B. M. S. College of Engineering
Bull Temple Road, Bangalore-560019
(Affiliated to Visvesvaraya Technological University, Belgaum)

LAB REPORT

DEPARTMENT : COMPUTER SCIENCE AND ENGINEERING

PROGRAM : UG

ACADEMIC YEAR : Sept-Dec 2020

NAME : RAMYA RAMESH

USN : 1BM19CH038

COURSE NAME : Object Oriented Java Programming

COURSE CODE : 19CS3PCOOJ

NAME: Ramya Ramesh
USN: 1BM19CH038

LAB Program 1

```
import java.util.*;
class Quad {
    public static void main (String args[])
    {
        int a,b,c,count = 0;
        double D;
        Scanner sc = new Scanner(System.in);
        System.out.println("\n Enter the values
          of a,b,c:");
        a = sc.nextInt();
        b = sc.nextInt();
        c = sc.nextInt();
        D = (b*b) - (4*a*c);
        if (D==0)
        {
            System.out.println("\n Roots are real
                and equal");
            count = 1;
        }
        else if (D>0)
        {
            System.out.println("\n Roots are real
            and distinct");
            count = 1;
        }
        else if (D<0)
        {
            System.out.println("\n Roots are
                imaginary");
        }
```

```
if (count == 1)
    {
        double r1 = ((-b + Math.sqrt(D))/(2*a));
        double r2 = ((-b - Math.sqrt(D))/(2*a));
        System.out.println("\n Roots are : " + r1 +
            " , " + r2);
    }
}
}
```

LAB PROGRAM 1

QUESTION:

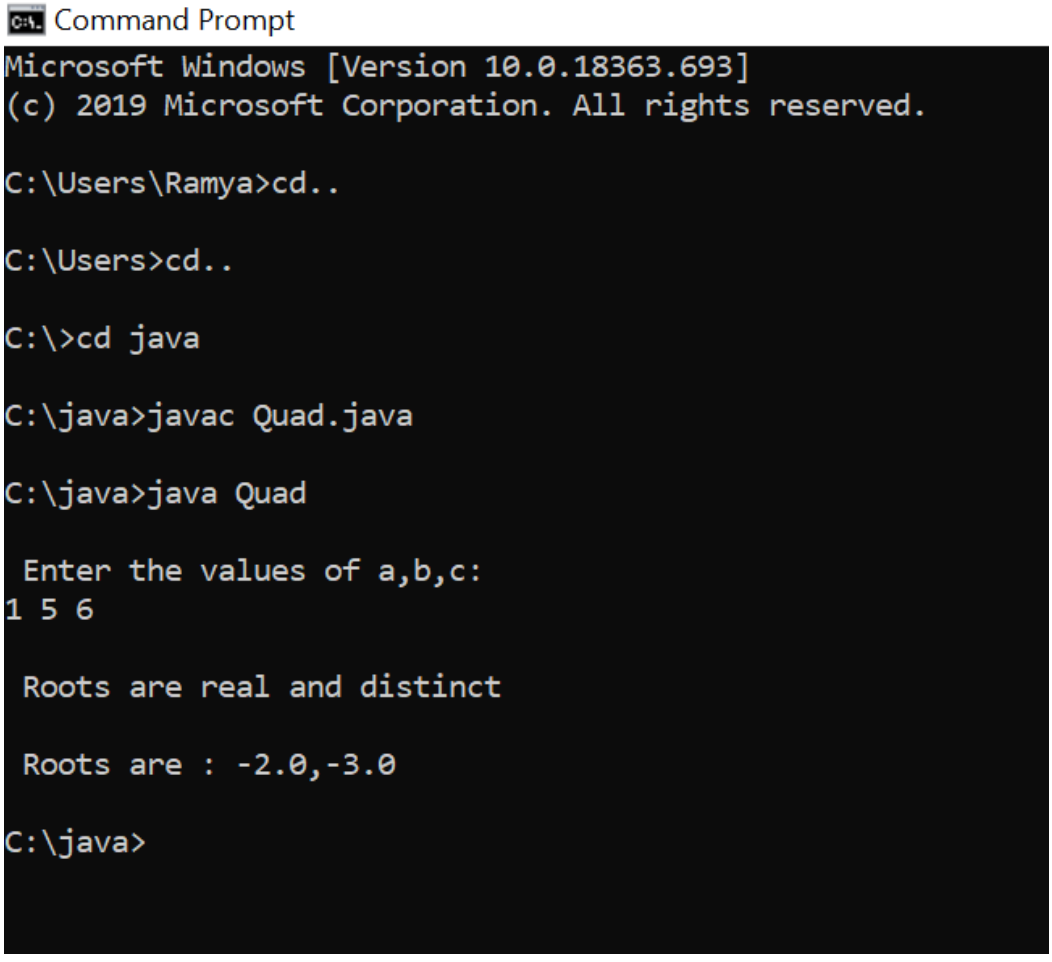
Develop a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c = 0$. Read in a, b, c and use the quadratic formula. If the discriminate b^2-4ac is negative, display a message stating that there are no real solutions.

CODE:

```
import java.util.*;

class Quad{

public static void main(String args[])

{

int a,b,c,count=0;

double D;

Scanner sc=new Scanner(System.in);

System.out.println("\n Enter the values of a,b,c: ");

a=sc.nextInt();

b=sc.nextInt();

c=sc.nextInt();

D= (b*b)-(4*a*c);

if(D==0)

{

System.out.println("\n Roots are real and equal");

count=1;

}

else if(D>0)

{

System.out.println("\n Roots are real and distinct");

count=1;

}

else if(D<0)

{

System.out.println("\n Roots are imaginary");
```

```
}
if(count==1)
{
double r1 = ((-b + Math.sqrt(D))/(2*a));
double r2 = ((-b - Math.sqrt(D))/(2*a));
System.out.println("\n Roots are : "+r1+ "," +r2);
}
}
}
```

OUTPUT:



NAME: Ramya Ramesh
USN: 1BM19CH038

Lab Program 2

```
import java.util.*;
class Student
  {
    String USN;
    String name;
    int credits[];
    int marks[];
    int n, total = 0;
    double SGPA;

  Student()
    {
      SGPA = 0;
    }
  void input() {
    Scanner sc = new Scanner(System.in);
    System.out.println("\n Enter the USN and the
    name of the student: ");
    USN = sc.nextLine();
    name = sc.nextLine();
    System.out.println("\n Enter the number of
      subjects: ");
    n = sc.nextInt();
    credits = new int[n];
    marks = new int[n];
    for(int i=0; i<n; i++)
      {
        System.out.println("\n Enter the marks
        of the student for the subject: "+(i+1));
```

```
        marks[i] = sc.nextInt();
    }
}

void grade_points()
{
    int i;
    for(i=0; i<n; i++)
    {
        if(marks[i] >=90 && marks[i] <100)
        {
            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 if (marks[i] <40)
        {
```

```
            marks[i] = 0 ;
        }
    }
}
    void calculate_SGPA()
    {
        int i ;
        for (i=0 ; i<n ; i++)
        {
            SGPA = SGPA + (credits[i] * marks[i]);
        }
        SGPA = SGPA / total ;
    }
void display_details()
    {
        System.out.println("\n The student with
USN : "+USN+", Name : "+name+" has
secured SGPA : "+SGPA);
    }
public static void main(String[] args)
    {
        Student obj = new Student();
        obj.input();
        obj.grade_points();
        obj.calculate_SGPA();
        obj.display_details();
    }
}
```

LAB PROGRAM 2

QUESTION : 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 credits[];

 int marks[];

 int n,total=0;

 double SGPA;


Student()

{

 SGPA=0;

}

void input()

{

 Scanner sc=new Scanner(System.in);

 System.out.println("\n Enter the USN and the name of the student: ");

 USN=sc.nextLine();

 name=sc.nextLine();

 System.out.println("\n Enter the number of subjects: ");

 n=sc.nextInt();

 credits=new int[n];

 marks =new int[n];

 for(int i=0;i<n;i++)
```

```
{
 System.out.println("\n Enter the credits for subject: "+(i+1));
 credits[i]=sc.nextInt();
 total=total+credits[i];
}
for(int i=0;i<n;i++)
{
 System.out.println("\n Enter the marks of the student for subject: "+(i+1));
 marks[i]=sc.nextInt();
}
}
void grade_points()
{
   int i;
   for(i=0;i<n;i++)
   {
      if(marks[i]>=90 && marks[i]<100)
      {
         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]=4;
        }
        else if(marks[i]<40)
        {
           marks[i]=0;
        }
    }
}
void calculate_SGPA()
{
 int i;
 for(i=0;i<n;i++)
 {
  SGPA=SGPA+(credits[i]*marks[i]);
 }
  SGPA=SGPA/total;
}
void display_details()
{
   System.out.println("\n The student with USN: "+USN+" and Name: "+name+" has secured SGPA:
"+SGPA);
}
public static void main(String[] args)
{
```

```
Student obj=new Student();
obj.input();
obj.grade_points();
obj.calculate_SGPA();
obj.display_details();
}
}
```

OUTPUT :

Command Prompt

```
Microsoft Windows [Version 10.0.18363.693]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Ramya>cd..

C:\Users>cd..

C:\>cd java

C:\java>javac Student.java

C:\java>java Student

 Enter the USN and the name of the student:
1BM19CS001
Ramya

 Enter the number of subjects:
5

 Enter the credits for subject: 1
4

 Enter the credits for subject: 2
3

 Enter the credits for subject: 3
3

 Enter the credits for subject: 4
3

 Enter the credits for subject: 5
4

 Enter the marks of the student for subject: 1
89

 Enter the marks of the student for subject: 2
90

 Enter the marks of the student for subject: 3
83

 Enter the marks of the student for subject: 4
87

 Enter the marks of the student for subject: 5
91

 The student with USN: 1BM19CS001 and Name: Ramya has secured SGPA: 9.411764705882353

C:\java>
```

NAME : Ramya Ramesh
USN : 1BM19CH038

LAB PROGRAM 3 :

```
class Book_details
{
    private String name, author;
    private double price;
    private int num_pages;

    Book_details()
    {
        name = "The Immortals of Meluha";
        author = "Amish";
        price = 234.00;
        num_pages = 390;
    }

    void getDetails()
    {
        Scanner in = new Scanner(System.in);
        System.out.println("\n Enter the book name:");
        name = in.nextLine();
        System.out.println("\n Enter the author name:");
        author = in.nextLine();
        System.out.println("\n Enter the no. of pages:");
        num_pages = in.nextInt();
        System.out.println("\n Enter the price");
        price = in.nextDouble();
    }
```

```
public String toString()
    {
        String temp = "\n Book name : " +name+ "\n
Author name : "+author+"\n No. of pages : "
+num_pages + "\n Price : " +price+ "\n";
        return (temp);
    }
}

class Book
{
    public static void main (String args[])
    {
        int i, n;
        Scanner in = new Scanner (System.in);
        System.out.println("\n Enter the number of
            books : ");
        n = in.nextInt();
        Book_details [] obj = new Book_details [n];
        for (i=0; i<n; i++)
        {
            obj[i] = new Book_details();
        }
        System.out.println ("\t\t *** Enter Book
            Details *** ");
        for (i=0; i<n; i++)
        {
            System.out.println ("\n Book " +(i+1)+ ":");
            obj[i].getDetails();
        }
    System.out.println ("\t\t *** Book Details *** ");
        for (i=0; i<n; i++)
```

```
{
        System.out.println(obj[i]);
    }
}
}
```

LAB PROGRAM 3:

QUESTION :

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.

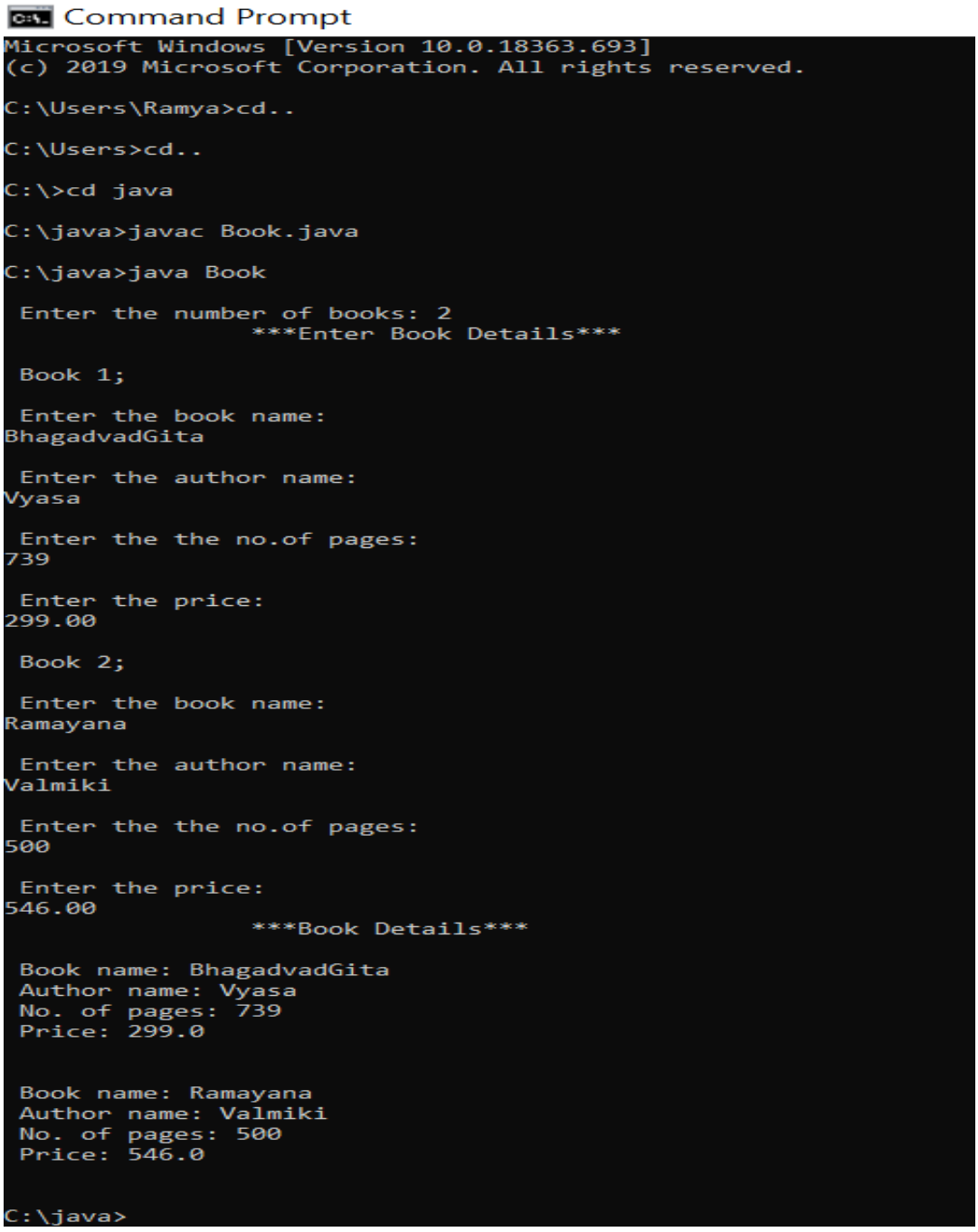
CODE :

```
class Book_details
{
        private String name,author;
        private double price;
        private int num_pages;

    Book_details()
        {
                name="The Immortals of Meluha";
                author="Amish";
                price=234.00;
                num_pages=390;
        }

        void getDetails()
        {
                Scanner in=new Scanner(System.in);
                System.out.println("\n Enter the book name: ");
                name=in.nextLine();
                System.out.println("\n Enter the author name: ");
                author=in.nextLine();
                System.out.println("\n Enter the the no.of pages: ");
                num_pages=in.nextInt();
                System.out.println("\n Enter the price: ");
                price=in.nextDouble();
        }
```

```
	public String toString()
	{
		String temp="\n Book name: "+name+"\n Author name: "+author+"\n No. of pages: "+num_pages+"\n Price: "+price+"\n";
		return(temp);
	}

}
class Book
{
  public static void main(String args[])
  {
	int i,n;
	Scanner in=new Scanner(System.in);
	System.out.print("\n Enter the number of books: ");
	n=in.nextInt();
	Book_details[] obj=new Book_details[n];
	for(i=0;i<n;i++)
	{
		obj[i]=new Book_details();
	}
	System.out.println("\t\t***Enter Book Details***");
      for(i=0;i<n;i++)
	{
		System.out.println("\n Book "+(i+1)+";");
		obj[i].getDetails();
	}
	System.out.println("\t\t***Book Details***");
      for(i=0;i<n;i++)
	{
		System.out.println(obj[i]);
	}
```

```
  }
}
```

OUTPUT :


Command Prompt
Microsoft Windows [Version 10.0.18363.693]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Book.java
C:\java>java Book
Enter the number of books: 2
Enter Book Details
Book 1;
Enter the book name:
BhagadvadGita
Enter the author name:
Vyasa
Enter the the no.of pages:
739
Enter the price:
299.00
Book 2;
Enter the book name:
Ramayana
Enter the author name:
Valmiki
Enter the the no.of pages:
500
Enter the price:
546.00
Book Details
Book name: BhagadvadGita
Author name: Vyasa
No. of pages: 739
Price: 299.0
Book name: Ramayana
Author name: Valmiki
No. of pages: 500
Price: 546.0
C:\java>

Date: YOUVA

NAME : Ramya Ramesh
USN : 1BM19CH038

LAB PROGRAM 4

```
abstract Class Shape {
    int a = 3;
    int b = 4;
abstract public void print_area();
}

class rectangle extends Shape
    {
        public int area_rect;
        public void print_area()
        {
            area_rect = a*b;
            System.out.println("\n The area of the
rectangle is : " + area_rect);
        {
    }
class triangle extends Shape
    {
        int area_tri;
        public void print_area()
        {
            area_tri=(int)(0.5 * a * b);
            System.out.println("\n The area of
triangle is : " + area_tri);
        }
    }
class circle extends Shape
    {
        int area_circle;
        public void print_area()
        {
```

```
        area_circle = (int)(3.14 * 2 * 2);
        System.out.println("\n The area of circle is : "
            +area_circle);
        }
}
class Shape_area {
    public static void main (String[] args) {
        rectangle rec = new rectangle();
        rec.print_area();
        triangle tri = new triangle();
        tri.print_area();
        circle cir = new circle();
        cir.print_area();
    }
}
```

LAB PROGRAM 4

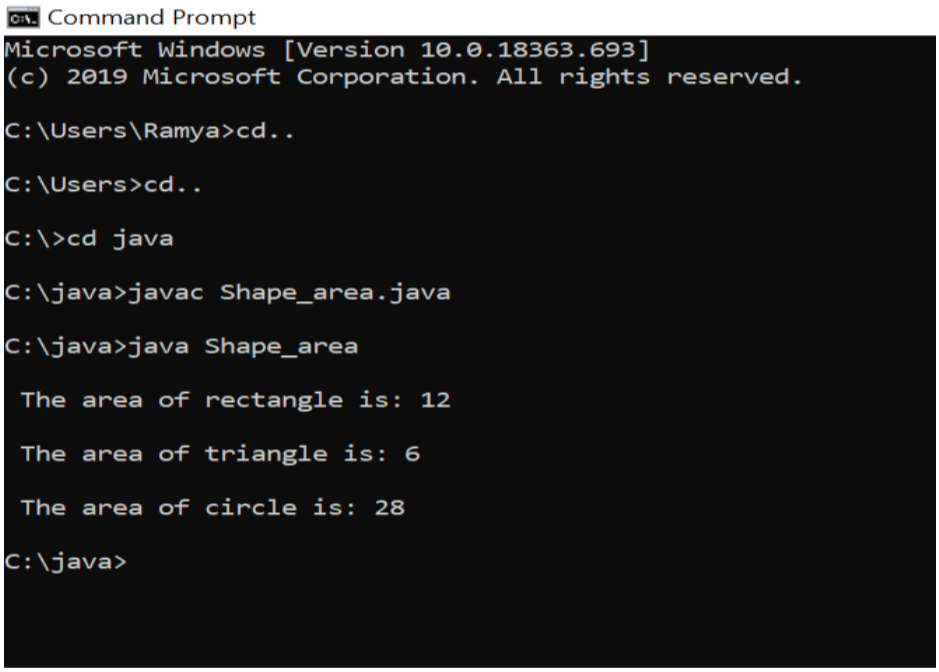
QUESTION: Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method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

CODE:

```
abstract class Shape
{
int a=3;
int b=4;
abstract public void print_area();
}
class rectangle extends Shape
{
public int area_rect;

public void print_area()
{
area_rect=a*b;
System.out.println("\n The area of rectangle is: "+area_rect);
}
}
class triangle extends Shape
{
int area_tri;

public void print_area()
{
area_tri=(int) (0.5*a*b);
System.out.println("\n The area of triangle is: "+area_tri);
}
}
class circle extends Shape
```

```
{
int area_circle;

public void print_area()
{
area_circle=(int) (3.14*a*a);
System.out.println("\n The area of circle is: "+area_circle);
}
}
class Shape_area{
public static void main(String[] args){
rectangle rec = new rectangle();
rec.print_area();
triangle tri = new triangle();
tri.print_area();
circle cir = new circle();
cir.print_area();
}
}
```

OUTPUT:



NAME : Ramya Ramesh
USN : 1BM19CH038

LAB PROGRAM 5 :

```
import java.util.Scanner ;
class Account {
  private String name ;
  private long account_number;
  private int account_type;
  double balance;
 void get_data()
  {
     Scanner ss = new Scanner(System.in)
     System.out.println("\n Enter your Name : ");
     name = ss.nextLine();
     System.out.println("\n Enter the Account Number : ");
     account_number = ss.nextLong();
     System.out.println("\n Choose the account type : ");
      System.out.println("\n1. Savings account ");
      System.out.println("\n 2. Current account ");
       account_type = ss.nextInt();
  }
  int return_account_type()
    {
      return account_type;
    }
}
```

```
class Sav_acct extends Account
  {
     Scanner ss = new Scanner(System.in);
     double amount;
     void get_sav_balance()
        {
           System.out.println("\n Enter the amount to
     be placed in your savings account: ");
           amount = ss.nextDouble();
           balance = balance + amount;
        }

     void display_sav_blnce()
        {
          System.out.println("\n Balance = " +balance);
        }

     void compute_sav_interest()
        {
          System.out.println("\n Interest of 5% shall be
     added to your balance ");
           balance = balance + (0.05*balance);
        }

     void withdrawal_sav()
        {
          System.out.println("\n Enter the amount to be
            withdrawn: ");
          amount = ss.nextDouble();
          balance = balance - amount;
        }
  }
```

```
class Curr_acct extends Account
{
  Scanner ss = new Scanner (System.in);
  double amount;
  final double min_balance = 5000;

void get_cur_balance()
 {
  System.out.println("\n Enter the amount to be placed
    in your current account : ");
    amount = ss.nextDouble();
    balance += amount;
 }
 void display_cur_blnce()
  {
    System.out.println("\n Balance = " +balance);
  }

void compute_cur_service_charges()
 {
   if(balance < min_balance)
   {
    System.out.println("\n Service tax of rs. 500 shall be
      levied");
    balance = balance - 500;
   }
  else
    {
     System.out.println("\nMinimum balance is
     maintained");
    }
}
```

```
void withdrawal_cur()
 {
   System.out.println("\n Enter the amount to be
    withdrawn : ");
    amount = ss.nextDouble();
    balance = balance - amount;
  }
}

class Bank
{
  public static void main (String[] args)
   {
     System.out.println("\n Enter the bank details: ");
     Account acc = new Account();
     acc.get_data();
     int type = acc.return_accant_type();
     if(type == 1)
  {
     System.out.println("*** SAVINGS ACCOUNT *** ");
     Sav_acct sav = new Sav_acct();
     sav.get_sav_balance();
     sav.display_sav_blnce();
     sav.compute_sav_interest();
     sav.display_sav_blnce();
     sav.withdrawal_sav();
     sav.display_sav_blnce();
  }
```

```
if (type == 2)
{
    System.out.println("*** CURRENT ACCOUNT ***");
    Curr_acct cur = new Curr_acct();
    cur.get_cur_balance();
    cur.display_cur_Bnce();
    cur.compute_cur_service_charges();
    cur.display_cur_Bnce();
    cur.withdrawal_cur();
    cur.display_cur_Bnce();
}
}
```

LAB PROGRAM 5

QUESTION: 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 Curr-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
{
        private String name;
        private long account_number;
        private int account_type;
        double balance;


        void get_data()
        {
         Scanner ss=new Scanner(System.in);
         System.out.println("\n Enter your name: ");
         name=ss.next();
         System.out.println("\n Enter the Account number: ");
         account_number=ss.nextLong();
      System.out.println("\n Choose the account type: ");
         System.out.println("\n 1. Savings account");
         System.out.println("\n 2. Current account");
         account_type=ss.nextInt();
        }
```

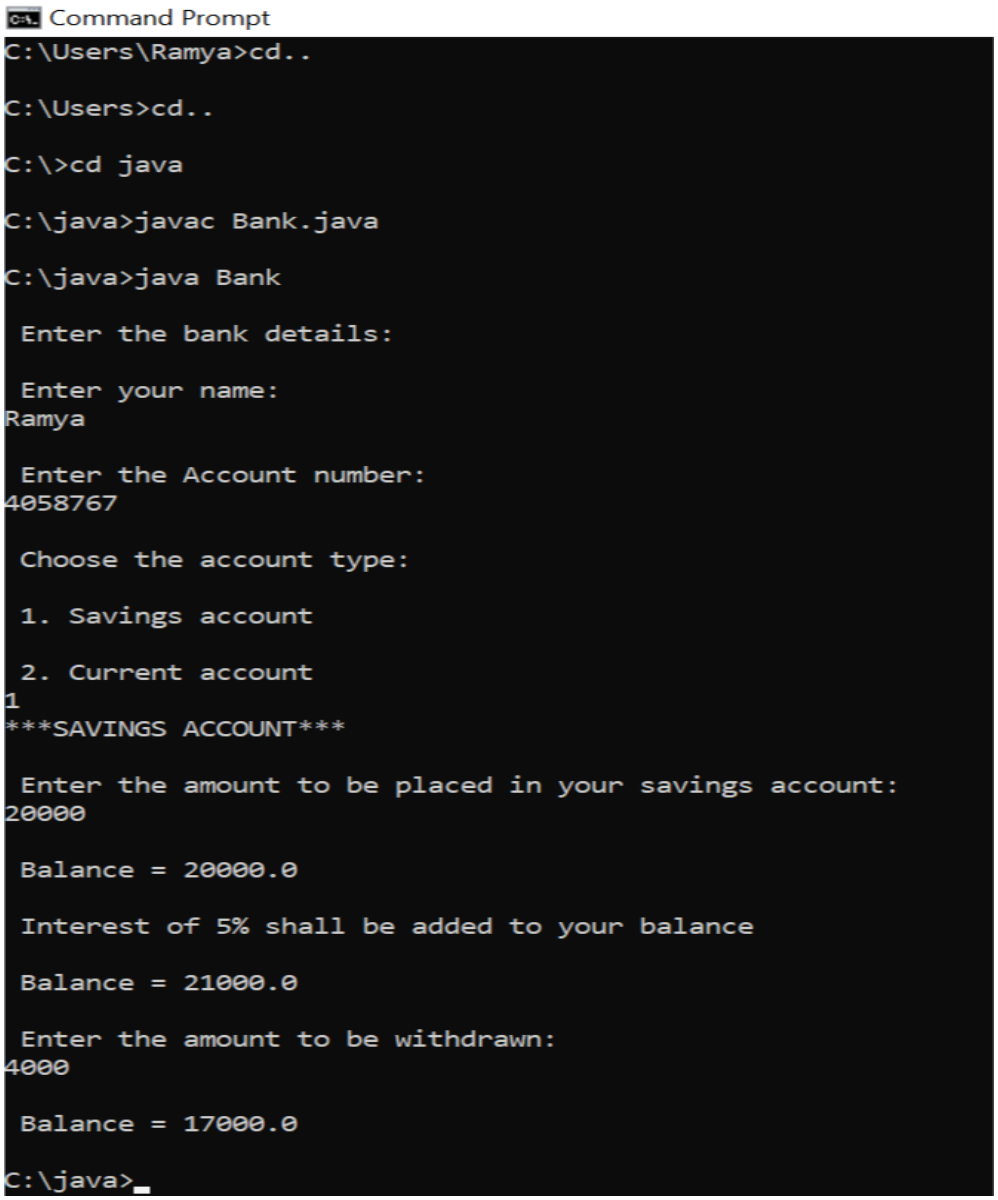
```
        int return_account_type()
        {
         return account_type;
        }
}

class Sav_acct extends Account
{
        Scanner ss=new Scanner(System.in);
        double amount;

        void get_sav_balance()
        {
         System.out.println("\n Enter the amount to be placed in your savings account: ");
         amount=ss.nextDouble();
         balance= balance+amount;
        }

        void display_sav_blnce()
        {
         System.out.println("\n Balance = "+balance);
        }

        void compute_sav_interest()
        {
         System.out.println("\n Interest of 5% shall be added to your balance");
         balance=balance+(0.05*balance);
        }

        void withdrawl_sav()
```

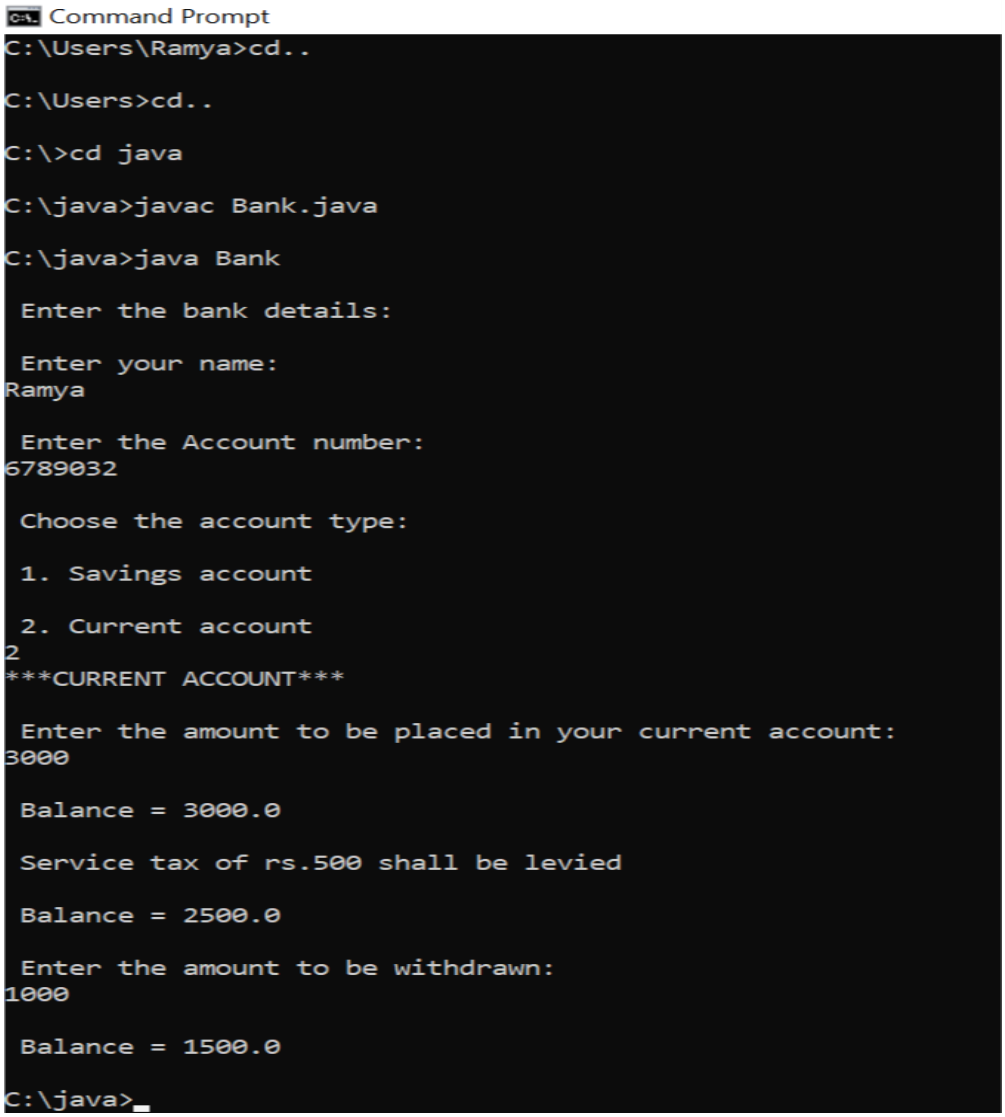
```
        {
         System.out.println("\n Enter the amount to be withdrawn: ");
         amount=ss.nextDouble();
         balance=balance-amount;
        }

}

class Curr_acct extends Account
{
        Scanner ss=new Scanner(System.in);
        double amount;
        final double min_balance=5000;

        void get_cur_balance()
        {
         System.out.println("\n Enter the amount to be placed in your current account: ");
         amount=ss.nextDouble();
         balance+=amount;
        }

        void display_cur_blnce()
        {
         System.out.println("\n Balance = "+balance);
        }

        void compute_cur_service_charges()
        {
          if(balance<min_balance)
                {
                 System.out.println("\n Service tax of rs.500 shall be levied");
```

```
                balance=balance-500;
                }
            else
                {
                System.out.println("\n Minimum balance is maintained");
                }
        }

        void withdrawl_cur()
        {
         System.out.println("\n Enter the amount to be withdrawn: ");
         amount=ss.nextDouble();
         balance=balance-amount;
        }

}

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

        System.out.println("\n Enter the bank details: ");
        Account acc=new Account();
        acc.get_data();
        int type=acc.return_account_type();
        if (type==1)
     {
      System.out.println("***SAVINGS ACCOUNT***");
      Sav_acct sav=new Sav_acct();
           sav.get_sav_balance();
```

```
            sav.display_sav_blnce();
            sav.compute_sav_interest();
            sav.display_sav_blnce();
            sav.withdrawl_sav();
            sav.display_sav_blnce();
        }
        if(type==2)
        {
         System.out.println("***CURRENT ACCOUNT***");
            Curr_acct cur=new Curr_acct();
            cur.get_cur_balance();
            cur.display_cur_blnce();
            cur.compute_cur_service_charges();
            cur.display_cur_blnce();
            cur.withdrawl_cur();
            cur.display_cur_blnce();
           }
}

}
```

OUTPUT:


Command Prompt
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Bank.java
C:\java>java Bank
Enter the bank details:
Enter your name:
Ramya
Enter the Account number:
4058767
Choose the account type:
1. Savings account
2. Current account
1
SAVINGS ACCOUNT
Enter the amount to be placed in your savings account:
20000
Balance = 20000.0
Interest of 5% shall be added to your balance
Balance = 21000.0
Enter the amount to be withdrawn:
4000
Balance = 17000.0
C:\java>


Command Prompt
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Bank.java
C:\java>java Bank
Enter the bank details:
Enter your name:
Ramya
Enter the Account number:
6789032
Choose the account type:
1. Savings account
2. Current account
2
CURRENT ACCOUNT
Enter the amount to be placed in your current account:
3000
Balance = 3000.0
Service tax of rs.500 shall be levied
Balance = 2500.0
Enter the amount to be withdrawn:
1000
Balance = 1500.0
C:\java>

NAME : Ramya Ramesh
USN : 1BM19CH038

LAB PROGRAM 6

i) Student.java

```
package CIE
import java.util.Scanner;
public class Student
 {
    public String name;
    public String usn;
    public int sem;
    public void display()
     {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter Name: \n");
        name = s.nextLine();
        System.out.println("Enter USN: \n");
        usn = s.nextLine();
        System.out.println("Enter Semester: \n");
        sem = s.nextInt();
     }
 }
```

ii) Internals.java

```
package CIE;
import java.util.Scanner;
public class Internals extends Student
 {
    public double cie[];
```

```
public void display()
{
    cie = new double[5];
    Scanner sc = new Scanner(System.in);
    System.out.println("The CIE marks for 5
    subjects (out of 50) are :\n");
    for(int i=0; i<5; i++)
    {
        cie[i] = sc.nextDouble();
    }
}
}
```

iii) Externals.java

```
package SEE;
import java.util.*;
import CIE.*;
public class Externals extends CIE.Student
{
    public double see[];

    public void display()
    {
        see = new double[5];
        Scanner s = new Scanner(System.in);
        System.out.println("The SEE marks for
        the 5 subjects (out of 100) are :\n");
        for(int i=0; i<5; i++)
        {
            see[i] = s.nextDouble();
        }
    }
}
```

iv) Main.java

```
import CIE.*;
import SEE.*;
import java.util.Scanner;
public class Main
{
  public static void main(String args[])
  {
    int n;
    Scanner s = new Scanner(System.in);
    System.out.println("Enter the number of
      students: \n");
    n = s.nextInt();
    CIE.Student st[] = new CIE.Student[n];
    CIE.Internals in[] = new CIE.Internals[n];
    SEE.Externals ex[] = new SEE.Externals[n];
    for(int i=0; i<n; i++)
    {
      st[i] = new CIE.Student();
      in[i] = new CIE.Internals();
      ex[i] = new SEE.Externals();
      st[i].display();
      in[i].display();
      ex[i].display();
    System.out.println("The Total marks of the
      student " +st[i].name+ " in the 5
      subjects are: \n");
    for(int j=0; j<5; j++)
    {
      System.out.println(in[i].cie[j] + (
        ex[i].see[j]/2));
    }
  }
}
}
```

NAME: RAMYA RAMESH

USN: 1BM19CH038

LAB PROGRAM – 6

QUESTION:

Create a package CIE which has two classes- Student and Internals. The class Personal 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:

1) Student.java

```
package CIE;
import java.util.Scanner;
public class Student
{
public String name;
public String usn;
public int sem;
public void display()
{
Scanner s=new Scanner(System.in);
System.out.println("Enter Name: \n");
name=s.nextLine();
System.out.println("Enter USN: \n");
usn=s.nextLine();
 System.out.println("Enter Semester: \n");
```

```
sem=s.nextInt();
}
}
```

2) Internals.java

```
package CIE;
import java.util.Scanner;
public class Internals extends Student
{
public double cie[];

public void display()
{
cie=new double[5];
Scanner sc=new Scanner(System.in);
System.out.println("The CIE marks for 5 subjects(out of 50) are: \n");
 for(int i=0;i<5;i++)
{
cie[i]=sc.nextDouble();
}
}
}
```

3) Externals.java

```
package SEE;
import java.util.*;
import CIE.*;
public class Externals extends CIE.Student
{
public double see[];

public void display()
{
```

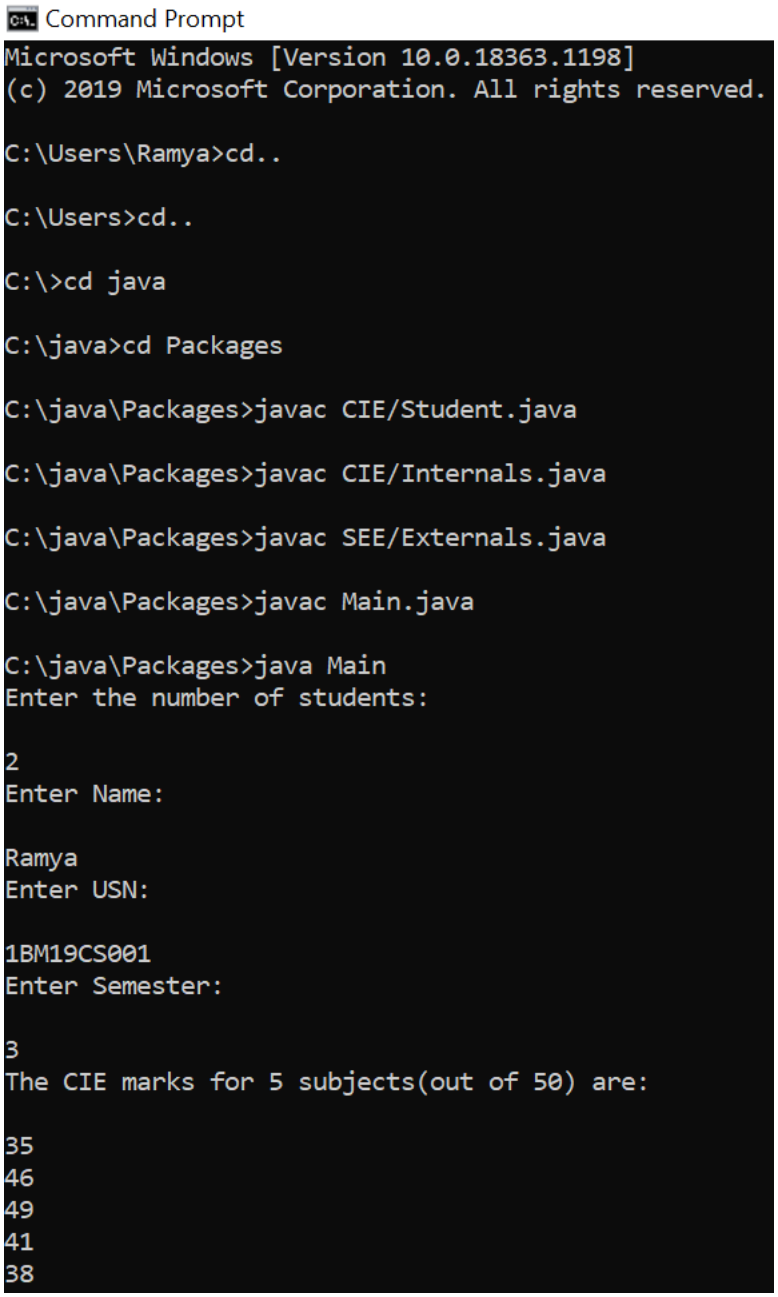
```
see=new double[5];
Scanner s=new Scanner(System.in);
System.out.println("The SEE Marks for the 5 subjects(out of 100) are: \n");
for(int i=0;i<5;i++)
{
see[i]=s.nextDouble();
}
}
}
```

4) Main.java

```
import CIE.*;
import SEE.*;
import java.util.Scanner;
public class Main
{
public static void main(String args[])
{
int n;
Scanner s=new Scanner(System.in);
System.out.println("Enter the number of students:\n");
n=s.nextInt();
CIE.Student st[]=new CIE.Student[n];
CIE.Internals in[]=new CIE.Internals[n];
SEE.Externals ex[]=new SEE.Externals[n];
for(int i=0;i<n;i++)
{
st[i]=new CIE.Student();
in[i]=new CIE.Internals();
ex[i]=new SEE.Externals();
st[i].display();
in[i].display();
```

```
ex[i].display();
System.out.println("The Total marks of the student "+st[i].name+ " in the 5 subjects are: \n");
 for(int j=0;j<5;j++)
{
System.out.println(in[i].cie[j]+(ex[i].see[j]/2));
}
}
}
}
```

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>cd Packages
C:\java\Packages>javac CIE/Student.java
C:\java\Packages>javac CIE/Internals.java
C:\java\Packages>javac SEE/Externals.java
C:\java\Packages>javac Main.java
C:\java\Packages>java Main
Enter the number of students:
2
Enter Name:
Ramya
Enter USN:
1BM19CS001
Enter Semester:
3
The CIE marks for 5 subjects(out of 50) are:
35
46
49
41
38

```
The SEE Marks for the 5 subjects(out of 100) are:

78
89
94
83
80
The Total marks of the student Ramya in the 5 subjects are:

74.0
90.5
96.0
82.5
78.0
Enter Name:

Ramesh
Enter USN:

1BM19CS002
Enter Semester:

3
The CIE marks for 5 subjects(out of 50) are:

45
46
39
41
37
```

```
The SEE Marks for the 5 subjects(out of 100) are:

93
95
77
81
75
The Total marks of the student Ramesh in the 5 subjects are:

91.5
93.5
77.5
81.5
74.5

C:\java\Packages>
```

NAME : Ramya Ramesh
USN : 1BM19CH038

LAB PROGRAM 7

```
class Student <V,U> {
    V v;
    U u;
  Student (V v, U u) {
        this.v = v;
        this.u = u;
    }
  public void display () {
      System.out.println ("\n" +this.v+ " is : " +this.u);
    System.out.println ("Type of V is" + v.getClass().
            getName());
  System.out.println(" Type of U is : +u.getClass().getName());
      }
  }
  public class Generics {
      public static void main (String args[]) {
      Student <String, String> std1 = new Student <String,
  String> ("Name", "Ramya");
      Student <String, Integer> std2 = new Student
      <String, Integer> ("Age", "18");
  Student <String, Float> std3 = new Student <String,
  Float> ("Percentage", 94.5f);
          std1.display();
          std2.display();
          std3.display();
      }
  }
```

LAB PROGRAM 7

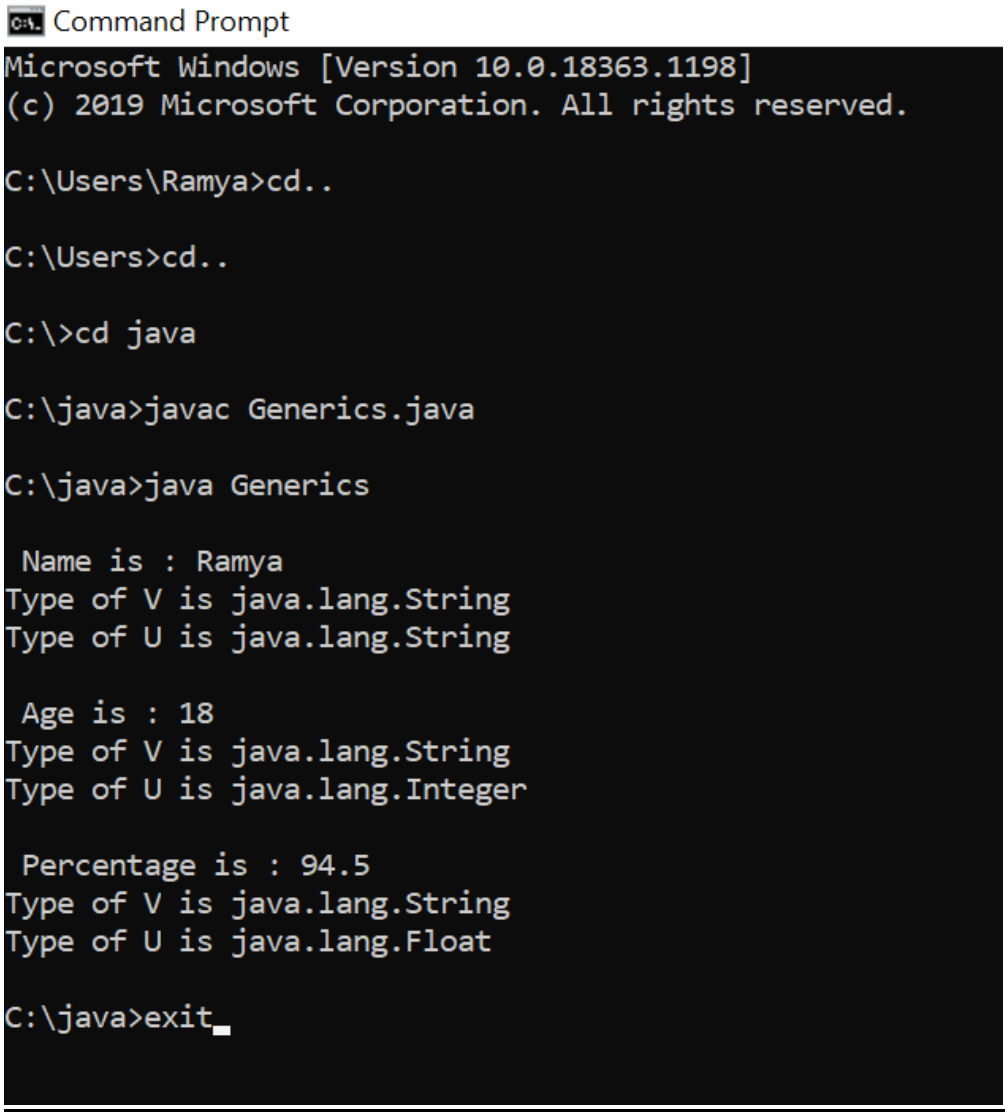
QUESTION:

Write a program to demonstrate generics with multiple object parameters.

CODE:

```
class Student<V, U> {
  V v;
  U u;
  Student(V v, U u){
    this.v = v;
    this.u = u;
  }
  public void display() {
    System.out.println("\n "+this.v+" is : "+this.u);
    System.out.println("Type of V is " +v.getClass().getName());
    System.out.println("Type of U is " +u.getClass().getName());
  }
}
public class Generics {
  public static void main(String args[]) {
    Student<String, String> std1 = new Student<String, String>("Name", "Ramya");
    Student<String, Integer> std2 = new Student<String, Integer>("Age", 18);
    Student<String, Float> std3 = new Student<String, Float>("Percentage", 94.5f);
    std1.display();
    std2.display();
    std3.display();
  }
}
```

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Generics.java
C:\java>java Generics
Name is : Ramya
Type of V is java.lang.String
Type of U is java.lang.String
Age is : 18
Type of V is java.lang.String
Type of U is java.lang.Integer
Percentage is : 94.5
Type of V is java.lang.String
Type of U is java.lang.Float
C:\java>exit

NAME : Pamya Pamesh
USN : 1BM19CH038

LAB PROGRAM 8

```
import java.util.Scanner;
class WrongAge extends Exception {
    int fatherAge;
WrongAge (int fAge)
{
    this.fatherAge = fAge;
}
public String toString()
    {
        return ("\n ERROR : Father's age can't be negative!");
    }
}
class SonException extends Exception
  {
     int f, s;
     SonException(int fAge, int sAge)
    {
       this.f = fAge;
       this.s = sAge;
     }
  public String toString()
     {
        if (f==s)
        return ("\n ERROR : Son's age can't be equal to
     father's age!");
        if(s<0)
        return ("\n ERROR : Son's age can't be lesser
than zero!");
        else
        return ("\n ERROR : Son's age can't be
greater than father's age!");
     }
  }
```

```
class Father
{
    int fAge;
    Scanner sc = new Scanner(System.in);
    Father()
    {
        System.out.println("Enter the father's age :\n");
        fAge = sc.nextInt();
    }
    void exception1() throws WrongAge
    {
        if(fAge < 0)
        throw new WrongAge(fAge);
    }
}
class Son extends Father
{
    int sAge;
    Scanner sc = new Scanner(System.in);
    Son()
    {
        super();
        System.out.println("Enter the son's age:\n");
        sAge = sc.nextInt();
    }
void exception2() throws SonException
    {
        if(sAge < 0 || sAge >= fAge)
        throw new SonException(fAge, sAge);
    }
}
class except
{
    public static void main(String args[])
```

```
{
    Son s : new Son();
    try {
        s.exception1();
    }
catch (WrongAge e)
    {
        System.out.println("Exception caught"+e);
    }
try {
    s.exception2();
}
catch (SonException e)
    {
        System.out.println("Exception caught" +e);
    }
}
```

NAME: Ramya Ramesh

USN: 1BM19CH038

LAB PROGRAM 8

QUESTION: 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.Scanner;

class WrongAge extends Exception {

   int fatherAge;

 WrongAge(int fAge)

 {

 this.fatherAge=fAge;

 }

 public String toString()

 {

 return("\n ERROR: Father's age can't be negative!");

 }

}


class SonException extends Exception

{

int f,s;

SonException(int fAge,int sAge)

{

 this.f=fAge;

 this.s=sAge;

}

public String toString()

 {
```

```
 if(f==s)
 return("\n ERROR: Son's age can't be equal to father's age!");
 if(s<0)
 return("\n ERROR: Son's age can't be less than zero!");
 else
 return("\n ERROR: Son's age can't be greater than father's age!");
 }
}

class Father
{
int fAge;
Scanner sc=new Scanner(System.in);
 Father()
 {
 System.out.println("Enter the father's age:\n");
 fAge=sc.nextInt();
 }
void exception1() throws WrongAge
 {
 if(fAge<0)
 throw new WrongAge(fAge);
}
}
class Son extends Father
{
int sAge;
Scanner sc=new Scanner(System.in);
Son()
 {
 super();
```

```
 System.out.println("Enter the son's age:\n");
 sAge=sc.nextInt();
 }
void exception2() throws SonException
 {
 if(sAge<0 || sAge>=fAge)
 throw new SonException(fAge,sAge);
 }
}
class except
{
public static void main(String args[])
 {
 Son s =new Son();
 try{
 s.exception1();
 }
 catch(WrongAge e)
 {
 System.out.println("Exception caught" + e);
 }
 try{
 s.exception2();
 }
 catch(SonException e)
 {
 System.out.println("Exception caught" + e);
 }
 }
}
```

OUTPUT:

Command Prompt

```
C:\Users>cd..

C:\>cd java

C:\java>javac except.java

C:\java>java except
Enter the father's age:

45
Enter the son's age:

45
Exception caught
 ERROR: Son's age can't be equal to father's age!

C:\java>java except
Enter the father's age:

-10
Enter the son's age:

15
Exception caught
 ERROR: Father's age can't be negative!
Exception caught
 ERROR: Son's age can't be greater than father's age!

C:\java>java except
Enter the father's age:

20
Enter the son's age:

25
Exception caught
 ERROR: Son's age can't be greater than father's age!

C:\java>
```

Date: YOUVA

NAME : Pamya Ramesh
USN : 1BM19CH038

Lab Program 9

```
import java.util.Scanner;
class College extends Thread {
    int time;
    String str;
    College (int t, String s) {
        time = t;
        str = s;
    }
    public void run () {
        try {
            int i=0;
            while (i<5) {
                System.out.println (str);
                sleep(time);
                i++;
            }
        } catch (Exception e) {
        }
    }
}
class MultithreadDemo {
    public static void main(String args[]) {
        College t1 = new College (10000, "BMS College
                                  of Engineering");
        College t2 = new College (2000, "CSE");
        t1.start();
        t2.start();
    }
}
```

NAME: Ramya Ramesh

USN: 1BM19CH038

LAB PROGRAM 9

QUESTION:

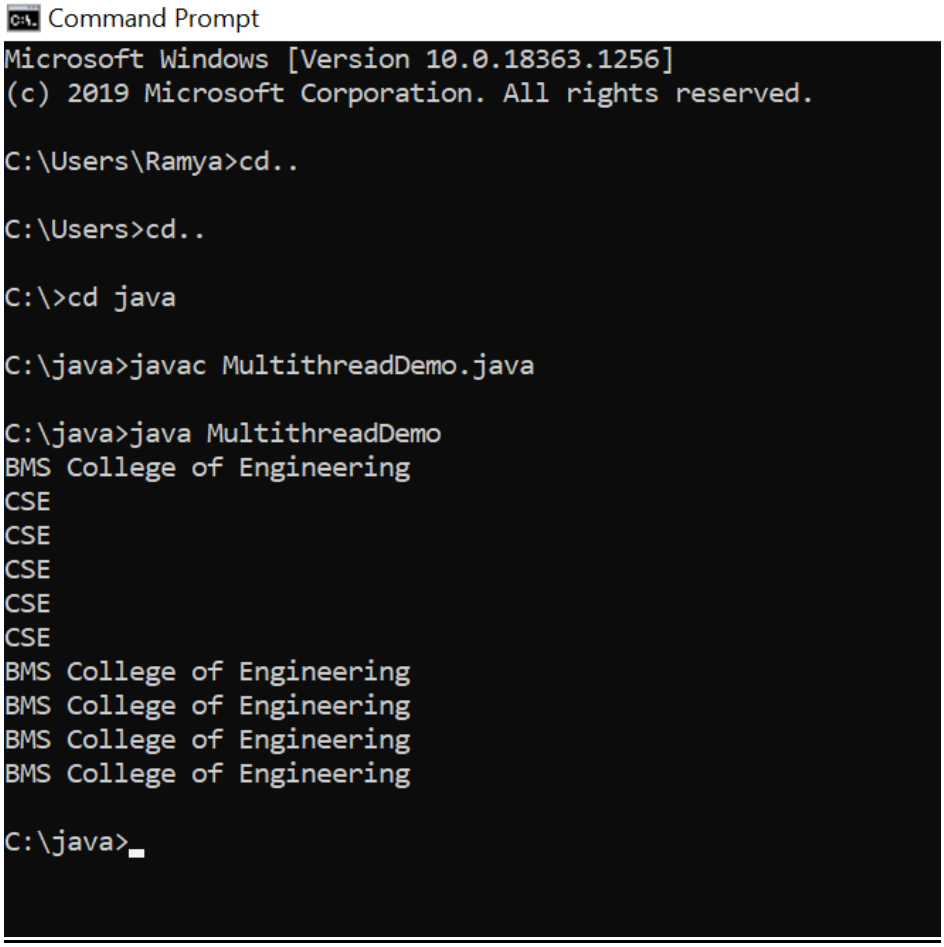
Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

CODE:

```
import java.util.Scanner;
class College extends Thread{
 int time;
 String str;
 College(int t,String s){
  time =t;
  str = s;
 }
  public void run(){
      try{
         int i=0;
         while (i<5){
            System.out.println(str);
            sleep(time);
            i++;
         }
      } catch (Exception e){
      }
   }
}
class MultithreadDemo {
```

```
    public static void main(String args[]) {
        College t1 = new College(10000,"BMS College of Engineering");
        College t2 = new College(2000,"CSE");
        t1.start();
        t2.start();
      }
}
```

OUTPUT:



NAME : Ramya Ramesh
USN : 1BM19CH038

LAB PROGRAM 10

```
import java.awt.*;
import java.awt.event.*;

public class Dir extends Frame implements ActionListener{
    TextField t1,t2;
    String msg = " ";
    Button btn;
    Dir(){

    Label l1 = new label("First Number:",
        lab.RIGHT);
    t1 = new TextField(10);
    Label l2 = new Label("Second Number:",
        Label.Right);
    t2 = new TextField(10);
    btn = new Button("Submit");
    // Label l = new label("Updates:");
    l1.setBackground(Color.YELLOW);
    l2.setBackground(Color.YELLOW);
    // this.setResizable(false);
    this.add(l1);
    this.add(t1);
    this.add(l2);
    this.add(t2);
    // t1.setEchoChar('*');
    // t2.setEchoChar('#');
    this.add(btn, BorderLayout.CENTER);
    this.setVisible(true);
    this.setSize(600,300);
```

```
this.setLayout (new FlowLayout (FlowLayout.CENTER, 20, 10));
//t1.addActionListener (this);
btn.addActionListener (this);
addWindowListener (new MyWindow ());
setBackground (Color.YELLOW);
//System.out.println (BorderLayout.CENTER);
}

@Override
public Insets getInsets () {
        return new Insets (50, 10, 10, 20);
}

@Override
public void actionPerformed (ActionEvent e) {
        String st1 = t1.getText ();
        String st2 = t2.getText ();
        double n1, n2;
        n1 = 0.0;
        n2 = 0.0;
        if (st1.equals ("") || st2.equals ("")) {
            msg = "Text elements cannot be left
                    blank!";
        } else {
    try {
        n1 = Double.parseDouble (st1);
        n2 = Double.parseDouble (st2);
        try {
                double res = n1/n2;
                msg = "Result of division : " + res;
    } catch (ArithmeticException e1) {
                msg = e1.toString ();
        }
```

```
} catch (NumberFormatException e2) {
    msg = "Only numbers accepted as text
          elements!";
  }
}

new MyDialog (this, "Result Dialog", false, msg,
n1, n2);
}

public static void main (String[] args) {
      new Div();
  }
}

class MyDialog extends Dialog implements ActionListener {

public MyDialog (Frame owner, String title, boolean modal,
 String msg, double n1, double n2) {
    super (owner, title, modal);
    this.setVisible (true);
    this.setSize (300, 400);
    this.setLayout (new FlowLayout());
    //System.out.println (owner);
    Label l1 = new Label ("  RESULT: ");
    //l1.setSize (300, 20);
   this.add (l1);
    this.add (new Label ("First Number: " + n1));
    this.add (new Label ("Second Number: " + n2));
    this.add (new Label (msg));

    Button b = new Button ("Close");
    this.add (b);
    b.addActionListener (this);
   this.addWindowListener (new WindowAdapter () {
     public void windowClosing (WindowEvent e) {
```

```
            dispose();
        }
    });
}

@Override
public void actionPerformed(ActionEvent e) {
        dispose();
    }
}

class MyWindow extends WindowAdapter {
    public void windowClosing(WindowEvent e){
        System.exit(0);
    }
}
```

LAB PROGRAM 10

QUESTION: Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

CODE:

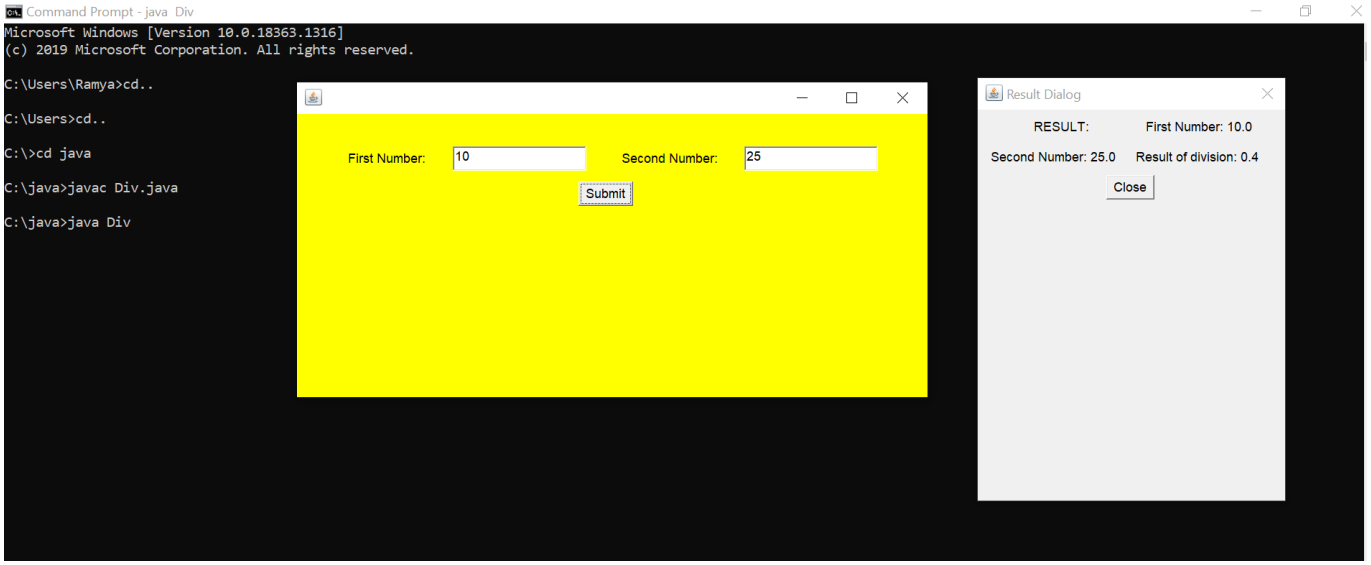
```
import java.awt.*;

import java.awt.event.*;


public class Div extends Frame implements ActionListener{

        TextField t1,t2;

        String msg="";

        Button btn;

        Div(){

                Label l1 = new Label("First Number: ",Label.RIGHT);

                t1 = new TextField(10);

                Label l2 = new Label("Second Number: ",Label.RIGHT);


                t2 = new TextField(10);

                btn = new Button("Submit");

                //Label l = new Label("Updates:");

                l1.setBackground(Color.YELLOW);

                l2.setBackground(Color.YELLOW);

                //this.setResizable(false);

                this.add(l1);

                this.add(t1);

                this.add(l2);

                this.add(t2);

                //the following command will make sure that the input char is not visible to the user

                //(it has been added just to demonstrate). Can be used for passwords.

                //t1.setEchoChar('*');
```

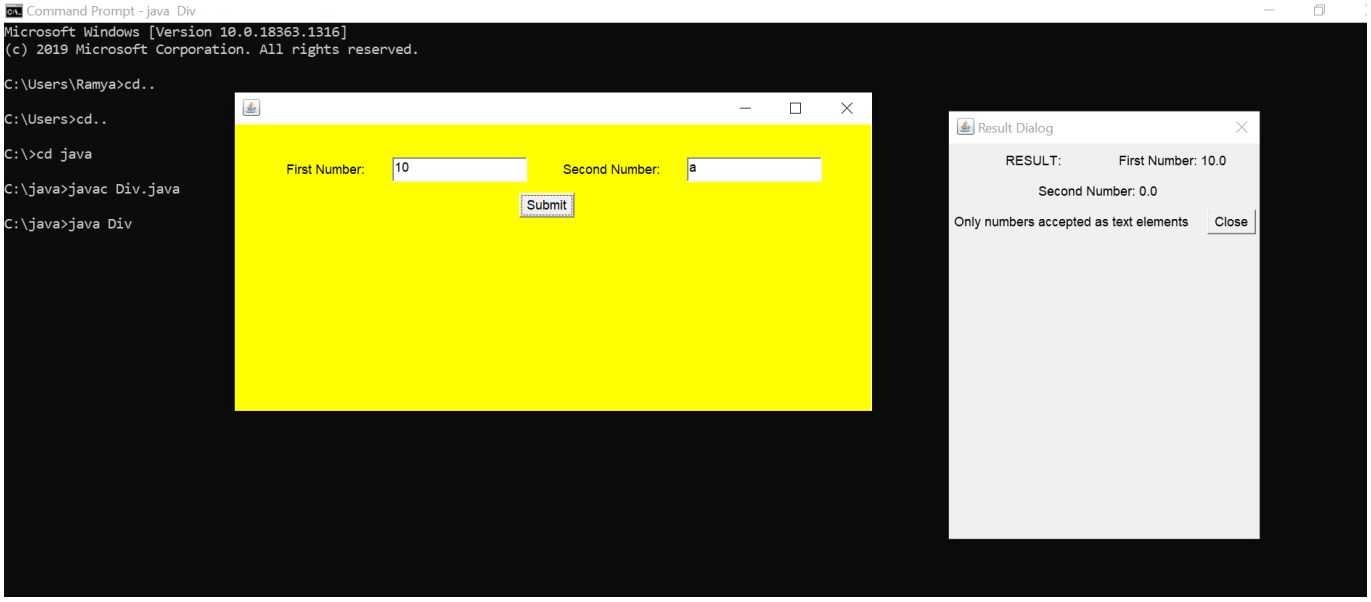
```
		//t2.setEchoChar('#');
		this.add(btn,BorderLayout.CENTER);
		this.setVisible(true);
		this.setSize(600, 300);
		this.setLayout(new FlowLayout(FlowLayout.CENTER,20,10));
		//t1.addActionListener(this);
		btn.addActionListener(this);
		addWindowListener(new MyWindow());
		setBackground(Color.YELLOW);
		//System.out.println(BorderLayout.CENTER);
}
@Override
public Insets getInsets() {
		return new Insets(50,10,10,20);

}

@Override
public void actionPerformed(ActionEvent e) {

		String st1 = t1.getText();
		String st2 = t2.getText();
		double n1,n2;
		n1 = 0.0;
		n2 = 0.0;
		if(st1.equals("")||st2.equals("")) {

			msg="Text elements cannot be left blank!";
		}else{
			try {
				n1 = Double.parseDouble(st1);
```

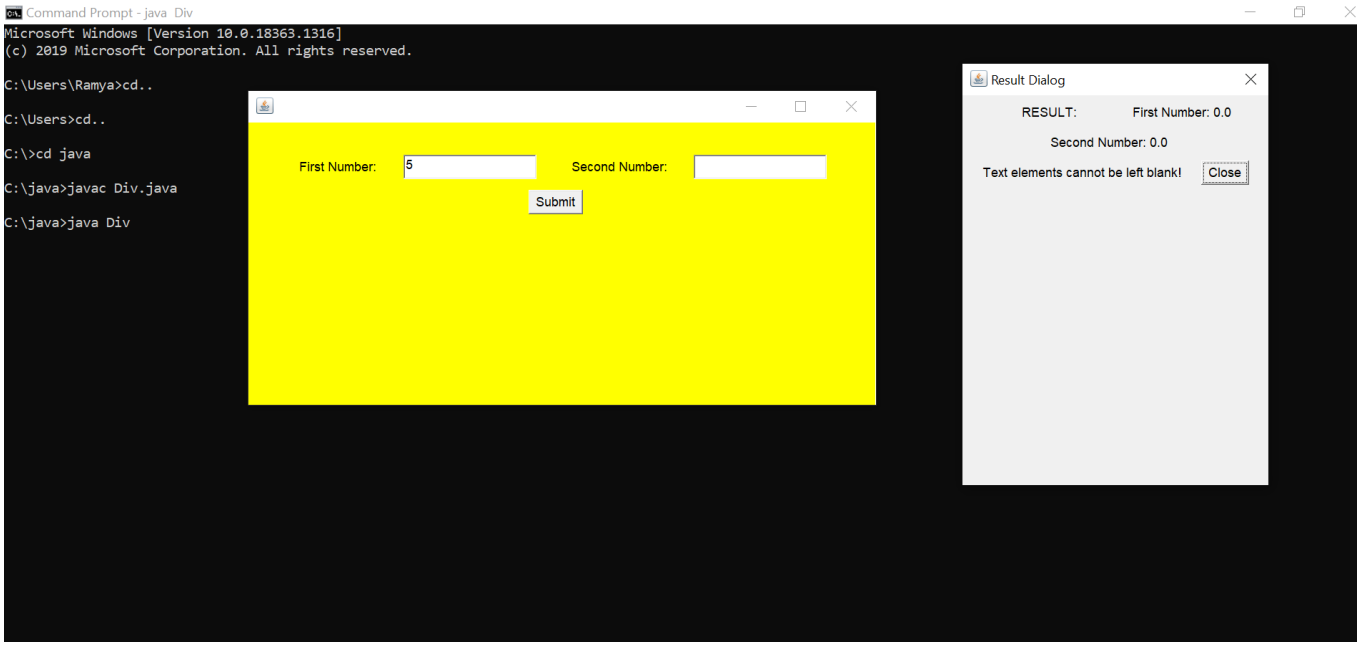
```
                    n2 = Double.parseDouble(st2);
                    try {
                        double res = n1/n2;
                        msg = "Result of division: "+res;
                    }catch(ArithmeticException e1) {
                        msg = e1.toString();
                    }
                }catch(NumberFormatException e2) {
                    msg = "Only numbers accepted as text elements";
                }
            }
            new MyDialog(this,"Result Dialog",false,msg,n1,n2);
        }
        public static void main(String[] args) {
            new Div();
        }
}

class MyDialog extends Dialog implements ActionListener{

        public MyDialog(Frame owner, String title, boolean modal,String msg, double n1, double n2)
{
            super(owner, title, modal);
            this.setVisible(true);
            this.setSize(300, 400);
            this.setLayout(new FlowLayout());
            //System.out.println(owner);
            Label l1 = new Label("        RESULT:        ");
            //l1.setSize(300, 20);
            this.add(l1);
            this.add(new Label("First Number: "+n1));
```

```
                this.add(new Label("Second Number: "+n2));
                this.add(new Label(msg));

                Button b = new Button("Close");
                this.add(b);
                b.addActionListener(this);
                this.addWindowListener(new WindowAdapter() {
                        public void windowClosing(WindowEvent e) {
                                dispose();
                        }
                });
        }

        @Override
        public void actionPerformed(ActionEvent e) {
                dispose();

        }


}

class MyWindow extends WindowAdapter{
        public void windowClosing(WindowEvent e) {
                System.exit(0);
        }
}
```

OUTPUT:


Command Prompt - java Div
Microsoft Windows [Version 10.0.18363.1316]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Div.java
C:\java>java Div
First Number: 10
Second Number: 25
Submit
Result Dialog
RESULT: First Number: 10.0
Second Number: 25.0 Result of division: 0.4
Close


Command Prompt - java Div
Microsoft Windows [Version 10.0.18363.1316]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Div.java
C:\java>java Div
First Number: 10
Second Number: a
Submit
Result Dialog
RESULT: First Number: 10.0
Second Number: 0.0
Only numbers accepted as text elements
Close


Command Prompt - java Div
Microsoft Windows [Version 10.0.18363.1316]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Ramya>cd..
C:\Users>cd..
C:\>cd java
C:\java>javac Div.java
C:\java>java Div
First Number: 5
Second Number:
Submit
Result Dialog
RESULT: First Number: 0.0
Second Number: 0.0
Text elements cannot be left blank!
Close