VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

OBJECT ORIENTED JAVA PROGRAMMING

Submitted by

HARSHITHA R (1BM21CS075)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
Oct 2022-Feb 2023

B. M. S. College of Engineering,

Bull Temple Road, Bangalore 560019

(Affiliated To Visvesvaraya Technological University, Belgaum)

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled "**OBJECT ORIENTED JAVA PROGRAMMING**" carried out by **HARSHITHA R(1BM21CS075),** who is bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022-23. The Lab report has been approved as it satisfies the academic requirements in respect of Object Oriented Java Programming Lab - **(22CS3PCOOJ)** work prescribed for the said degree.

Professor Vikranth B.M
Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

TABLE OF CONTENTS

Course Outcome

CO1	Apply the knowledge of Java concepts to find the solution for a given problem.
CO2	Analyse the given Java application for correctness/functionalities.
CO3	Develop Java programs / applications for a given requirement.
CO4	Conduct practical experiments for demonstrating features of Java.

PROGRAM 1

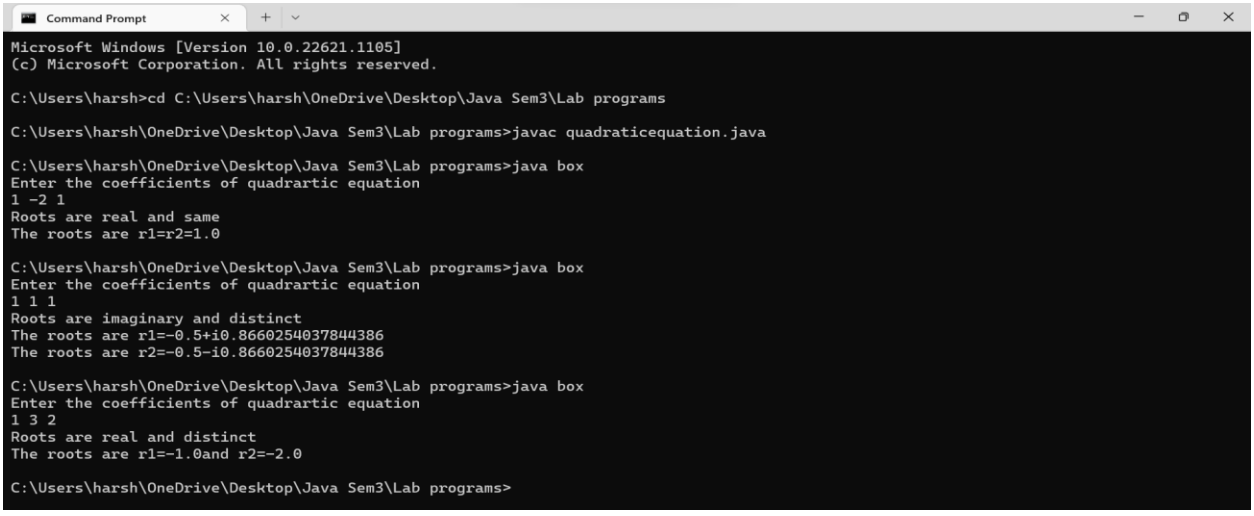
Develop a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c = 0$. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.

CODE:

```
import java.util.*;

import java.lang.*;

class box

{

 public static void main (String args[])

 {

  System.out.println("Enter the coefficients of quadrartic equation");

  Scanner sc= new Scanner(System.in);

  double a=sc.nextDouble();

  double b=sc.nextDouble();

  double c=sc.nextDouble();

  double d=(b*b)-(4*a*c);

  double r1;

  double r2;

  if(a==0)

  {

   System.out.println("Enter a valid value");

  }

  else

  {

  if(d>0)

  {

    r1=(-b+Math.sqrt(d))/(2*a);

    r2=(-b-Math.sqrt(d))/(2*a);
```

```
    System.out.println("Roots are real and distinct");
    System.out.println("The roots are r1="+r1+""+"and r2="+r2);
  }
 else if(d==0)
  {
    r1=r2=(-b)/(2*a);
    System.out.println("Roots are real and same");
    System.out.println("The roots are r1=r2="+""+r1);
  }
else
{
    r1=(-b)/(2*a);
    r2=(Math.sqrt(-d))/(2*a);
    System.out.println("Roots are imaginary and distinct");
    System.out.println("The roots are r1="+r1+"+i"+r2);
    System.out.println("The roots are r2="+r1+"-i"+r2);
}
}
}
```

OUTPUT:



PROGRAM 2

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

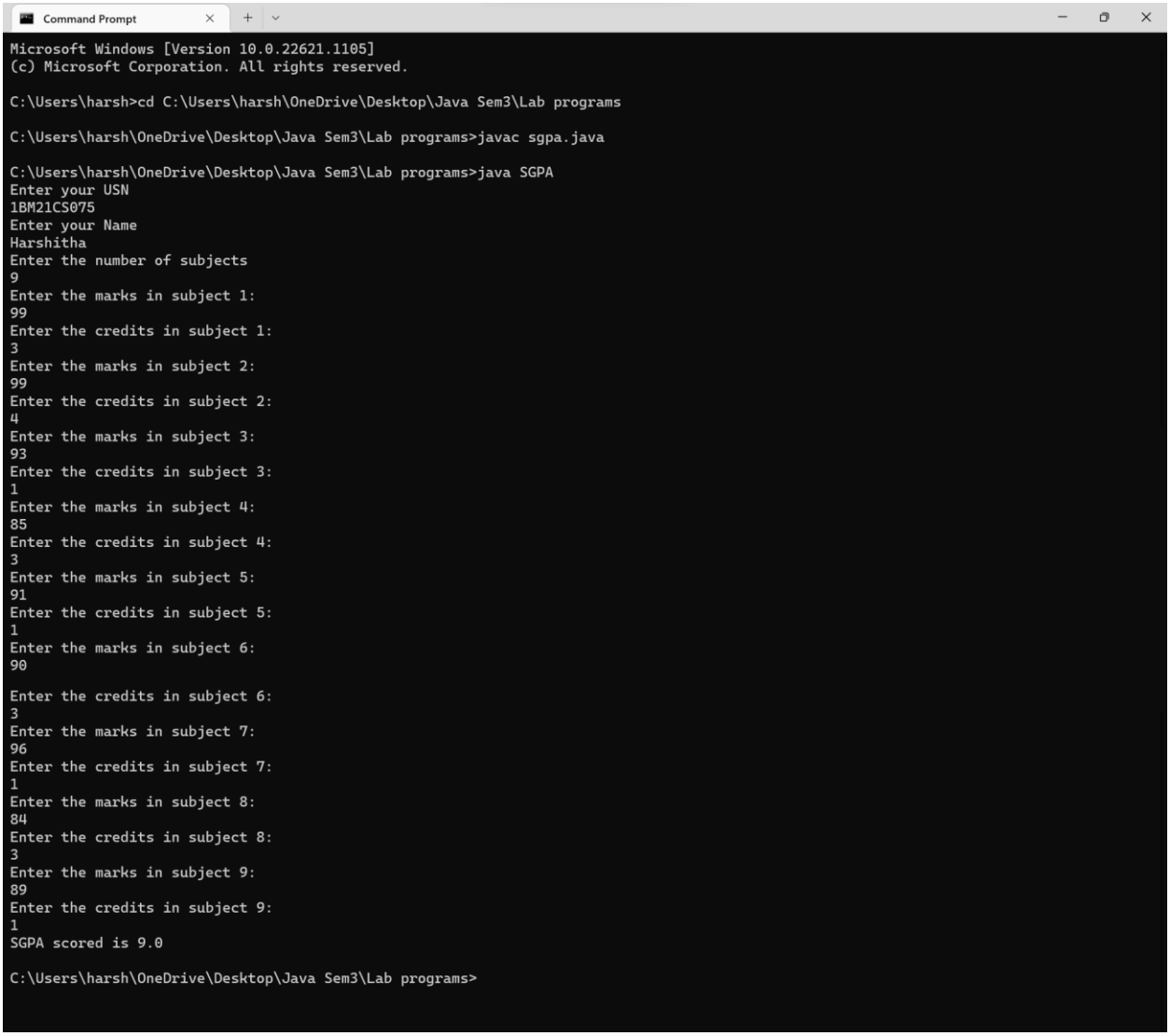
CODE:

```
import java.util.*;

class Student{

String usn,name;

Scanner sc = new Scanner(System.in);

Student(){

System.out.println("Enter your USN");

usn=sc.next();

System.out.println("Enter your Name");

name=sc.next();

}

int marks[]=new int[50];

int credits[]=new int[50];

int sum1=0;

int sum2=0;

void markenter(int x)

{

for (int i=0;i<x;i++){

System.out.println("Enter the marks in subject "+i+": ");

marks[i]=sc.nextInt();

System.out.println("Enter the credits in subject "+i+": ");

credits[i]=sc.nextInt();

sum1+=credits[i];

if (marks[i]>=90){

marks[i]=10;
```

```
}
else if (marks[i]>=80 && marks[i]<90)
{
marks[i]=9;
}
else if (marks[i]>=70 && marks[i]<80)
{
marks[i]=8;
}
else if (marks[i]>=60 && marks[i]<70)
{
marks[i]=7;
}
else if(marks[i]>=50 && marks[i]<60)
{
marks[i]=6;
}
else if(marks[i]>=40 && marks[i]<50)
{
marks[i]=5;
}
else
{
marks[i]=0;
}
}
}
```

```
double calc(int z)
{
for (int f=0;f<z;f++){
sum2+=marks[f]*credits[f];


}
return (sum2/sum1);
}
}

class SGPA{
public static void main(String args[]){
Student stud=new Student();
Scanner sc= new Scanner(System.in);
System.out.println("Enter the number of subjects");
 int n=sc.nextInt();
stud.markenter(n);
System.out.println("SGPA scored is "+ stud.calc(n) );
}
}
```

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.22621.1105]
(c) Microsoft Corporation. All rights reserved.
C:\Users\harsh>cd C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>javac sgpa.java
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>java SGPA
Enter your USN
1BM21CS075
Enter your Name
Harshitha
Enter the number of subjects
9
Enter the marks in subject 1:
99
Enter the credits in subject 1:
3
Enter the marks in subject 2:
99
Enter the credits in subject 2:
4
Enter the marks in subject 3:
93
Enter the credits in subject 3:
1
Enter the marks in subject 4:
85
Enter the credits in subject 4:
3
Enter the marks in subject 5:
91
Enter the credits in subject 5:
1
Enter the marks in subject 6:
90
Enter the credits in subject 6:
3
Enter the marks in subject 7:
96
Enter the credits in subject 7:
1
Enter the marks in subject 8:
84
Enter the credits in subject 8:
3
Enter the marks in subject 9:
89
Enter the credits in subject 9:
1
SGPA scored is 9.0
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>

PROGRAM 3

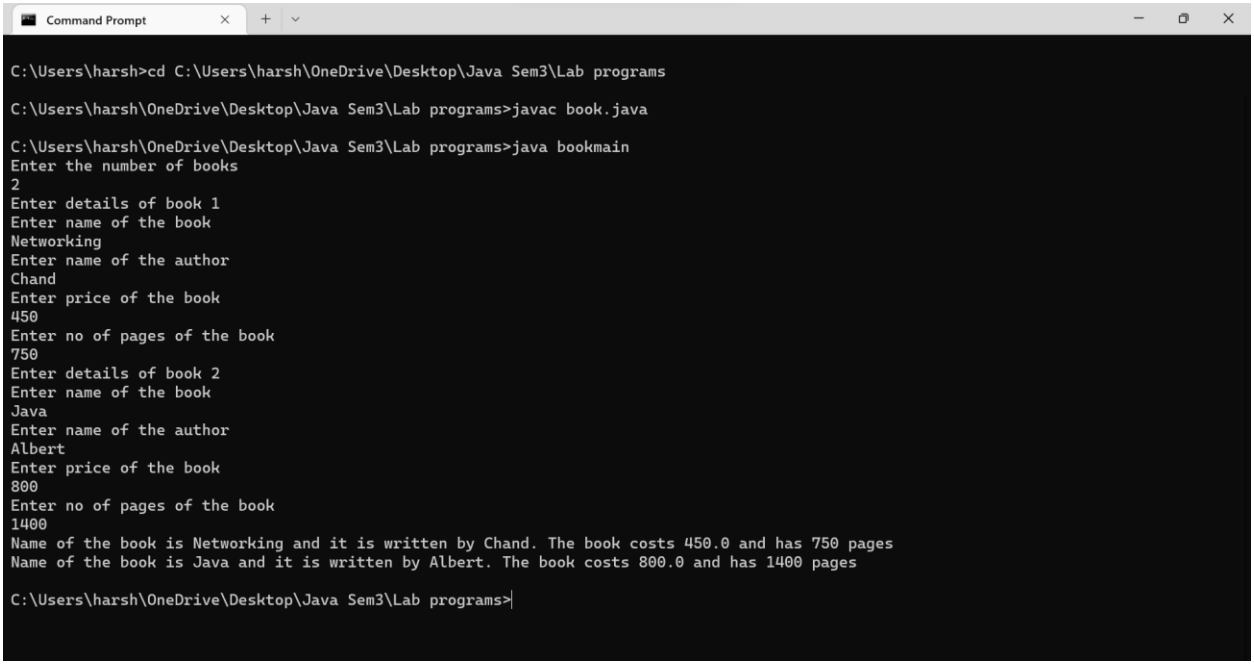
Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

CODE:

```
import java.util.*;

class book{

String name;

String author;

double price;

int pages;

Scanner sc= new Scanner(System.in);

book(){

name="";

author="";

price=0.0;

pages=0;

}

void set(){

System.out.println("Enter name of the book");

name=sc.next();

System.out.println("Enter name of the author");

author=sc.next();

System.out.println("Enter price of the book");

price=sc.nextDouble();

System.out.println("Enter no of pages of the book");

pages=sc.nextInt();

}
```

```
public String toString()
{
String s="Name of the book is "+name+" and it is written by "+author+". The book costs "+price+" and has "+ pages+" pages";
return s;
}
}

class bookmain{
public static void main(String args[]){
Scanner sc= new Scanner(System.in);
System.out.println("Enter the number of books");
int n;
n=sc.nextInt();
book b[]=new book[n]; //this is how you create an array of class
for(int i=0;i<n;i++){
System.out.println("Enter details of book "+(i+1));
b[i]=new book();
b[i].set();
}
for(int i=0;i<n;i++){
System.out.println(b[i]);
}
}
}
```

OUTPUT:


Command Prompt
C:\Users\harsh>cd C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>javac book.java
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>java bookmain
Enter the number of books
2
Enter details of book 1
Enter name of the book
Networking
Enter name of the author
Chand
Enter price of the book
450
Enter no of pages of the book
750
Enter details of book 2
Enter name of the book
Java
Enter name of the author
Albert
Enter price of the book
800
Enter no of pages of the book
1400
Name of the book is Networking and it is written by Chand. The book costs 450.0 and has 750 pages
Name of the book is Java and it is written by Albert. The book costs 800.0 and has 1400 pages
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>

PROGRAM 4

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

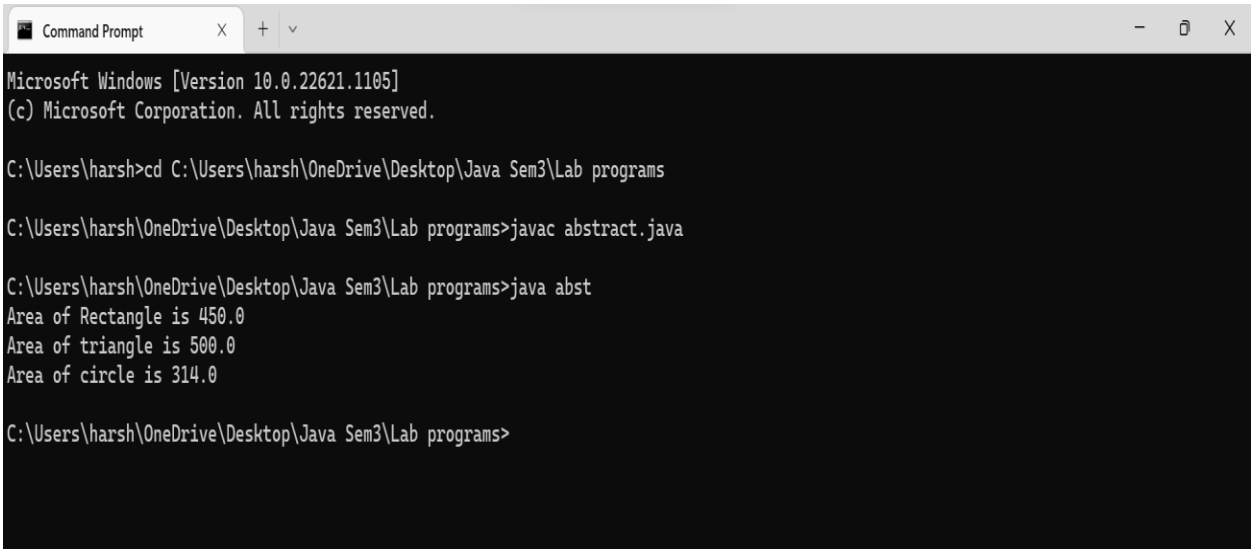
CODE:

```
abstract class Shape{
  int dim1,dim2;
  Shape(int x,int y)
   {
       dim1=x;
       dim2=y;
    }
 Shape(int z)
  {
    dim1=z;
  }
abstract double printarea();
}

class Rectangle extends Shape{
      Rectangle(int x,int y)
     {
        super(x,y);
      }
       double printarea()
    {
       return  dim1*dim2;
     }}
```

```
class triangle extends Shape{
        triangle(int x,int y)
        {
            super(x,y);
         }
        double printarea()
      {
        return  0.5*(dim1*dim2);
       }
  }

class circle extends Shape{
        circle(int z)
        {
            super(z);
         }
        double printarea()
      {
        return  3.14*dim1*dim1;
       }
  }

 class abst{
   public static void main(String args[])
    {
        Rectangle r1=new Rectangle(15,30);
        triangle t1=new triangle(20,50);
        circle c1=new circle(10);
```

```
        Shape f;
        f=r1;
        double a1=f.printarea();
        System.out.println("Area of Rectangle is"+" "+a1);
        f=t1;
        double a2=f.printarea();
        System.out.println("Area of triangle is"+" "+a2);
        f=c1;
        double a3=f.printarea();
        System.out.println("Area of circle is"+" "+a3);
    }
}
```

OUTPUT:



PROGRAM 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from customer and update the balance.

b) Display the balance.

c) Compute and deposit interest

d) Permit withdrawal and update the balance

Check for the minimum balance, impose penalty if necessary and update the balance.

CODE:

```
import java.util.Scanner;
class Account
{
        String name;
        int type;
        long accno;
        double balance;
        void setA()
        {
                Scanner s=new Scanner(System.in);
                System.out.print("Enter customer name: ");
                name=s.nextLine();

                System.out.print("Enter account number: ");
                accno=s.nextLong();
```

```
            System.out.print("Enter bank balance: ");
            balance=s.nextDouble();
        }
        void display()
        {
            System.out.println("Customer name is: "+name);
            if(type==1) {
                System.out.println("Customer account type is: Savings");
            }
            else {
                System.out.println("Customer account type is: Current");
            }
            System.out.println("Customer account number is: "+accno);
            System.out.println("Current balance is: "+balance);
        }
        void deposit()
        {
            System.out.print("Enter the amount to be deposited: ");
            Scanner x=new Scanner(System.in);
            double amt=x.nextDouble();
            balance+=amt;
        }
}

class Sav_acct extends Account
{
        double interest;
        Scanner s=new Scanner(System.in);
```

```
Sav_acct() {
        type=1;
}
void cinterest()
{
        int timey;
        float irate;
        System.out.println("Compound Interest details:");

        System.out.println("Enter time in years: ");
        timey=s.nextInt();
        System.out.println("Enter rate of interest: ");
        irate=s.nextFloat();
        System.out.println("Interest will be compunded 5 times a year");
        interest=balance*(Math.pow((1+irate/5),(5*timey)));
        balance+=interest;
}
void withdraw()
{
        System.out.println("Enter the amount to be withdrawn: ");
        double amt=s.nextDouble();
        if(balance>amt)
        {balance-=amt;}
        else
        {
           System.out.println("Amount to be withdrawn greater than balance!!!");
         }
```

```
        }
}

class Curr_acct extends Account
{
        double check_amt;

        Curr_acct() {
                type=2;
        }

        void cheque()
        {
                System.out.print("Enter the cheque amount: ");
                Scanner s=new Scanner(System.in);
                check_amt = s.nextDouble();
                if(check_amt>balance-5000)
                {
                    System.out.println("Rs. 500 penalty imposed...Is it ok to proceed? Enter y for
yes and n for no”);
                        String option=s.next();
                        if(option.equals("y")) {balance=balance-check_amt-500;}
                        else {System.out.println("no check debited");}
                }
                else
                {
                        System.out.println("Rupees "+check_amt+" debited"); balance-
=check_amt;
                }
```

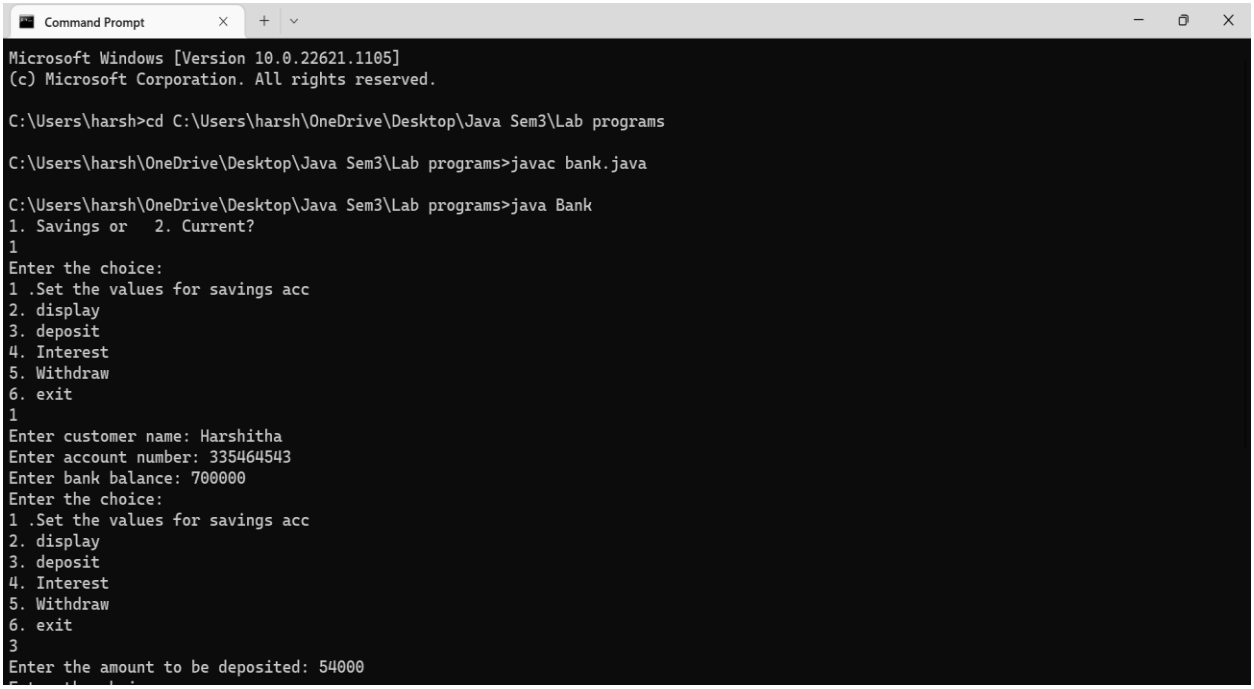
```
	}
	void withdraw()
	{
		System.out.println("Enter the amount to be withdrawn: "); Scanner s=new
Scanner(System.in);
		double amt=s.nextDouble();
		if(balance>amt)
		{balance-=amt;}
		else
		{System.out.println("Amount to be withdrawn greater than balance!!!");}
	}
}

class Bank {
	public static void main(String ss[]) {
		String op1,op2;
		Scanner s=new Scanner(System.in);
		System.out.println("1. Savings or   2. Current?");
		int q;
		q=s.nextInt();
		if(q==1) {
			Sav_acct s1 = new Sav_acct();
			while(true) {
			System.out.print("Enter the choice: \n1 .Set the values for savings acc\n2.
display\n3. deposit\n4. Interest\n5. Withdraw\n6. exit\n");
			op1=s.next();
			switch(op1)
			{
			case "1":s1.setA();
```

```
                    break;
                case "2":s1.display();
                    break;
                case "3":s1.deposit();
                    break;
                case "4":s1.cinterest();
                    break;
                case "5":s1.withdraw();
                    break;
                case "6":System.exit(0);
                }
                }
            }
            else if(q==2) {
                Curr_acct c1 = new Curr_acct();
                while(true) {
                System.out.print("Enter the choice: \n1.Set the values for current
account\n2. display\n3. deposit\n4. transferCheck\n5. Withdraw\n6. exit\n");
                op2=s.next();
                switch(op2)
                {
                case "1":c1.setA();
                    break;
                case "2":c1.display();
                    break;
                case "3":c1.deposit();
                    break;
                case "4":c1.cheque();
                    break;
```

```
                    case "5":c1.withdraw();
                            break;
                    case "6":System.exit(0);
                    }
                    }
                }
        }
}
```

OUTPUT:

1.Savings Account:



```
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
5
Enter the amount to be withdrawn:
10000
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
2
Customer name is: Harshitha
Customer account type is: Savings
Customer account number is: 335464543
Current balance is: 744000.0
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
6

C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>
```

2.Current Account

```
Command Prompt

C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>java Bank
1. Savings or   2. Current?
2
Enter the choice:
1.Set the values for current account
2. display
3. deposit
4. transferCheck
5. Withdraw
6. exit
1
Enter customer name: Harshitha
Enter account number: 6465544
Enter bank balance: 500000
Enter the choice:
1.Set the values for current account
2. display
3. deposit
4. transferCheck
5. Withdraw
6. exit
3
Enter the amount to be deposited: 14000
Enter the choice:
1.Set the values for current account
2. display
3. deposit
4. transferCheck
5. Withdraw
6. exit
5
Enter the amount to be withdrawn:
5000
```

```
Enter the choice:
1.Set the values for current account
2. display
3. deposit
4. transferCheck
5. Withdraw
6. exit
4
Enter the cheque amount: 8000
Rupees 8000.0 debited
Enter the choice:
1.Set the values for current account
2. display
3. deposit
4. transferCheck
5. Withdraw
6. exit
2
Customer name is: Harshitha
Customer account type is: Current
Customer account number is: 6465544
Current balance is: 501000.0
Enter the choice:
1.Set the values for current account
2. display
3. deposit
4. transferCheck
5. Withdraw
6. exit
6

C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>
```

PROGRAM 6

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.

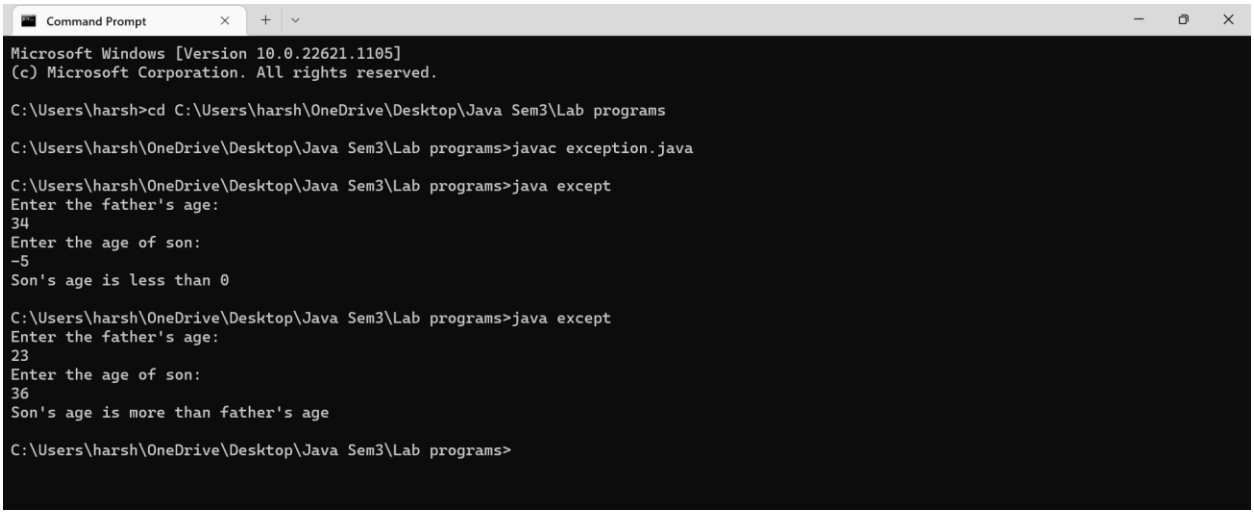
CODE:

```
import java.util.*;

class fatherAgeException extends Exception

{

  public String toString(){

      return("Father's age is less that 0");

  }

}


class sonAgeException extends Exception{

    int a;

    sonAgeException(int age){

        a=age;

    }

  public String toString(){

        if(a<0)

          return("Son's age is less than 0");

        else

          return("Son's age is more than father's age");

    }

}
```

```
class Father{
    int age;
    Scanner in=new Scanner(System.in);
    Father(){
        System.out.println("Enter the father's age: ");
        age=in.nextInt();
    }
    void ex1() throws fatherAgeException
    {
        if(age<0)
         throw new fatherAgeException();
    }
}

class Son extends Father{
    int age;
  Son(){
    System.out.println("Enter the age of son: ");
    age=in.nextInt();
  }
  void ex2() throws sonAgeException{
      if(age<0||age>super.age){
          throw new sonAgeException(age);
      }
  }
}
```

```
public class except {
    public static void main(String[] args){
        Son s=new Son();
        try{
            s.ex1();
        }
        catch(fatherAgeException e){
            System.out.println(e);
        }
        try{
            s.ex2();
        }
        catch(sonAgeException e){
            System.out.println(e);
        }
    }
}
```

OUTPUT:



PROGRAM 7

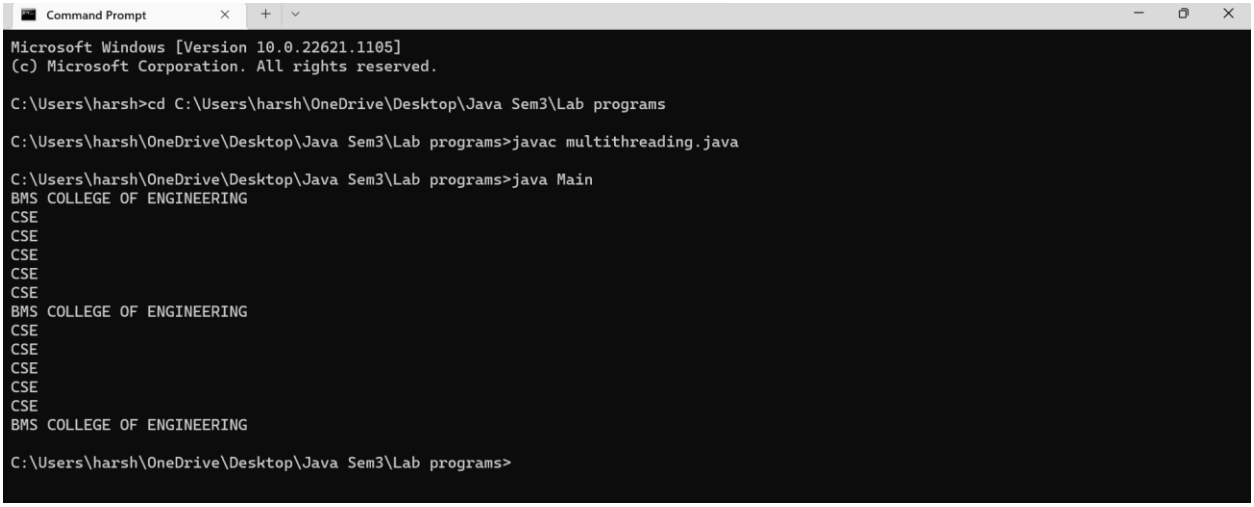
Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

CODE:

```
class MyThread extends Thread
{
long time;
private volatile boolean running = true;
MyThread(){
System.out.println("Default");
}
MyThread(String name, long time)
{
super(name);
this.time = time;
}
public void pause()
{
running = false;
}
public void run()
{
try
{
while(running)
{
System.out.println(this.getName());
Thread.sleep(time*1000);
}
```

```
}
 catch(InterruptedException e)
{
System.out.println("Exception caught in method");
}

}
}
class Main
{
public static void main(String args[ ])
{
MyThread t1 = new MyThread("BMS COLLEGE OF ENGINEERING", 10);
MyThread t2 = new MyThread("CSE", 2);
t1.start();
t2.start();
try
{
Thread.sleep(20*1000);
t1.pause();
t2.pause();
}
catch(InterruptedException e)
{
System.out.println("Exception caught in main");
}
}
}
```

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.22621.1105]
(c) Microsoft Corporation. All rights reserved.
C:\Users\harsh>cd C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>javac multithreading.java
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>java Main
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
C:\Users\harsh\OneDrive\Desktop\Java Sem3\Lab programs>