DEPARTMENT OF COMPUTER SCIENCE AND ENGNEERING

B. M. S. COLLEGE OF ENGINEERING

(AUTONOMOUS COLLEGE UNDER VTU,BELGAUM)

BANGALORE – 560019

2023-24

LAB REPORT ON

OBJECT ORIENTED JAVA PROGRAMMING (23CSEPCOOJ)

Submitted by

Bhoomika Hegde

1BM22CS342

Submitted to

Shravya AR
Assistant Professor
Dept. of CSE, BMSCE

CERTIFICATE

This is to certify that the lab work of the course "**Object Oriented Programming in Java**" carried out by **BHOOMIKA HEGDE (1BM22CS342),**who is a bonafide student of **B. M. S. College of Engineering.** It is in partial fullfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visveswaraiah Technological University, Belgaum during the year 2024. The lab report has been approved as it satisfies the academic requirements in respect of **Object Oriented Programming in Java (23CS3PCOOJ)** work prescribed for the said degree.

Signature of the Guide

Shravya A R
Assistant Professor
CSE
BMSCE, Bengaluru

Signature of the HOD

Dr. Jyothy S Nayak
Prof & Head of Dept of CSE
BMSCE, Bengaluru

OOJ LAB PROGRAMS

Lab Program	Program Details
1	Develop a Java program that prints all real solutions to the quadratic equation ax2+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.
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.
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.
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.
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 Savacct 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.
6	Create a package CIE which has two classes- Student and Internals. The class Student has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.
7	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=father’s age.
8	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

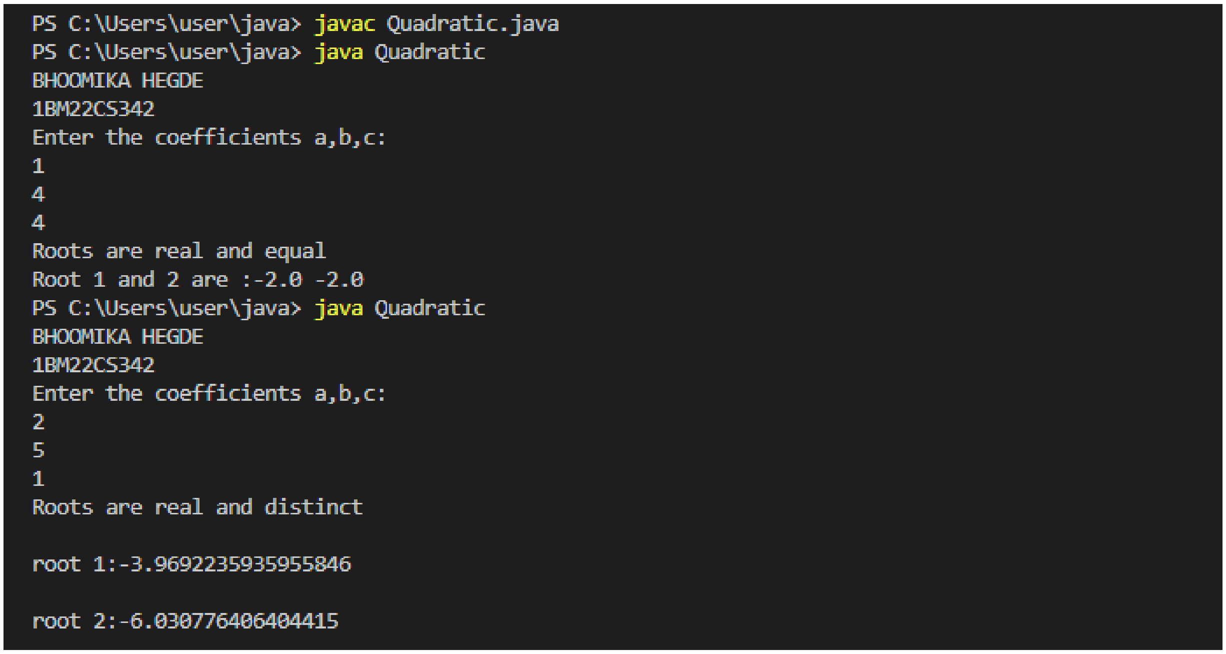
LAB PROGRAM-1

Develop a Java program that prints all real solutions to the quadratic equation ax2+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.math.*;
class Quad
{
    int a,b,c;
    double r1,r2,d;
    void input()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the coefficients a,b,c:");
        a=s.nextInt();
        b=s.nextInt();
        c=s.nextInt();
    }

    void compute()
    {
        if(a==0)
        {
            System.out.println("Invalid input");
        }
        else
        {
            d=(double)((b*b)-(4*a*c));
            if(d==0)
            {
                r1=r2=(-b/(2*a));
                System.out.println("Roots are real and equal");
                System.out.println("Root 1 and 2 are :" +r1 +" "+ r2);
            }
            else if(d>0)
            {
                r1=(-b+(Math.sqrt(d))/(double)(2*a));
                r2=(-b-(Math.sqrt(d))/(double)(2*a));
                System.out.println("Roots are real and distinct \n");
                System.out.println("root 1:"+r1+"\n");
                System.out.println("root 2:"+r2+"\n");
            }
            else
            {
                r1=(-b/(double)(2*a));
                System.out.println("Roots are imaginary\n");
                System.out.println("root 1="+ r1+ "+"+ Math.sqrt(-d)/(double)(2*a)+ "i \n");
                System.out.println("root 2="+ r1+ "-"+ Math.sqrt(-d)/(double)(2*a)+ "i\n");

            }
        }
    }
}

class Quadratic
{
    public static void main(String args[])
    {
        System.out.println("BHOOMIKA HEGDE \n1BM22CS342 ");
        Quad q=new Quad();
        q.input();
        q.compute();

    }
}
```

OUTPUT



```
PS C:\Users\user\java> javac Quadratic.java
PS C:\Users\user\java> java Quadratic
BHOOMIKA HEGDE
1BM22CS342
Enter the coefficients a,b,c:
1
4
4
Roots are real and equal
Root 1 and 2 are :-2.0 -2.0
PS C:\Users\user\java> java Quadratic
BHOOMIKA HEGDE
1BM22CS342
Enter the coefficients a,b,c:
2
5
1
Roots are real and distinct

root 1:-3.9692235935955846

root 2:-6.030776406404415
```

LAB PROGRAM-2

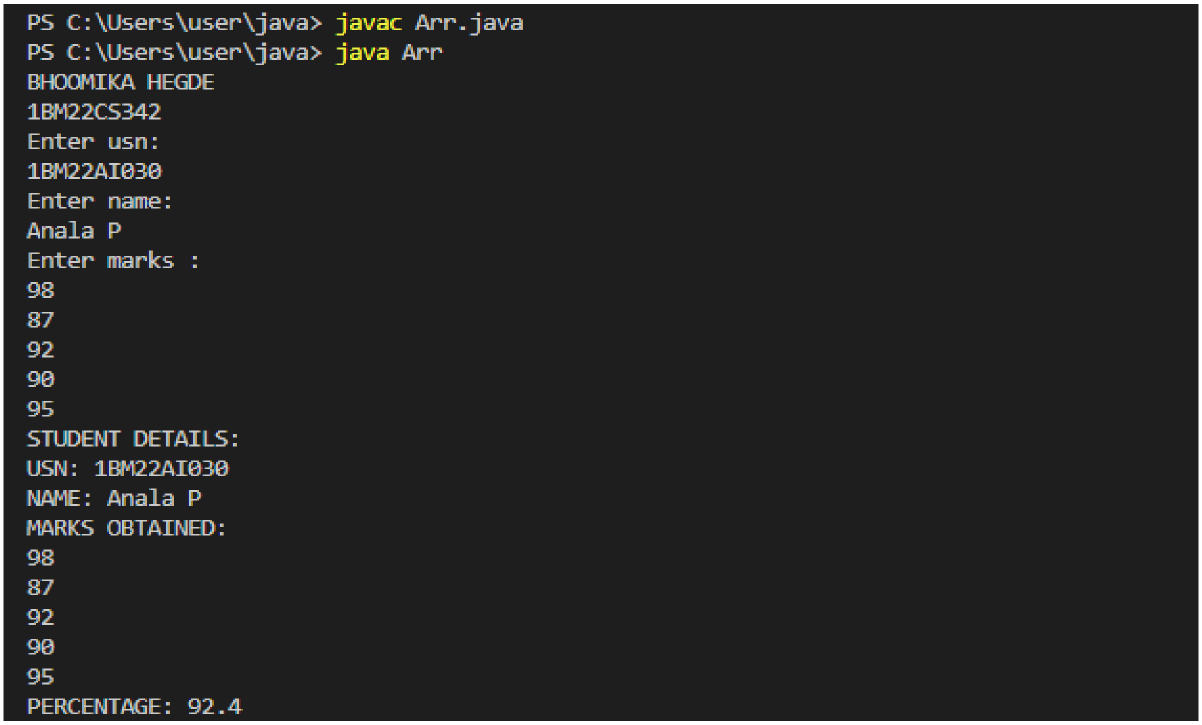
Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

CODE

```
import java.util.*;
class Student
{
   String usn;
   String name;
   int marks[]=new int[5];
   double perc;


   void input()
   {
      System.out.println("Enter usn: ");
      Scanner sc=new Scanner(System.in);
      usn=sc.nextLine();
      System.out.println("Enter name: ");
      name=sc.nextLine();
      System.out.println("Enter marks :");
      for(int i=0;i<5;i++)
      {
         marks[i]=sc.nextInt();
      }
   }

   public void percentage()
   {
      int total=0;
      for(int i=0;i<5;i++)
      {
         total=total+marks[i];
      }
      perc=(double)total/500*100;
   }

   void display()
   {
      System.out.println("STUDENT DETAILS:");
      System.out.println("USN: "+usn );
      System.out.println("NAME: "+name);
```

```
        System.out.println("MARKS OBTAINED: ");
        for(int i=0;i<5;i++)
        {
            System.out.println(marks[i]);
        }
        System.out.println("PERCENTAGE: " + perc);
    }
}

class Arr
{
    public static void main(String args[])
    {
        System.out.println("BHOOMIKA HEGDE \n1BM22CS342 ");
        Student s1=new Student();
        s1.input();
        s1.percentage();
        s1.display();
    }
}
```

OUTPUT



```
PS C:\Users\user\java> javac Arr.java
PS C:\Users\user\java> java Arr
BHOOMIKA HEGDE
1BM22CS342
Enter usn:
1BM22AI030
Enter name:
Anala P
Enter marks :
98
87
92
90
95
STUDENT DETAILS:
USN: 1BM22AI030
NAME: Anala P
MARKS OBTAINED:
98
87
92
90
95
PERCENTAGE: 92.4
```

LAB PROGRAM-3

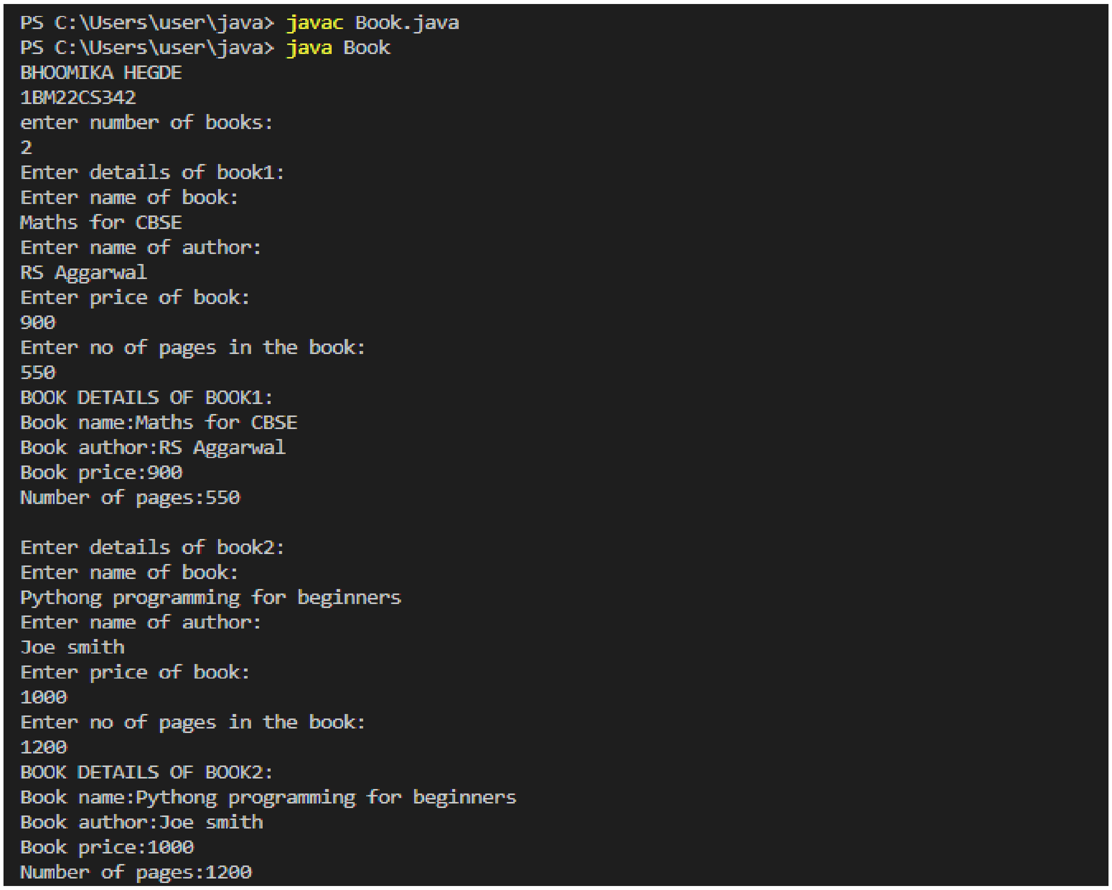
Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program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

CODE

```
import java.util.*;
class Book1
{
   String name;
   String author;
   int price;
   int pages;
   Book1()
   {
      this.name=name;
      this.author=author;
      this.price=price;
      this.pages=pages;
   }
   void getDetails()
   {
      Scanner s=new Scanner(System.in);
      System.out.println("Enter name of book: ");
      name=s.nextLine();
      System.out.println("Enter name of author: ");
      author=s.nextLine();
      System.out.println("Enter price of book: ");
      price=s.nextInt();
      System.out.println("Enter no of pages in the book: ");
      pages=s.nextInt();
   }
   public String toString()
   {
      String a,b,c,d;
      a="Book name:"+this.name+"\n";
      b="Book author:"+this.author+"\n";
      c="Book price:"+this.price+"\n";
      d="Number of pages:"+this.pages+"\n";
      return a,b,c,d;
   }
}
```

```
class Book
{
   public static void main(String args[])
   {
      System.out.println("1BM22CS342: BHOOMIKA HEGDE");
      int no_of_books;
      System.out.println("enter number of books: ");
      Scanner sc=new Scanner(System.in);
      no_of_books=sc.nextInt();
      Book1[] b=new Book1[no_of_books];
      for(int i=0;i<no_of_books;i++)
      {
         b[i]=new Book1();
         b[i].getDetails();
         System.out.println(b[i]);
      }

   }
}
```

OUTPUT



```
PS C:\Users\user\java> javac Book.java
PS C:\Users\user\java> java Book
BHOOMIKA HEGDE
1BM22CS342
enter number of books:
2
Enter details of book1:
Enter name of book:
Maths for CBSE
Enter name of author:
RS Aggarwal
Enter price of book:
900
Enter no of pages in the book:
550
BOOK DETAILS OF BOOK1:
Book name:Maths for CBSE
Book author:RS Aggarwal
Book price:900
Number of pages:550

Enter details of book2:
Enter name of book:
Pythong programming for beginners
Enter name of author:
Joe smith
Enter price of book:
1000
Enter no of pages in the book:
1200
BOOK DETAILS OF BOOK2:
Book name:Pythong programming for beginners
Book author:Joe smith
Book price:1000
Number of pages:1200
```

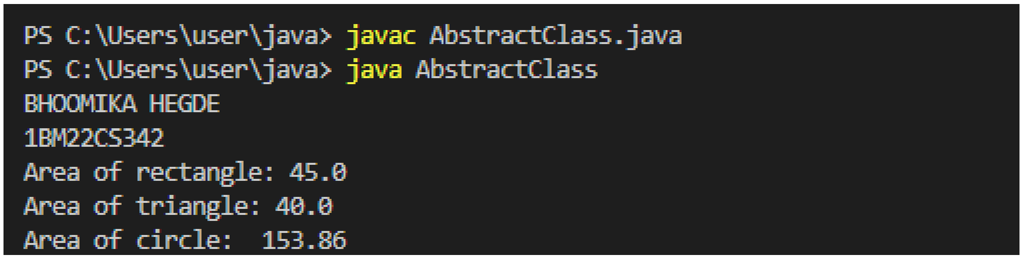
LAB PROGRAM-4

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 Figure
{
   double dim1;
   double dim2;

   Figure(double a, double b)
   {
      dim1 = a;
      dim2 = b;
   }

   abstract double area();
}

class Rectangle extends Figure
{
   Rectangle(double a, double b)
   {
      super(a, b);
   }

   double area()
   {
      return dim1 * dim2;
   }
   }

class Triangle extends Figure
{
   Triangle(double a, double b)
   {
      super(a, b);
   }

   double area()
   {
        return dim1 * dim2 / 2;
    }
}
class Circle extends Figure
{
    Circle(double a,double b)
    {
        super(a,b);
    }
    double area()
    {
        return 3.14*dim1*dim1;
    }
}

class AbstractClass
{
    public static void main(String args[])
    {
        System.out.println("BHOOMIKA HEGDE \n1BM22CS342 ");
        Rectangle r = new Rectangle(9, 5);
        Triangle t = new Triangle(10, 8);
        Circle c=new Circle(7,0);
        System.out.println("Area of rectangle: " + r.area());
        System.out.println("Area of triangle: " + t.area());
        System.out.println("Area of circle:  " + c.area());
    }
}
```

OUTPUT



```
PS C:\Users\user\java> javac AbstractClass.java
PS C:\Users\user\java> java AbstractClass
BHOOMIKA HEGDE
1BM22CS342
Area of rectangle: 45.0
Area of triangle: 40.0
Area of circle:  153.86
```

LAB 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 Savacct 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
{
   public static  int min=500;
   String name;
   int Account_num;
   public float o_Price;
   Scanner sc=new Scanner(System.in);
   public void get_info()
   {
      System.out.println("Enter Name:");
      name=sc.nextLine();
      System.out.println("Enter Account Number:");
      Account_num=sc.nextInt();
      System.out.println("Enter opening Ammount ,must be >500:");
      o_Price=sc.nextFloat();
      if(o_Price <500)
      {
         System.out.println("Enter opening Ammount,must be >500:");
      }
   }
   public void show()
   {
      System.out.println("Name:"+name);
      System.out.println("Account_number:"+Account_num);
      System.out.println("Ammount:"+o_Price);
   }
}
class  Current  extends Account
{
   float deposit,withdraw,penality;
   public void deposit()
```

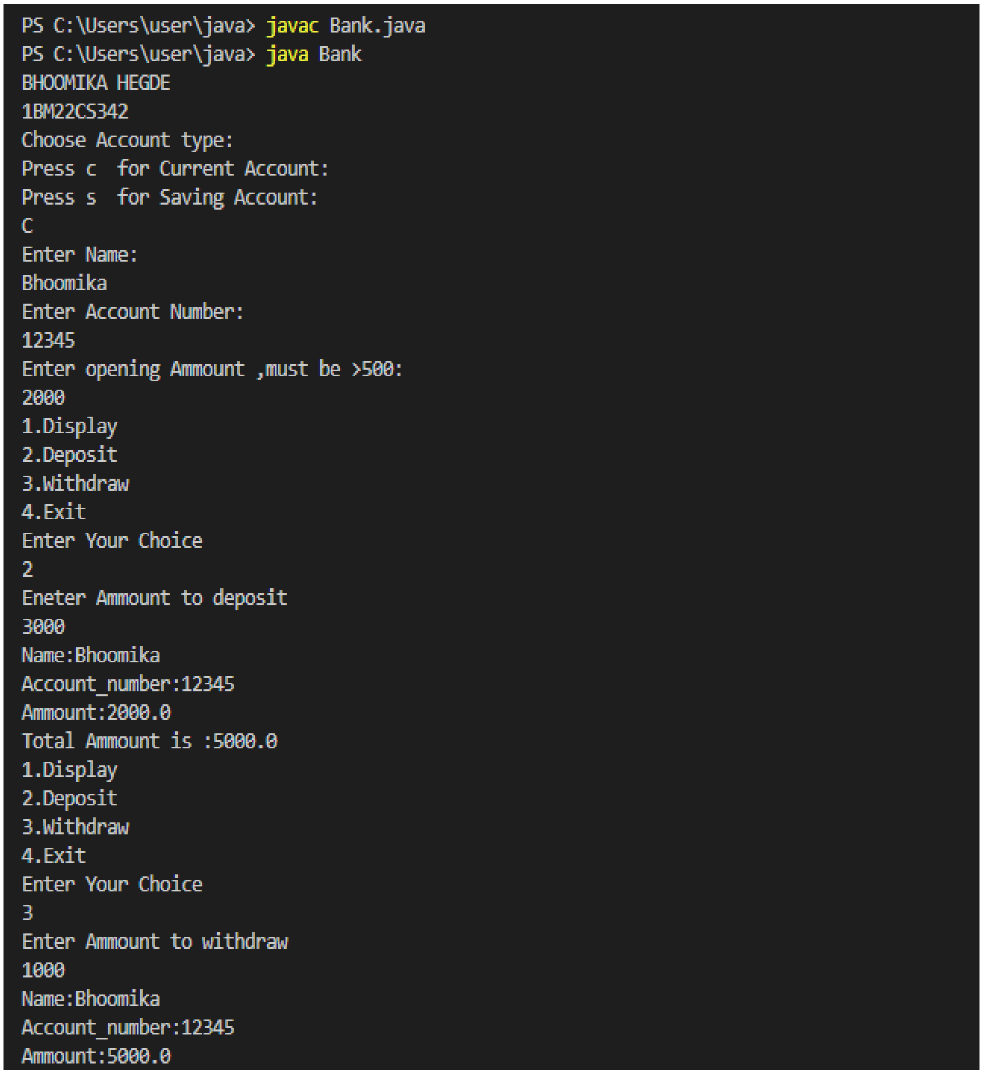
```
    {
        System.out.println("Eneter Ammount to deposit") ;
        deposit =sc.nextFloat();
        show();
        o_Price=o_Price+deposit;
        System.out.println("Total Ammount is :"+o_Price) ; }
        public void check_Bal()
        {
            if(o_Price<min)
            {
                System.out.println("Ammont Should be >500");
                o_Price=o_Price-150;
                System.out.println("You have debited ammount 150 from your acccount     Account balance
is:"+o_Price);
            }

        }
    public void withdraw_Bal()
    {
        System.out.println("Enter Ammount to withdraw");
        withdraw=sc.nextFloat();
        show();
        if(withdraw<o_Price)
        {
            o_Price=o_Price-withdraw;
            System.out.println("After Withdawl Balance "+o_Price);
        }
        else
        {
            System.out.println("Insufficent Balance cant not be less than 500");
        }
        check_Bal();

    }
}
class Saving extends Account
{

    float deposit,withdraw,intr;
    public void deposit()
    {
        System.out.println("Enter Ammount to deposit") ;
        deposit =sc.nextFloat();
        show();
        o_Price=o_Price+deposit;
```

```
      System.out.println("Total Ammount is :"+o_Price) ;
   }

   public void check_intrest()
   {
      intr=(o_Price*2)/100;
      o_Price=o_Price+intr;
      System.out.println("Total Ammount with interest is :"+o_Price) ;
   }
   public void withdraw_Bal()
   {
      System.out.println("Enter Ammount to withdraw:");
      withdraw=sc.nextFloat();
      show();
      if(withdraw<o_Price)
      {
         o_Price=o_Price-withdraw;
         System.out.println("After Withdawl Balance: "+o_Price);
      }
      else
      {
         System.out.println("Insufficent Balance!");
      }
   }
}

public class Bank
{
   static String ch;
   public static void main(String[] args)
   {
      System.out.println("BHOOMIKA HEGDE \n1BM22CS342");
      int count=0;
      Scanner sc=new Scanner(System.in);
      Current  cu=new Current ();
      Saving  sav=new Saving ();
      System.out.println("Choose Account type:");
      System.out.println("Press c  for Current Account:");
      System.out.println("Press s  for Saving Account:");
      ch=sc.nextLine();
      if(ch.equalsIgnoreCase("c"))
      {
         cu.get_info();
         cu.check_Bal();
         while(count!=4)
```

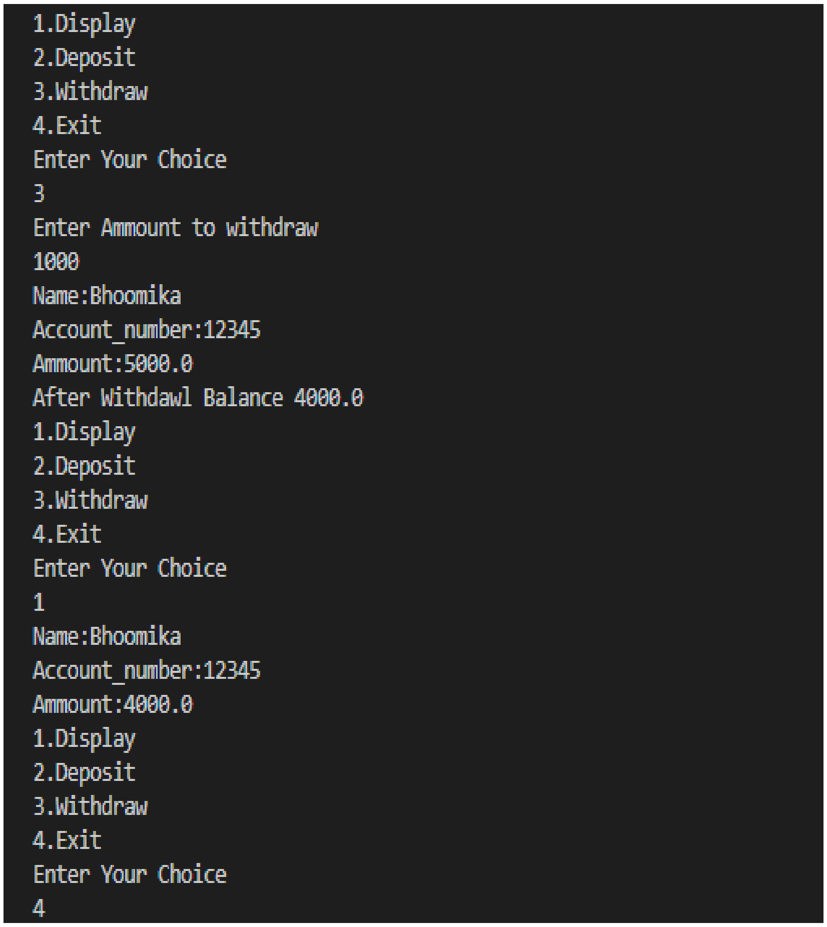
```
    {

        System.out.println("1.Display\n2.Deposit\n3.Withdraw\n4.Exit");
        System.out.println("Enter Your Choice");
        int cho=sc.nextInt();
        switch(cho)
        {
        case 1: cu.show();
        break;
        case 2: cu.deposit();
        break;
        case 3: cu.withdraw_Bal();
        break;
        case 4: System.exit(0);
        break;
        default:System.out.println("Wrong Choice!");
        }
    }
}
else if(ch.equalsIgnoreCase("s"))
{
    sav.get_info();

    while(count!=5)
    {
    System.out.println("1.Display\n2.Deposit\n3.Withdraw\n4Intrest\n5.Exit");
    System.out.println("Enter Your Choice");
    int cho=sc.nextInt();
    switch(cho)
    {
    case 1: sav.show();
    break;
    case 2: sav.deposit();
    break;
    case 3: sav.withdraw_Bal();
    break;
    case 4: sav.check_intrest();
    break;
    case 5: System.exit(0);
    break;
    default:System.out.println("Wrong Choice!");
    }
    }
}
else
```

```
        {
           System.out.println("Wrong choice!");
        }
    }
}
```

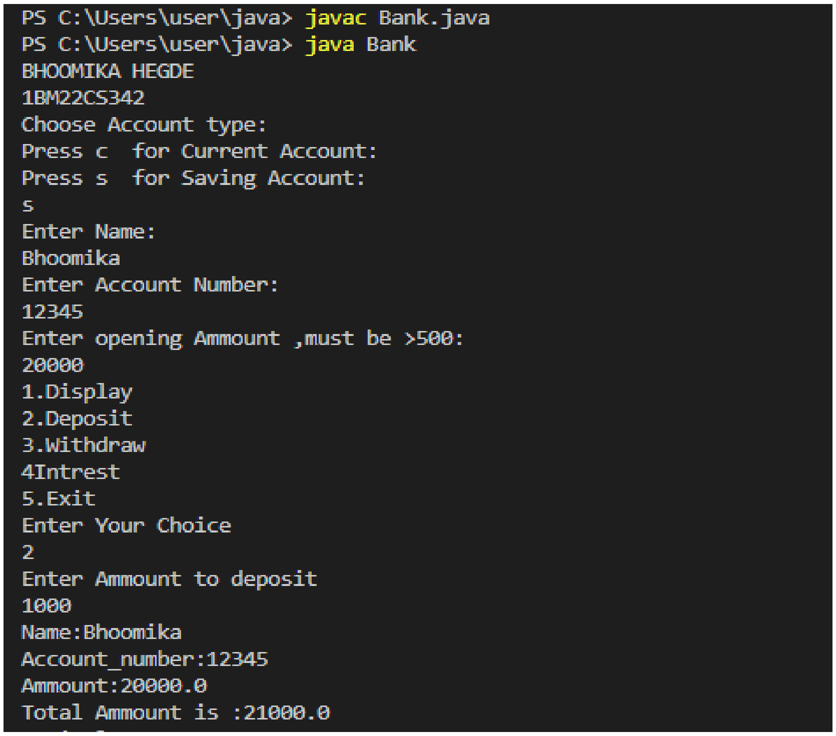
OUTPUT



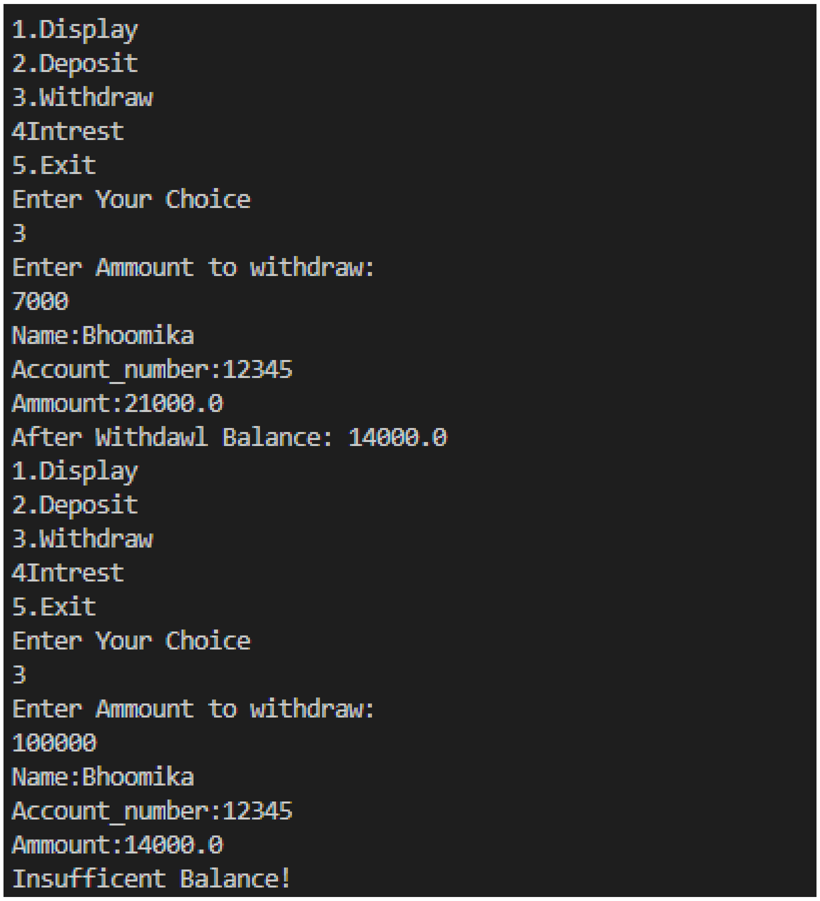
```
PS C:\Users\user\java> javac Bank.java
PS C:\Users\user\java> java Bank
BHOOMIKA HEGDE
1BM22CS342
Choose Account type:
Press c  for Current Account:
Press s  for Saving Account:
C
Enter Name:
Bhoomika
Enter Account Number:
12345
Enter opening Ammount ,must be >500:
2000
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Choice
2
Eneter Ammount to deposit
3000
Name:Bhoomika
Account_number:12345
Ammount:2000.0
Total Ammount is :5000.0
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Choice
3
Enter Ammount to withdraw
1000
Name:Bhoomika
Account_number:12345
Ammount:5000.0
```



```
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Choice
3
Enter Ammount to withdraw
1000
Name:Bhoomika
Account_number:12345
Ammount:5000.0
After Withdawl Balance 4000.0
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Choice
1
Name:Bhoomika
Account_number:12345
Ammount:4000.0
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Choice
4
```



```
PS C:\Users\user\java> javac Bank.java
PS C:\Users\user\java> java Bank
BHOOMIKA HEGDE
1BM22CS342
Choose Account type:
Press c  for Current Account:
Press s  for Saving Account:
s
Enter Name:
Bhoomika
Enter Account Number:
12345
Enter opening Ammount ,must be >500:
20000
1.Display
2.Deposit
3.Withdraw
4Intrest
5.Exit
Enter Your Choice
2
Enter Ammount to deposit
1000
Name:Bhoomika
Account_number:12345
Ammount:20000.0
Total Ammount is :21000.0
```



```
1.Display
2.Deposit
3.Withdraw
4Intrest
5.Exit
Enter Your Choice
3
Enter Ammount to withdraw:
7000
Name:Bhoomika
Account_number:12345
Ammount:21000.0
After Withdawl Balance: 14000.0
1.Display
2.Deposit
3.Withdraw
4Intrest
5.Exit
Enter Your Choice
3
Enter Ammount to withdraw:
100000
Name:Bhoomika
Account_number:12345
Ammount:14000.0
Insufficent Balance!
```

LAB PROGRAM-6

Create a package CIE which has two classes- Student and Internals. The class Student has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses

CODE

Internals.java

```
package CIE;

public class Internals extends Student
{
   public int[] internalMarks = new int[5];

   public Internals(String usn, String name, int sem, int[] marks) {
      super(usn, name, sem);
      this.internalMarks = marks;
   }
}
```

Student.java

```
package CIE;

public class Student
{
   public String usn;
   public String name;
   public int sem;

   public Student(String usn, String name, int sem) {
      this.usn = usn;
      this.name = name;
      this.sem = sem;
   }
}
```

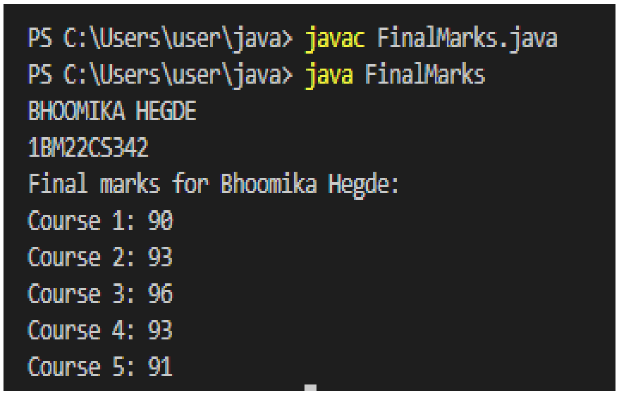
Externals.java

```
package SEE;
import CIE.Student;

public class External extends Student
{
   public int[] externalMarks = new int[5];

   public External(String usn, String name, int sem, int[] marks) {
      super(usn, name, sem);
      this.externalMarks = marks;
   }
}
```

FinalMarks.java

```
import CIE.Internals;
import SEE.External;

public class FinalMarks
{
   public static void main(String[] args)
   {
      System.out.println("BHOOMIKA HEGDE \n1BM22CS342 ");
      int[] internalMarks = {45,46,47,43,49};
      int[] externalMarks = {45,47,49,50,42};

      Internals student1Internal = new Internals("1BM22AI030", "Bhoomika Hegde", 6, internalMarks);
      External student1External = new External("1BM22AI030", "Bhoomika Hegde", 6, externalMarks);

      System.out.println("Final marks for " + student1Internal.name + ":");
      for (int i = 0; i < 5; i++)
      {
         int finalMark = student1Internal.internalMarks[i] + student1External.externalMarks[i];
         System.out.println("Course " + (i + 1) + ": " + finalMark);
      }
   }
}
```

OUTPUT



```
PS C:\Users\user\java> javac FinalMarks.java
PS C:\Users\user\java> java FinalMarks
BHOOMIKA HEGDE
1BM22CS342
Final marks for Bhoomika Hegde:
Course 1: 90
Course 2: 93
Course 3: 96
Course 4: 93
Course 5: 91
```

LAB PROGRAM-7

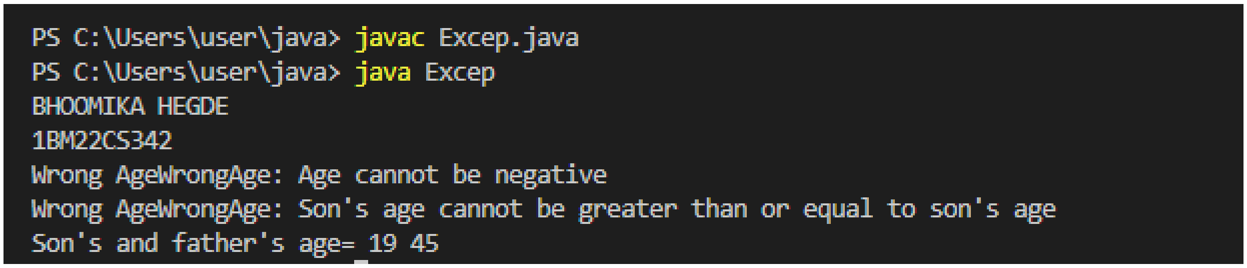
Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age=father’s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

CODE

```
class WrongAge extends Exception
{
    public WrongAge(String message)
    {
        super(message);
    }
}

class Father
{
    int fatherAge;
    public Father(int fatherAge) throws WrongAge
    {
        if(fatherAge<0)
        {
            throw new WrongAge("Age cannot be negative ");
        }
        this.fatherAge=fatherAge;
    }
}
```

```
class Son extends Father
{
    int sonAge;
    public Son(int fatherAge,int sonAge) throws WrongAge
    {
        super(fatherAge);
        if(sonAge>=fatherAge)
        {
            throw new WrongAge("Son's age cannot be greater than or equal to son's age");
        }
        this.sonAge=sonAge;
    }
    void display()
    {
        System.out.println("Son's and father's age= "+ sonAge +" "+ fatherAge);
    }
}

class Excep
{
    public static void main(String args[])
    {
        System.out.println("BHOOMIKA HEGDE \n1BM22CS342 ");
        try
        {
            Father f=new Father(-3);
        }
        catch(WrongAge e)
        {
            System.out.println("Wrong Age"+e);
        }
        try
        {
            Son s=new Son(45,56);
        }
        catch(WrongAge e)
        {
            System.out.println("Wrong Age"+e);
        }
        try
        {
            Son s1=new Son(45,19);
        }
        catch(WrongAge e)
        {
            System.out.println("Wrong Age"+e);
        }
        try
        {
            Son s2=new Son(45,19);
            s2.display();
        }
        catch(WrongAge e)
        {
            System.out.println("Wrong Age"+e);
        }


    }
}
```

OUTPUT



```
PS C:\Users\user\java> javac Excep.java
PS C:\Users\user\java> java Excep
BHOOMIKA HEGDE
1BM22CS342
Wrong AgeWrongAge: Age cannot be negative
Wrong AgeWrongAge: Son's age cannot be greater than or equal to son's age
Son's and father's age= 19 45
```

LAB PROGRAM-8

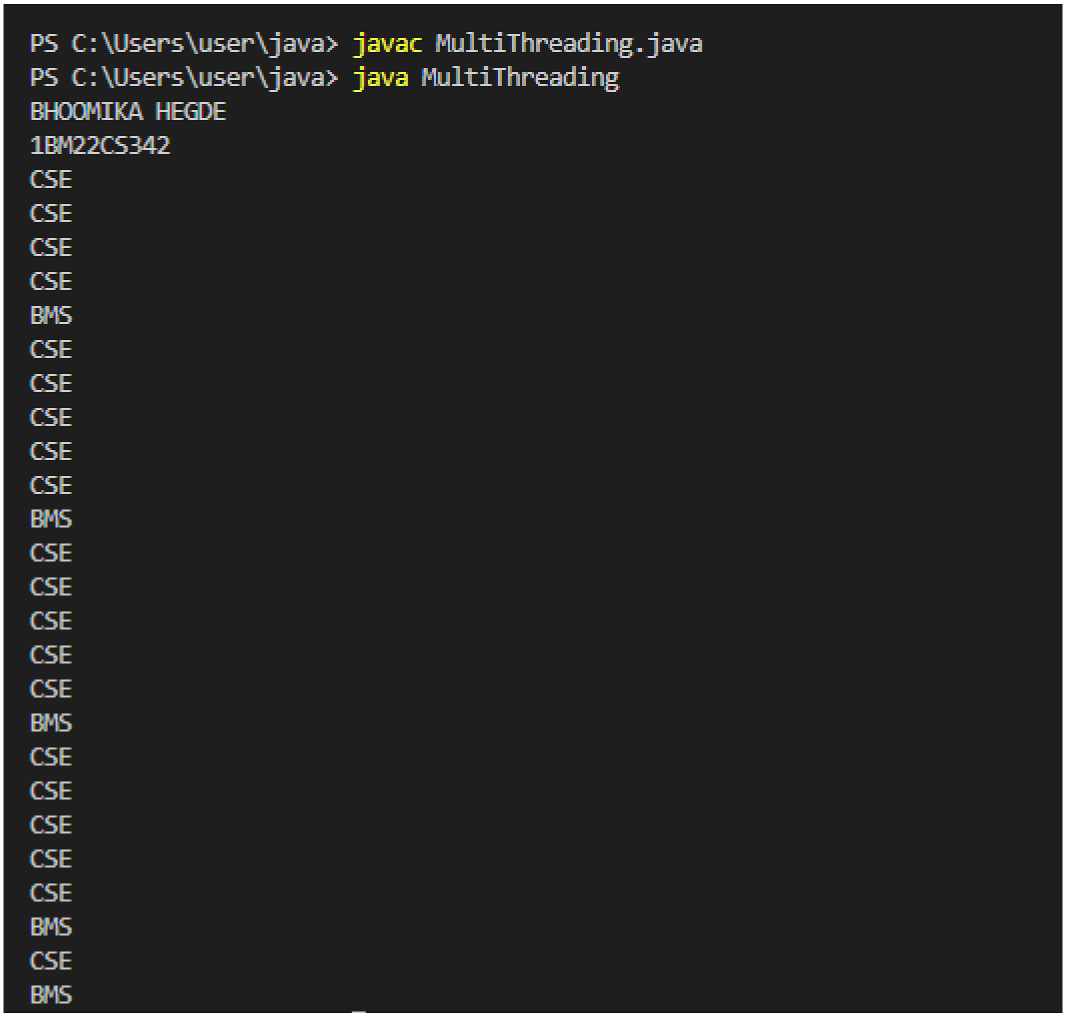
Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds

CODE

```
class Bms extends Thread
{
   public void run()
   {
      int i=0;
      while(i<5)
      {
         try
         {
            Thread.sleep(10000);
            System.out.println("BMS");
            i++;
         }
         catch(InterruptedException e)
         {
            System.out.println("Interrupted");
         }
      }
   }
}
class Cs extends Thread
{
   public void run()
   {
      int i=0;
      while(i<20)
      {
         try
         {
            Thread.sleep(2000);
            System.out.println("CSE");
            i++;
         }
         catch(InterruptedException e)
         {
            System.out.println("Interrupted");
         }
```

```
        }
    }
}

class MultiThreading
{
    public static void main(String args[])
    {
        System.out.println("BHOOMIKA HEGDE \n1BM22CS342 ");
        Bms b=new Bms();
        Cs c=new Cs();
        b.start();
        c.start();
    }
}
```

<u>OUTPUT</u>



```
PS C:\Users\user\java> javac MultiThreading.java
PS C:\Users\user\java> java MultiThreading
BHOOMIKA HEGDE
1BM22CS342
CSE
CSE
CSE
CSE
BMS
CSE
CSE
CSE
CSE
CSE
BMS
CSE
CSE
CSE
CSE
CSE
BMS
CSE
CSE
CSE
CSE
CSE
BMS
CSE
BMS
```