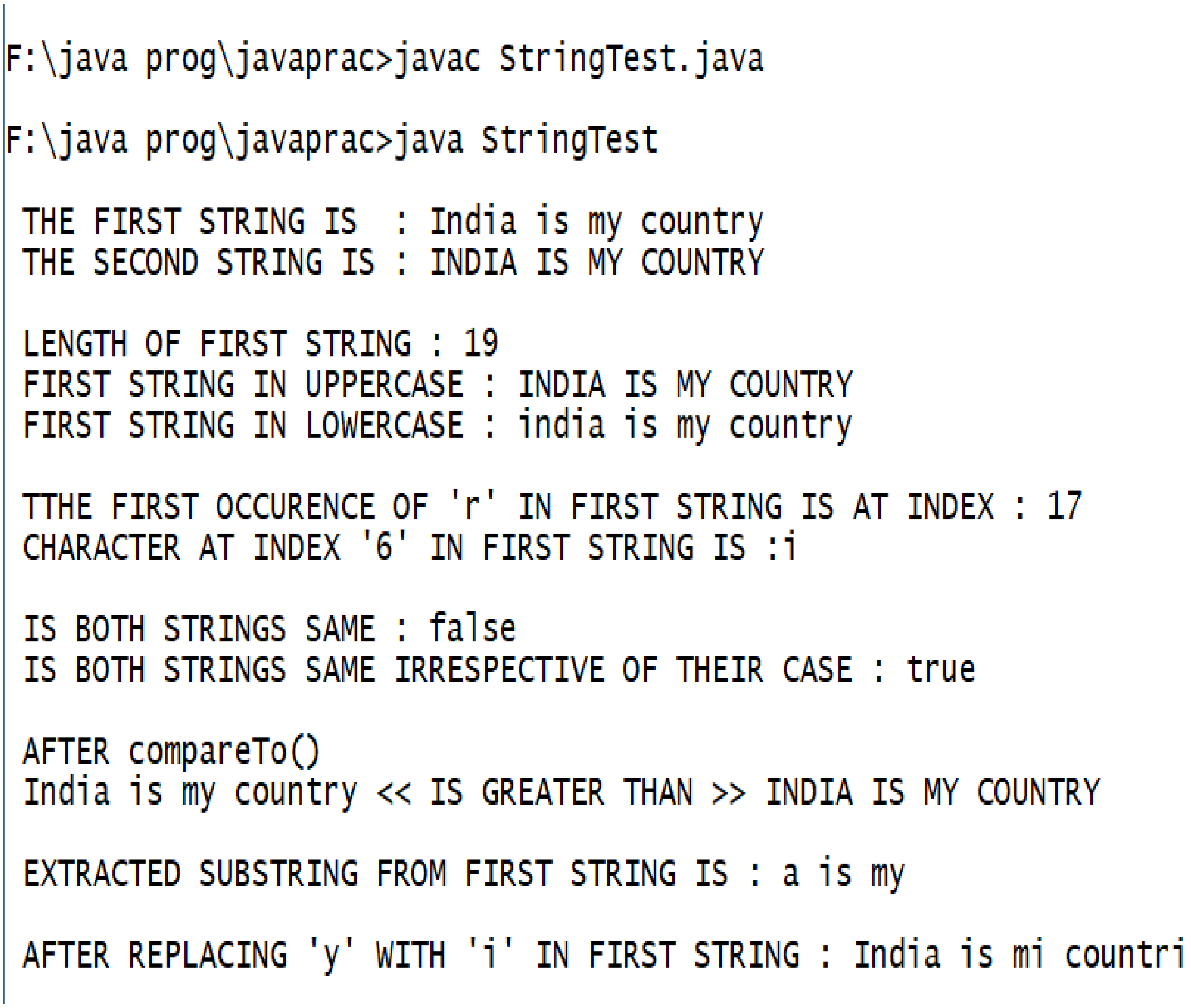
Program to practice using STRING class and its methods.

Code:

```
import java.lang.String;

class StringTest
{
        public static void main(String arg[])
        {
        String str1=new String("India is my country");
        String str2="INDIA IS MY COUNTRY";
        System.out.println("\n THE FIRST STRING IS  : " +str1);
        System.out.println(" THE SECOND STRING IS : " +str2);
        System.out.println("");
        System.out.println(" LENGTH OF FIRST STRING : " +str1.length());
        System.out.println(" FIRST STRING IN UPPERCASE : "
        +str1.toUpperCase());
        System.out.println(" FIRST STRING IN LOWERCASE : "
        +str1.toLowerCase());
        System.out.println("");
        System.out.println(" TTHE FIRST OCCURENCE OF 'r' IN FIRST STRING
        IS AT INDEX : " +str1.indexOf('r'));
        System.out.println(" CHARACTER AT INDEX '6' IN FIRST STRING IS :"
        +str1.charAt(6));
        System.out.println("");
        System.out.println(" IS BOTH STRINGS SAME : " +str1.equals(str2));
        System.out.println(" IS BOTH STRINGS SAME IRRESPECTIVE OF
        THEIR CASE : " +str1.equalsIgnoreCase(str2));
        System.out.println("");
        int result=str1.compareTo(str2);
        System.out.println(" AFTER compareTo()");
            if(result==0)
                System.out.println( " " +str1 + " << IS EQUAL TO >>
        "+str2);
```

```
            else if(result>0)
                System.out.println(" "+ str1 + " << IS GREATER THAN >>
        "+str2);
            else
                System.out.println( " " +str1 + " << IS SMALLER THAN >>
        "+str2);
        System.out.println("");
        String str3=str1.substring(4,12);
        System.out.println(" EXTRACTED SUBSTRING FROM FIRST STRING IS
        : "+str3);
        System.out.println("");
        System.out.println(" AFTER REPLACING 'y' WITH 'i' IN FIRST STRING :
        " + str1.replace('y','i'));
        }
}
```

Output:



```
F:\java prog\javaprac>javac StringTest.java

F:\java prog\javaprac>java StringTest

 THE FIRST STRING IS  : India is my country
 THE SECOND STRING IS : INDIA IS MY COUNTRY

 LENGTH OF FIRST STRING : 19
 FIRST STRING IN UPPERCASE : INDIA IS MY COUNTRY
 FIRST STRING IN LOWERCASE : india is my country

 TTHE FIRST OCCURENCE OF 'r' IN FIRST STRING IS AT INDEX : 17
 CHARACTER AT INDEX '6' IN FIRST STRING IS :i

 IS BOTH STRINGS SAME : false
 IS BOTH STRINGS SAME IRRESPECTIVE OF THEIR CASE : true

 AFTER compareTo()
 India is my country << IS GREATER THAN >> INDIA IS MY COUNTRY

 EXTRACTED SUBSTRING FROM FIRST STRING IS : a is my

 AFTER REPLACING 'y' WITH 'i' IN FIRST STRING : India is mi countri
```

PROGRAM NO. 6

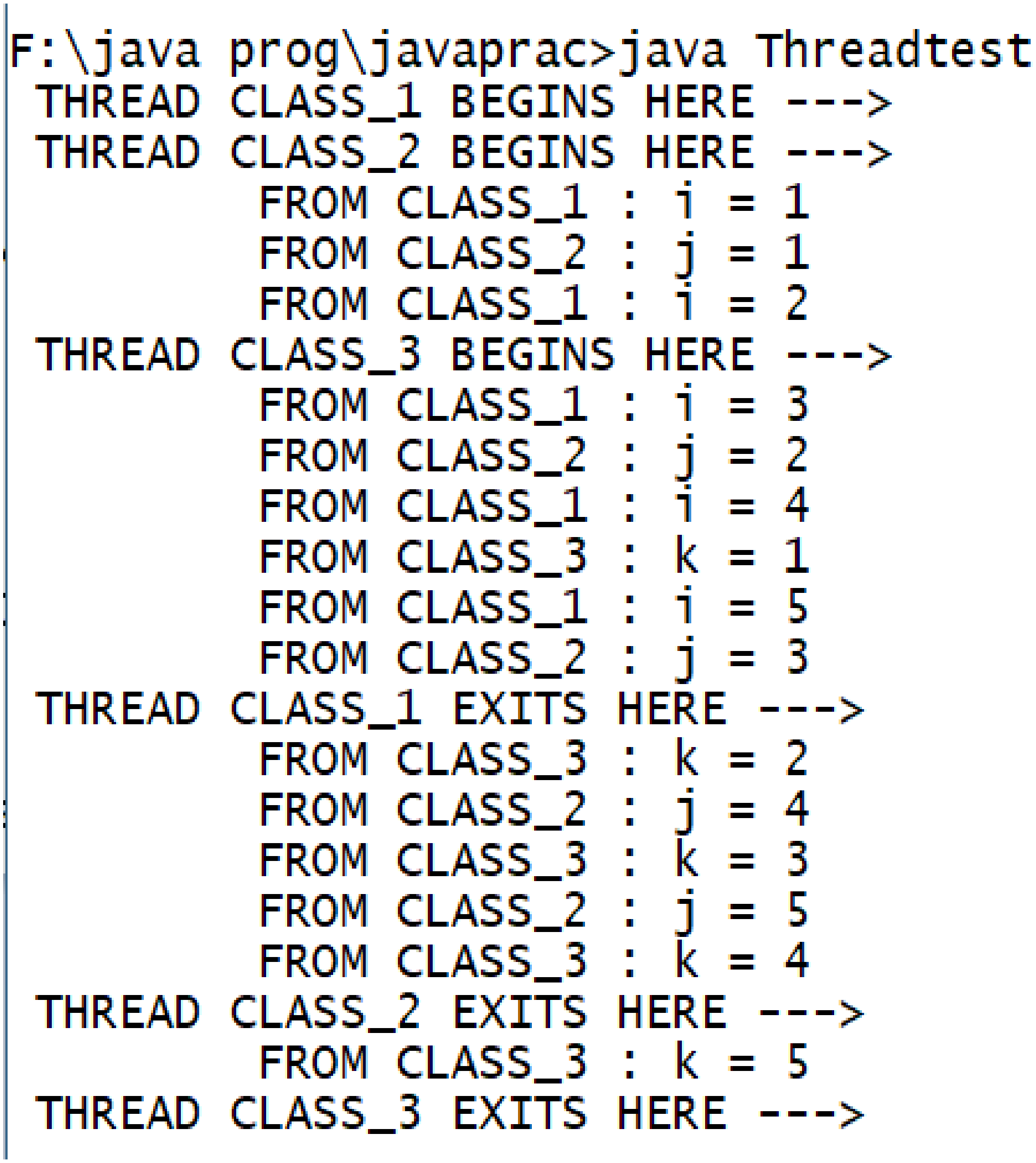
Program to implement the concept of THREADING by extending Thread class

---

Code:

```
import java.lang.*;

class ExceptionHandle
{
        public static void main(String args[])
        {
                int a,b,c,x,y;
                a=Integer.parseInt(args[0]);
                b=Integer.parseInt(args[1]);
                c=Integer.parseInt(args[2]);
                try
                {
                        x=a/(b-c);
                        System.out.println("\n VALUE OF x is "+x);
                }
                catch(ArithmeticException e)
                {
                        System.out.println("\n DIVISION BY ZERO ..hence value of x
        can't be computed!!");
                }
                y=a/(b+c);
                System.out.println("\n VALUE OF y is "+y);
        }
   }
```

Output:



```
F:\java prog\javaprac>java Threadtest
 THREAD CLASS_1 BEGINS HERE --->
 THREAD CLASS_2 BEGINS HERE --->
         FROM CLASS_1 : i = 1
         FROM CLASS_2 : j = 1
         FROM CLASS_1 : i = 2
 THREAD CLASS_3 BEGINS HERE --->
         FROM CLASS_1 : i = 3
         FROM CLASS_2 : j = 2
         FROM CLASS_1 : i = 4
         FROM CLASS_3 : k = 1
         FROM CLASS_1 : i = 5
         FROM CLASS_2 : j = 3
 THREAD CLASS_1 EXITS HERE --->
         FROM CLASS_3 : k = 2
         FROM CLASS_2 : j = 4
         FROM CLASS_3 : k = 3
         FROM CLASS_2 : j = 5
         FROM CLASS_3 : k = 4
 THREAD CLASS_2 EXITS HERE --->
         FROM CLASS_3 : k = 5
 THREAD CLASS_3 EXITS HERE --->
```

PROGRAM NO. 7

Program to implement the concept of THREADING by implementing Runnable interface

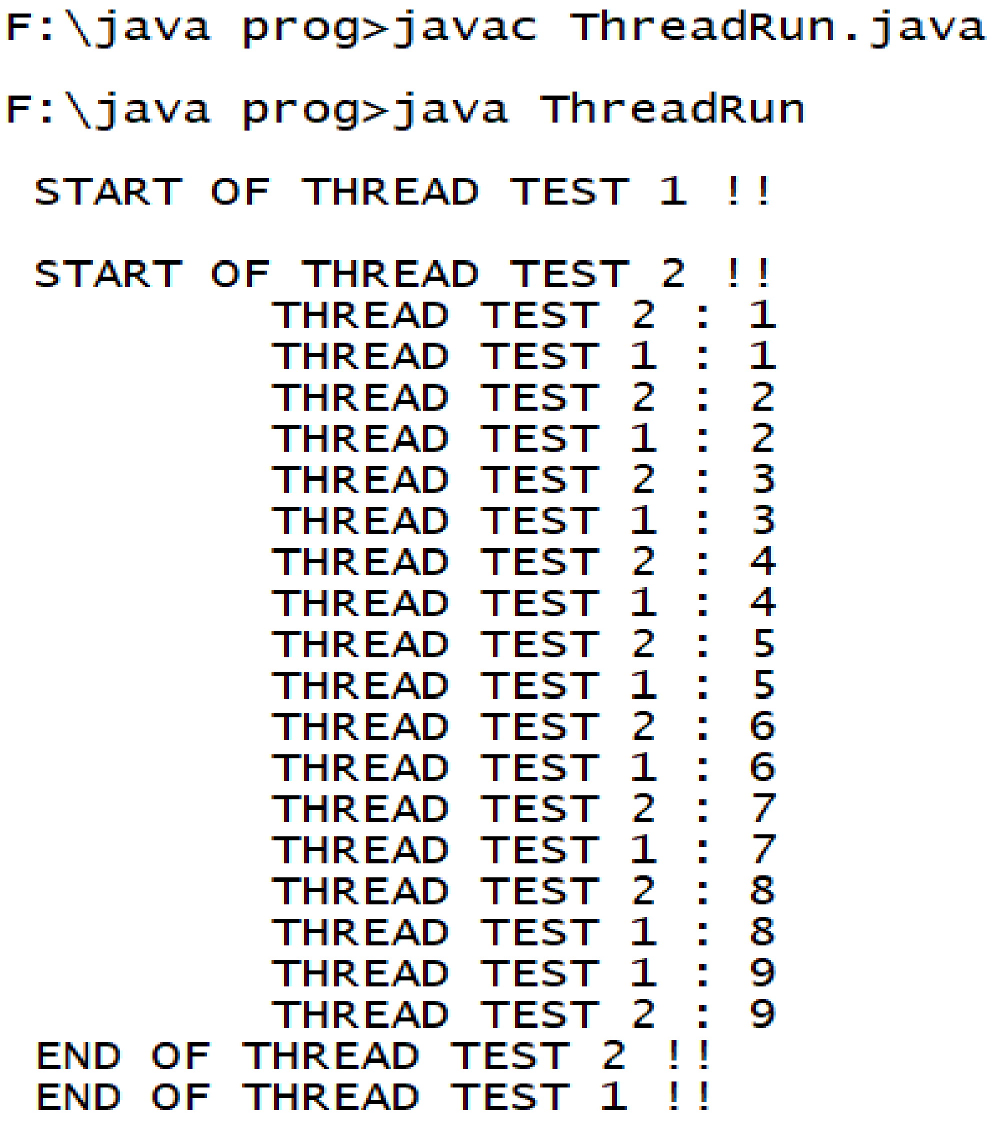
---

Code:

```
import java.lang.Runnable;

class Test1 implements Runnable
{
	public void run()
	{
		System.out.println("START OF THREAD TEST 1 !!");
		for(int i=1;i<10;i++)
		{
			System.out.println("\t THREAD TEST 1 : "+i);
		}
		System.out.println("END OF THREAD TEST 1 !!");
	}
}

class Test2 implements Runnable
{
	public void run()
	{
		System.out.println("START OF THREAD TEST 2 !!");
		for(int i=1;i<10;i++)
		{
			System.out.println("\t THREAD TEST 2 : "+i);
		}
		System.out.println("END OF THREAD TEST 2 !!");
	}
}
```

```
public class ThreadRun
{
        public static void main(String arg[])
        {
                Test1 t1 = new Test1();
                Test2 t2 = new Test2();
                Thread th1 = new Thread(t1);
                Thread th2 = new Thread(t2);
                th1.start();
                th2.start();
        }
}
```

Output:



```
F:\java prog>javac ThreadRun.java

F:\java prog>java ThreadRun

 START OF THREAD TEST 1 !!

 START OF THREAD TEST 2 !!
         THREAD TEST 2 : 1
         THREAD TEST 1 : 1
         THREAD TEST 2 : 2
         THREAD TEST 1 : 2
         THREAD TEST 2 : 3
         THREAD TEST 1 : 3
         THREAD TEST 2 : 4
         THREAD TEST 1 : 4
         THREAD TEST 2 : 5
         THREAD TEST 1 : 5
         THREAD TEST 2 : 6
         THREAD TEST 1 : 6
         THREAD TEST 2 : 7
         THREAD TEST 1 : 7
         THREAD TEST 2 : 8
         THREAD TEST 1 : 8
         THREAD TEST 1 : 9
         THREAD TEST 2 : 9
 END OF THREAD TEST 2 !!
 END OF THREAD TEST 1 !!
```

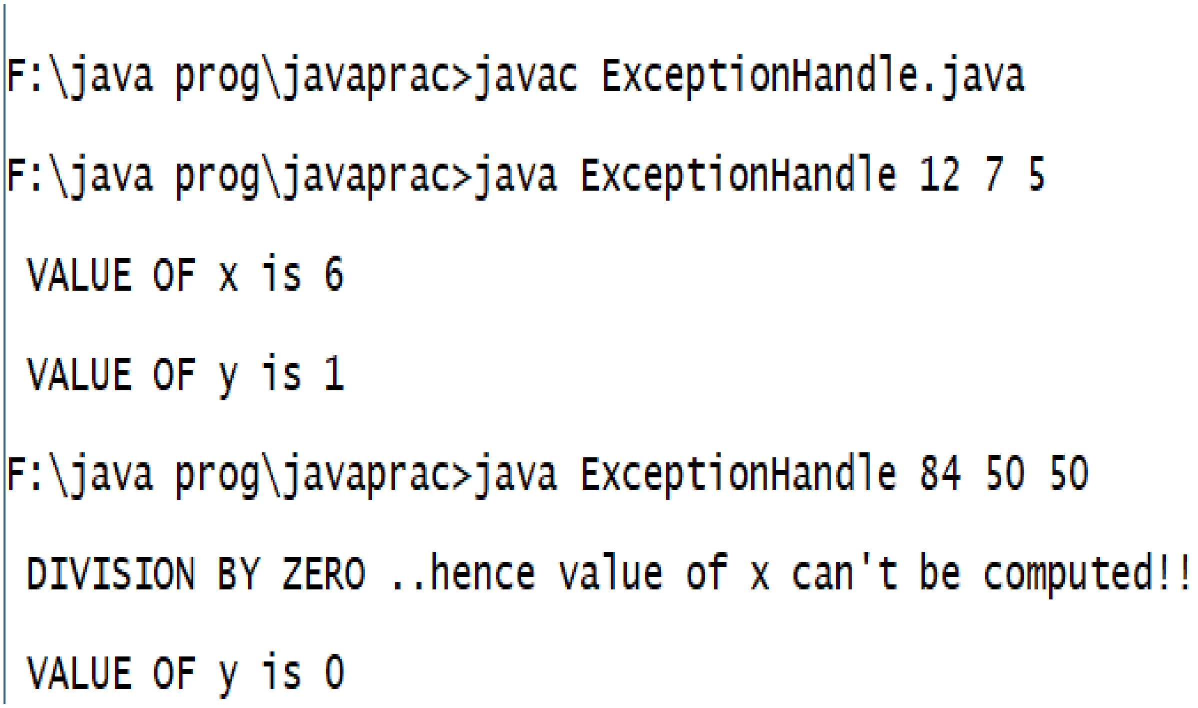
PROGRAM NO. 8

Program to implement the concept of EXCEPTION HANDLING using predefined exception.

Code:

```
import java.lang.Thread;

class FirstClass extends Thread
{
	public void run()
	{
		System.out.println(" THREAD CLASS_1 BEGINS HERE --->");
		for(int i=1;i<=5;i++)
			System.out.println("\t FROM CLASS_1 : i = "+i);
		System.out.println(" THREAD CLASS_1 EXITS HERE --->");
	}
}

class SecondClass extends Thread
{
	public void run()
	{
		System.out.println(" THREAD CLASS_2 BEGINS HERE --->");
		for(int j=1;j<=5;j++)
			System.out.println("\t FROM CLASS_2 : j = "+j);
		System.out.println(" THREAD CLASS_2 EXITS HERE --->");
	}
}

class ThirdClass extends Thread
{
	public void run()
	{
		System.out.println(" THREAD CLASS_3 BEGINS HERE --->");
		for(int k=1;k<=5;k++)
			System.out.println("\t FROM CLASS_3 : k = "+k);
		System.out.println(" THREAD CLASS_3 EXITS HERE --->");
	}
}
class Threadtest
{
	public static void main(String arg[])
	{
		new FirstClass().start();
		new SecondClass().start();
		new ThirdClass().start();
	}
}
```

Output:



```
F:\java prog\javaprac>javac ExceptionHandle.java

F:\java prog\javaprac>java ExceptionHandle 12 7 5

 VALUE OF x is 6

 VALUE OF y is 1

F:\java prog\javaprac>java ExceptionHandle 84 50 50

 DIVISION BY ZERO ..hence value of x can't be computed!!

 VALUE OF y is 0
```

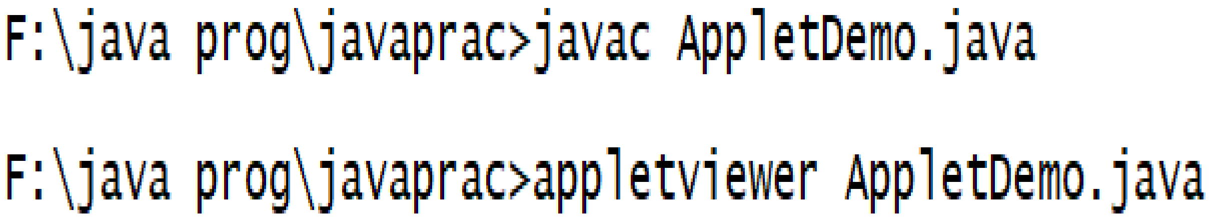
PROGRAM NO. 9

Program using Applet to display a message in the APPLET.

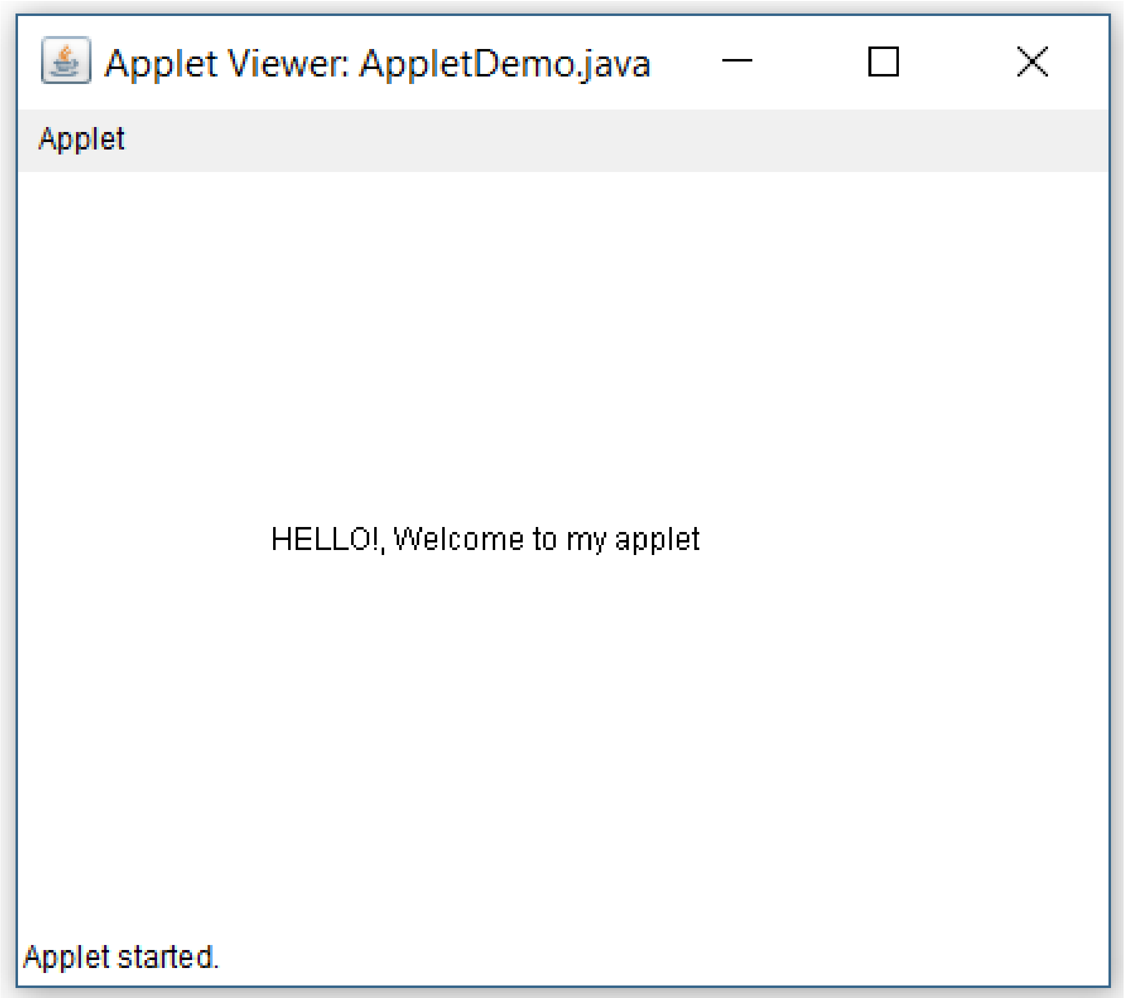
Code:

```
import java.applet.*;
import java.awt.Graphics;

/* <applet code="AppletDemo.java" width=300 height=300> </applet> */

public class AppletDemo extends Applet
{
        public void paint(Graphics g)
        {
                String msg="HELLO!, Welcome to my applet ";
                g.drawString(msg,100,150);
        }
}
```

Output:



```
F:\java prog\javaprac>javac AppletDemo.java

F:\java prog\javaprac>appletviewer AppletDemo.java
```



```
HELLO!, Welcome to my applet
```

```
Applet started.
```

PROGRAM NO. 10

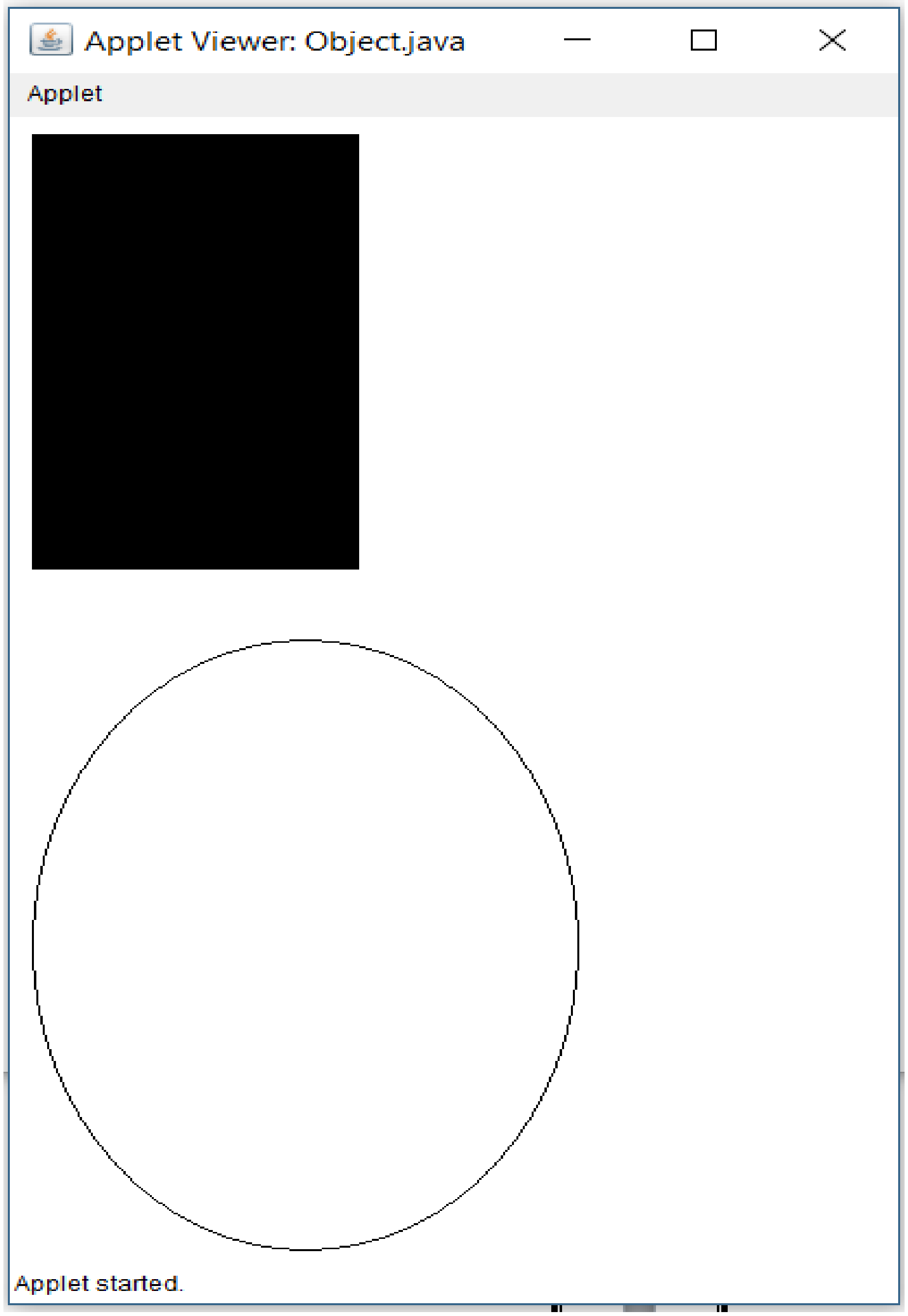
Programs for using Graphics class

Code:

```
import java.awt.*;
import java.applet.*;

/* <applet height=700 width=300 code=Object.java> </applet> */

public class Object extends Applet
{
        public void paint(Graphics g)
        {
                g.fillRect(10,10,150,250);
                g.drawOval(10,300,250,350);
        }
}
```

Output:

```
F:\java prog>javac Object.java

F:\java prog>appletviewer Object.java
```



Applet started.