Name: SaiPraveen Marni

USN: 1BM19CS138

Dept: CSE

Section: C

Lab *batch*: C-2

Lab Program-*6*(*20*/1*1*/2020)

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.

20/11/2020

LAB-6

SAI PRAVEEN MARNI
1BM19CS138

PACKAGE LAB PROGRAM

"student.java"

```
Package cie ;
import java.util.* ;
Public class student
{
   Public String usn ;
   public String name ;
   public int sem ;
   Public void read()
 {
   Scanner sc = new Scanner (System.in);
   System.out.println ("Enter usn of student :");
     usn = sc.next();
   system.out.println ("Enter name of student :");
     name = sc.next();
   System.out.print ("Enter semester :");
     sem = sc.nextInt();
 }
}
```

"internals.java"

```
Package cie ;
import java.util.* ;
Public class internals extends student
  {
     Public int[] cie_m = new int[5] ;
      Public void read()
    {  super.read();
    Scanner sc = new Scanner (System.in);
    System.out.println("Enter CIE marks :").
```

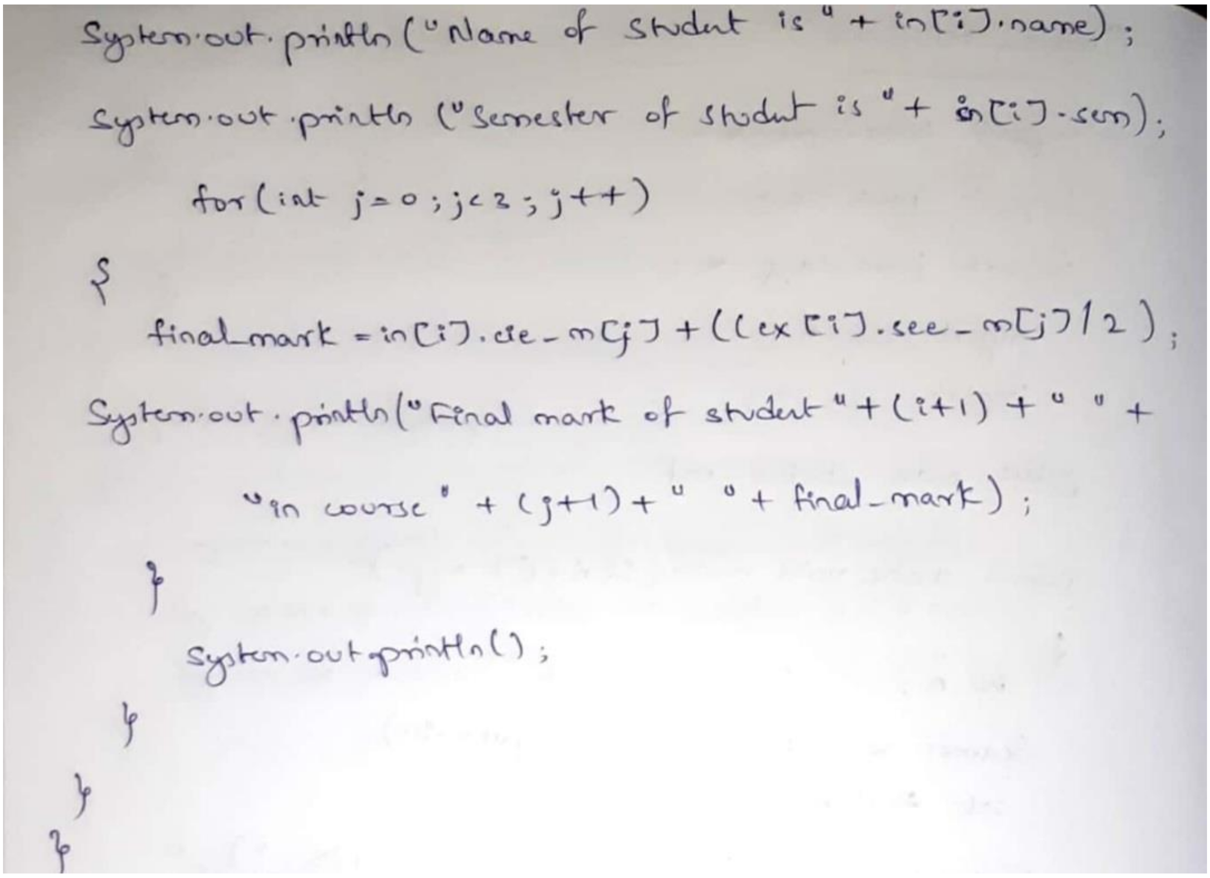
```
        for (int i=0;i<3;i++)
  {
   System.out.println("Enter marks of the course "+(i+1)+":
        cie_m[i] = sc.nextInt();
   }
}
  public void display()
 {
    System.out.println("USN of student is "+usn);
    System.out.println("Name of student is "+name);
    System.out.println("Semester of student is "+sem);
  }
}
```

"externals.java"

```
   package see;
   import java.util.*;
   import java.io;
   import java.lang.*;
   Public class external extends cie.student
   {
       Public int[] see_m = new int[3];
         public int[] mar;
         public void read()
     {
      Scannee sc = new Scannee(system.in);
     System.out.println("Entee SEE malks :");
       for(int i=0;i<3;i++)
      {
        System.out.print("Enter SEE marks of the course"
                              +(i+1)+": ");
           see_m[i] = sc.nextInt();
  } }
```

"main_stu.java"

```
import java.util.* ;
import java.io ;
import java.lang.* ;
 import cie.* ;
 import see.* ;
public class student_end
{
public static void main (string[] args)
 {
   int n ;
  Scanner sc = new Scanner (System.in);
   int final_mark ;
  System.out.print ("Enter number of students : ");
     n = sc.nextInt() ;
    internals[] in = new internals [n] ;
     external[] ex = new external[n] ;
     internals ob1 = new internals () ;
      external ob2 = new external () ;
       ob2 .mar = new int[n] ;
    for(int i=0; i<n ; i++ )
{
System.out.println ("Enter details of student " +(i+1)+ ": ");
       in[i] = new internals () ;
        in[i].read() ;
       ex[i] = new external () ;
        ex[i].read() ;
}
   System.out.println() ;
      for(int i=0 ;i<n ; i++)
  {  System.out.println("Details of Student " + (i+1)) ;
     System.out.println ("USN of student is" + in[i].usn)
```



```
System.out.println("Name of Student is " + in[i].name);
System.out.println("Semester of Student is "+ in[i].sem);
    for(int j=0;j<3;j++)
{
  final_mark = in[i].cie_m[j] + ((ex[i].see_m[j]/2);
System.out.println("Final mark of student "+(i+1) + " " +
        "in course " + (j+1)+ " " + final_mark);
  }
   System.out.println();
  }
 }
}
```

Student.java

```
packa
ge
cie;
       import java.util.*;
       public class student
       {
            public String usn;
            public String name;
            public int sem;

            public void read()
            {
                 Scanner sc=new Scanner(System.in);
            System.out.print("Enter usn of the student : ");
                 usn=sc.next();
            System.out.print("Enter name of the student : ");
                 name=sc.next();
       System.out.print("Enter semester of the student : ");
                 sem=sc.nextInt();
            }
       }
```

Internals.java

```
 packa
 ge
 cie;
        import java.util.*;

        public class internals extends student
        {
             public int[] cie_m=new int[3];

             public void read()
             {
                  super.read();
                 Scanner sc=new Scanner(System.in);
        System.out.println("Enter the CIE marks : ");
                 for(int i=0;i<3;i++)
                 {
        System.out.print("Enter marks of the course " +
        (i+1)+": ");
                  cie_m[i]=sc.nextInt();
                 }
             }
             public void display()
             {
        System.out.println("USN of the student is " +
        usn);
        System.out.println("Name of the stuednt is " +
        name);
        System.out.println("Semester of the student is "
        + sem);
        }
        }
```

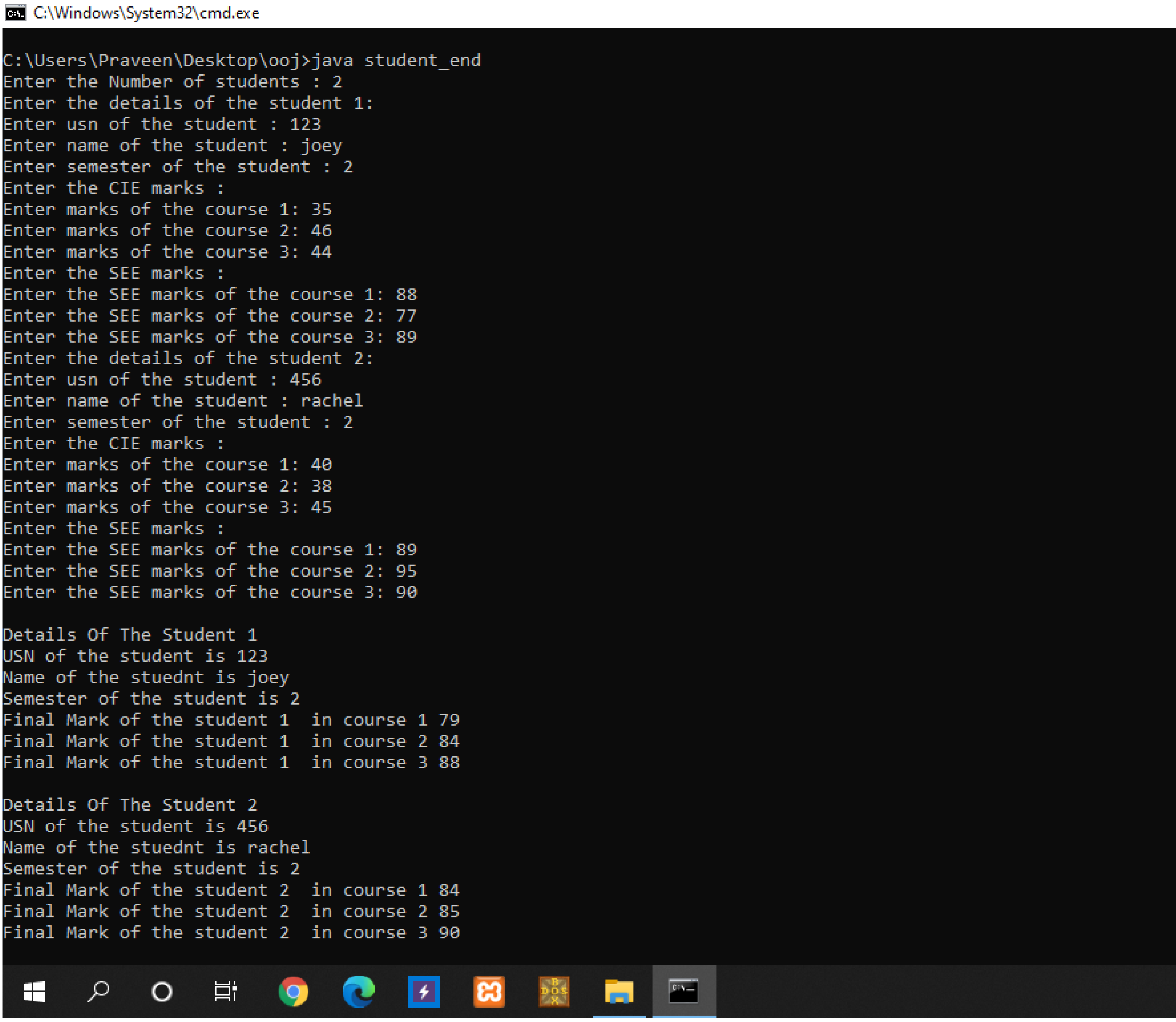
Externals.java

```
package
see;
        import java.util.*;
        import java.io.*;

        import java.lang.*;

        public class external extends
        cie.student
        {
             public int[] see_m=new int[3];
             public int[] mar;
             public void read()
             {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the SEE
        marks : ");
                  for(int i=0;i<3;i++)
                  {
        System.out.print("Enter the SEE marks
        of the course " + (i+1)+": ");
        see_m[i]=sc.nextInt();
                  }
             }
        }
```

Main_stu.java

```
 Import
 java.util
 .*;
           import java.io.*;
           import java.lang.*;
           import cie.*;
           import see.*;


           public class student_end
           {
           public static void main(String[]
           args)
                {
                     int n;
           Scanner sc=new Scanner(System.in);
                     int final_mark;
           System.out.print("Enter the Number
           of students : ");
           n=sc.nextInt();
internals[] in=new internals[n];
external[] ex=new external[n];
internals ob1=new internals();
external ob2=new external();
           ob2.mar=new int[n];

           for(int i=0;i<n;i++)
           {
System.out.println("Enter the
details of the student " + (i+1)+":
");
      in[i]=new internals();
      in[i].read();
      ex[i]=new external();
      ex[i].read();
           }
    System.out.println();
           for(int i=0;i<n;i++)
           {
System.out.println("Details Of The
Student " + (i+1));
System.out.println("USN of the
student is " + in[i].usn);

System.out.println("Name of the
stuednt is " + in[i].name);

System.out.println("Semester of the
student is " + in[i].sem);
for(int j=0;j<3;j++)
                {
final_mark=in[i].cie_m[j]+((ex[i].s
ee_m[j])/2);

System.out.println("Final Mark of
the student " + (i+1) + " " + " in
course " + (j+1) + " " +
final_mark);

                }
```

Output:



C:\Windows\System32\cmd.exe

```
C:\Users\Praveen\Desktop\ooj>java student_end
Enter the Number of students : 2
Enter the details of the student 1:
Enter usn of the student : 123
Enter name of the student : joey
Enter semester of the student : 2
Enter the CIE marks :
Enter marks of the course 1: 35
Enter marks of the course 2: 46
Enter marks of the course 3: 44
Enter the SEE marks :
Enter the SEE marks of the course 1: 88
Enter the SEE marks of the course 2: 77
Enter the SEE marks of the course 3: 89
Enter the details of the student 2:
Enter usn of the student : 456
Enter name of the student : rachel
Enter semester of the student : 2
Enter the CIE marks :
Enter marks of the course 1: 40
Enter marks of the course 2: 38
Enter marks of the course 3: 45
Enter the SEE marks :
Enter the SEE marks of the course 1: 89
Enter the SEE marks of the course 2: 95
Enter the SEE marks of the course 3: 90

Details Of The Student 1
USN of the student is 123
Name of the stuednt is joey
Semester of the student is 2
Final Mark of the student 1  in course 1 79
Final Mark of the student 1  in course 2 84
Final Mark of the student 1  in course 3 88

Details Of The Student 2
USN of the student is 456
Name of the stuednt is rachel
Semester of the student is 2
Final Mark of the student 2  in course 1 84
Final Mark of the student 2  in course 2 85
Final Mark of the student 2  in course 3 90
```

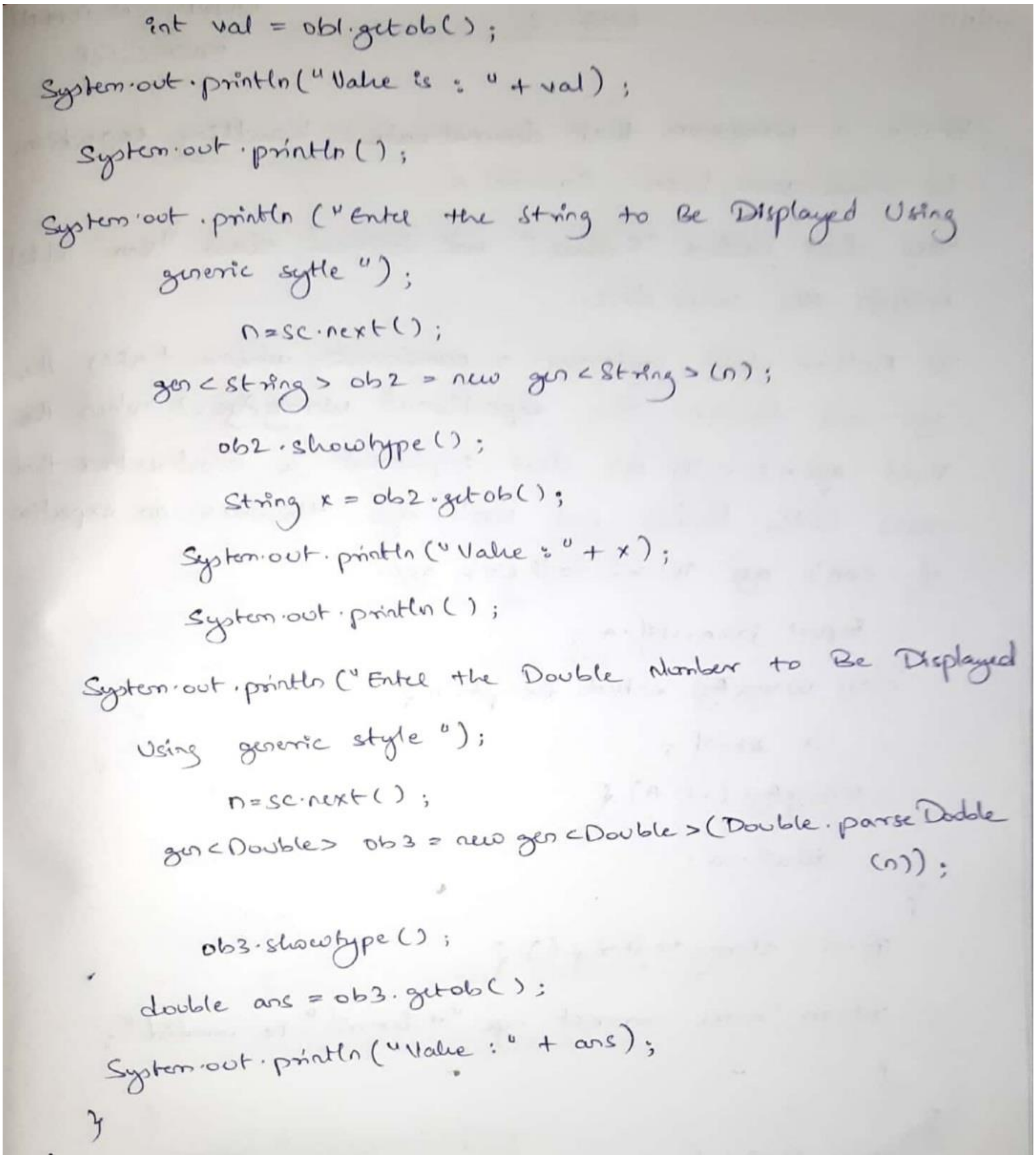
Lab Program-7(*27*/11/2020)

Write a program to demonstrate generics with multiple object parameters.

27/11/20 LAB - 7 SAIPRAVEEN MARN
1BM19CS138

Write a program to demonstrate generics with multiple object parameters.

```
      import java.io.*;
      import java.lang.*;
      import java.util.*;
   class gen<T>
   {
      T ob;
      gen(T o)
      { ob=o;
     }
      T getob()
     {
        return ob;
     }
      void showtype()
   {
   System.out.println("Type of T is " + ob.getClass().getName())
   }
 }
   class generic
  {
   public void main (String[] args)
  {
    String n;
    Scanner sc = new Scanner (System.in);
   System.out.println ("Enter the Integer number to be
        Displayed using the generic style");
           n=sc.next();
      gen<Integer> ob1 = new gen<Integer> (Integer.parseInt(n));
          ob1.showtype();
```

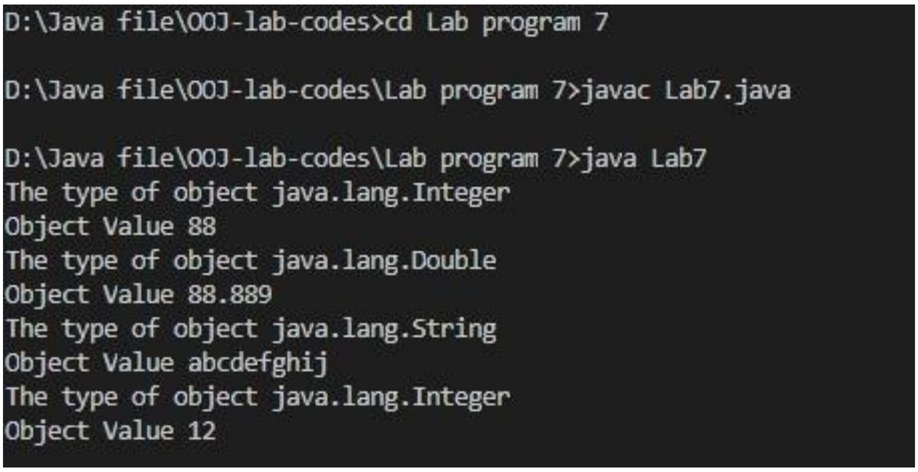


```
      int val = ob1.getob();
System.out.println("Value is : " + val);
   System.out.println();
System.out.println("Enter the String to Be Displayed Using
        generic sytle");
              n=sc.next();
     gen<String> ob2 = new gen<String>(n);
              ob2.showtype();
              String x = ob2.getob();
       System.out.println("Value :" + x);
       System.out.println();
  System.out.println("Enter the Double Number to Be Displayed
     Using generic style");
              n=sc.next();
     gen<Double> ob3 = new gen<Double>(Double.parseDouble
                                            (n));

          ob3.showtype();
     double ans = ob3.getob();
  System.out.println("Value :" + ans);
  }
```

```
importjava.io
.*;
                import java.lang.*;
                import java.util.*;

                class gen<T>
                {
                      T ob;
                      gen(T o)
```

```
     {
          ob=o;
     }
     T getob()
     {
          return ob;
     }
     void showtype()
     {
System.out.println("Type of T is " +
ob.getClass().getName());
     }
}

class generic
{
     public static void main(String[] args)
     {
          String n;
          Scanner sc=new Scanner(System.in);
          System.out.println("Enter the Integer
Number to Be Displayed Using the generic style");
          n=sc.next();
          gen<Integer> ob1=new
gen<Integer>(Integer.parseInt(n));
          ob1.showtype();
          int val=ob1.getob();
          System.out.println("Value is: " + val);

          System.out.println();

          System.out.println("Enter the String to
Be Displayed Using the generic style");
          n=sc.next();
          gen<String> ob2=new gen<String>(n);
          ob2.showtype();
          String x=ob2.getob();
          System.out.println("Value : " + x);

          System.out.println();


System.out.println("Enter the Double Number to Be
Displayed Using the generic style");
n=sc.next();
gen<Double> ob3=new
gen<Double>(Double.parseDouble(n));
ob3.showtype();
double ans=ob3.getob();
System.out.println("Value : " + ans);

}
}
```

Output:



```
D:\Java file\OOJ-lab-codes>cd Lab program 7

D:\Java file\OOJ-lab-codes\Lab program 7>javac Lab7.java

D:\Java file\OOJ-lab-codes\Lab program 7>java Lab7
The type of object java.lang.Integer
Object Value 88
The type of object java.lang.Double
Object Value 88.889
The type of object java.lang.String
Object Value abcdefghij
The type of object java.lang.Integer
Object Value 12
```

Lab Program-8(*27*/11/2020)

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.

27/11/20

LAB - 8

SAIPRAVEEN MARNI
1BM19CS138

Write a program that demonstrates handling expections in inheritance tree. Create a

base class called "Father" and derived class "Son" which extends the base dass.

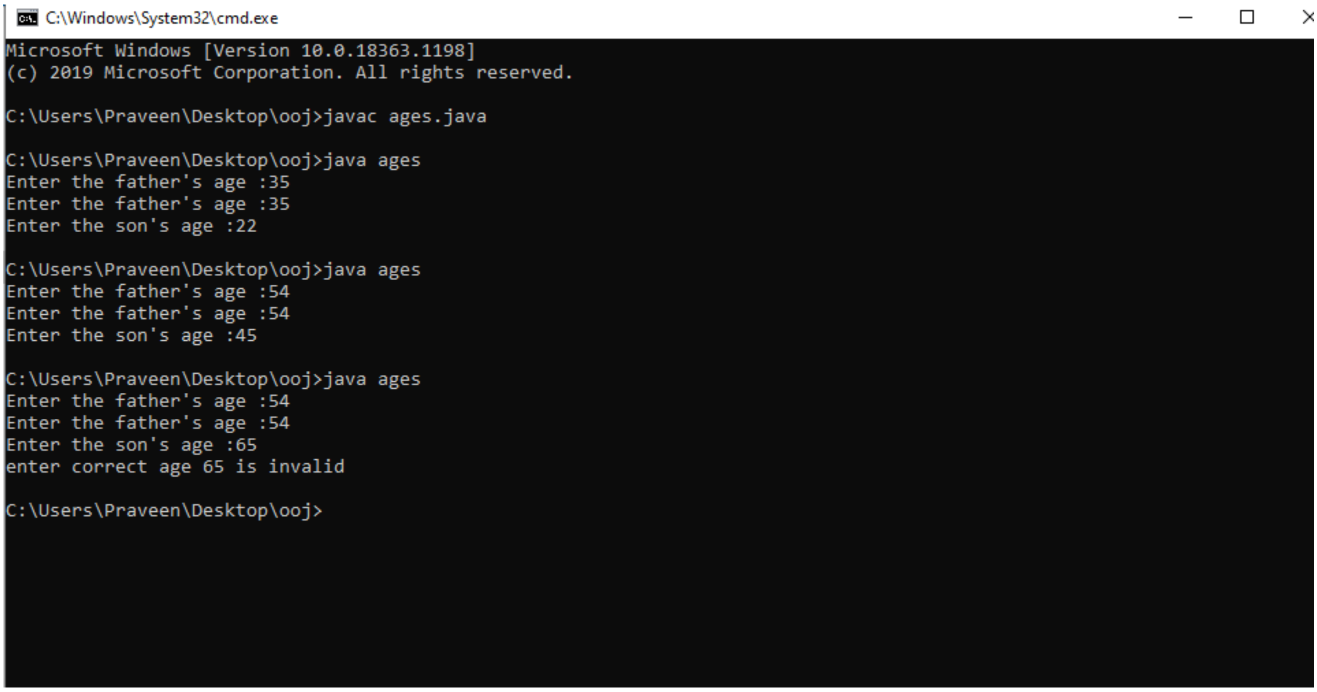
In Father class, implement a construct which takes the age and throws the expectional WrongAge() when the input age < 0. In son class, implement a constructor that cases both father and soo's age throws an expection if son's age is >= father's age.

```
    import java.util.* ;
  class WrongAge extends Exception {
      int detail ;
    WrongAge (int A) {
       detail = a ;
  }
    public string toString () {
    return "enter correct age "+ detail " is invalid ";
    }
 }
   class father {
      Public int age ;
   Scanner in = new Scanner (System.in) ;
    father () throws WrongAge {
  System.out.print ("Enter Father's age :");
       age = in.nextInt () ;
       if ( age < 0 )
    }   throws new WrongAge (age) ;
```

```
class son extends father{
Scanner in = new Scanner(System.in);
    int fage;
   son(father f) throws WrongAge {
    this.fage = f.age;
 System.out.print("Enter son's age :");
    this.age = in.nextInt();
      if (this.age < 0)
     throw new WrongAge(age);
     if (this.age > f.age)
    throw new WrongAge(age);
 }
}
  class ages{
  public static void main(String args[]) {
     try{
        father f = new father();
         son s = new son(f);
   }
    catch(Exception e){
    System.out.println(e);
   }
  }
}
```

```
importjava.util.*;
                    class WrongAge extends Exception {
                     int detail;
                     WrongAge(int a) {
                      detail = a;
                     }
                     public String toString() {
                      return "enter correct age "+detail+" is
                    invalid";
                     }

                    }
                    class father{
                     public int age;
                     Scanner in =new Scanner(System.in);
                     father() throws WrongAge{
                    System.out.print("Enter the father's age :");
                      age= in.nextInt();
                      if(age<0)
                        throw new WrongAge(age);
                     }
                    }
                    class son extends father{
                     Scanner in =new Scanner(System.in);
                     int fage;
                     son(father f) throws WrongAge{
                      this.fage=f.age;
                      System.out.print("Enter the son's age :");
                      this.age= in.nextInt();
                      if(this.age<0)
                        throw new WrongAge(age);
                      if(this.age>f.age)
                        throw new WrongAge(age);
                     }

                    }
                    class ages{
                     public static void main(String[] args){
                      try{
                       father f= new father();
                       son s= new son(f);
                       }
                      catch(Exception e){
                        System.out.println(e);
                     }
                     }
```

Output:



```
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Praveen\Desktop\ooj>javac ages.java

C:\Users\Praveen\Desktop\ooj>java ages
Enter the father's age :35
Enter the father's age :35
Enter the son's age :22

C:\Users\Praveen\Desktop\ooj>java ages
Enter the father's age :54
Enter the father's age :54
Enter the son's age :45

C:\Users\Praveen\Desktop\ooj>java ages
Enter the father's age :54
Enter the father's age :54
Enter the son's age :65
enter correct age 65 is invalid

C:\Users\Praveen\Desktop\ooj>
```

Lab Program-9(*11*/12/2020)

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.

1/12/20

LAB-9

SAI PRAVEEN MARNI
1BM19CS138

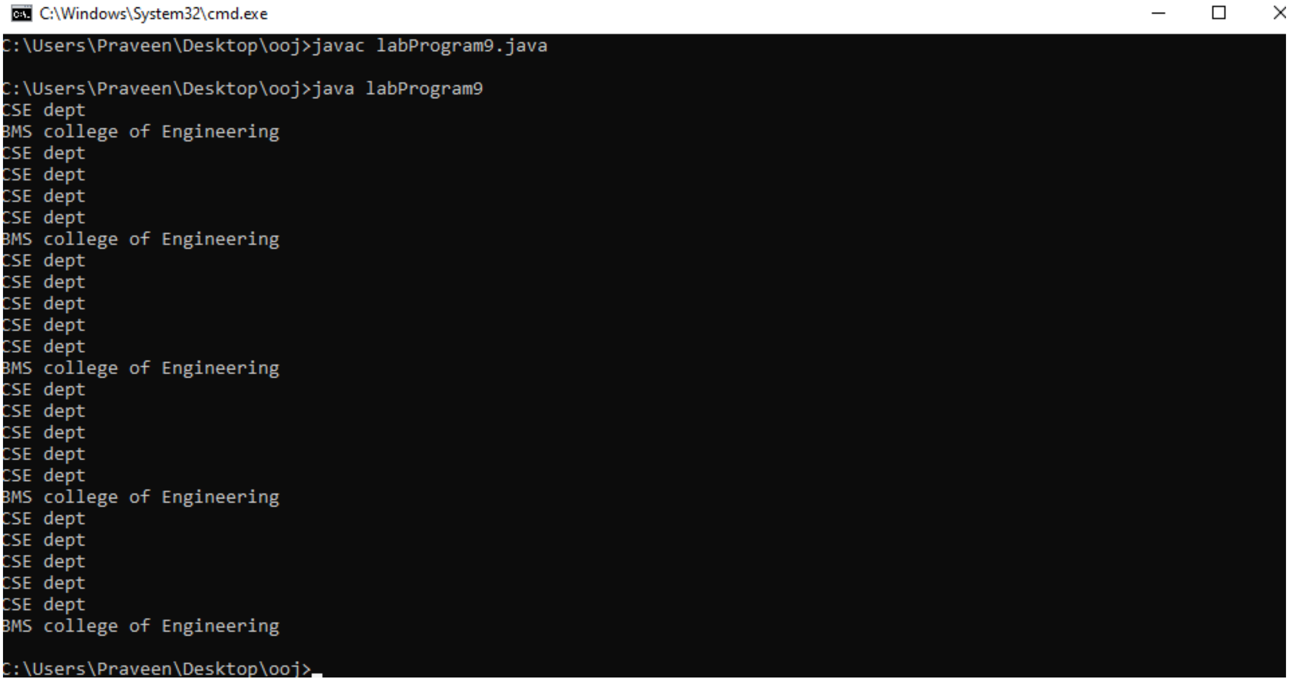
Write a program which creates two threads, one thread displaying "BMS college of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

```
class Thread1 implements Runnable {
    Thread t ;
    String name ;
   Thread1 (String name) {
      this.name = name ;
     t = new Thread (this. this.name);
      t.start ;
  }
    public void run() {
      try {
         for(int i=0 ; i<20 ; i++) {
         System.out.println ("CSE dept ");
            Thread.sleep(2000) ;
        }
      } catch (InterruptedException e) {
            System.out.println(e) ;
        }
      }
   }
     Class Thread2 implements Runnable {
            Thread t ;
            String name ;
        Thread2 (String name) {
           this.name = name ;
          t = new Thread (this , this.name) ;
            t.start();
        }
    Public void run() {
        try{
            for(int i=0;i<5;i++)
  System.out.println("Bms college of Engineering");
          Thread.sleep(10000);
      }
   } catch (InterruptedException e) {
        System.out.println(e);
   }
  }
}
 class labProgram9 {
   Public static void main(String[] args) {
  Thread1 obj1 = new Thread1("Dept.name");
  Thread2 obj2 = new Thread2 ("College name");
//System.out.println(obj1.name + " " + obj1.t.isAlive());
//System.out.println (obj2.name + " " + obj2.t.isAlive());
    try{
         obj1.t.join();
         obj2.t.join();
    } catch(Exception e) {
     System.out.println ("Interrupted");
    }
  }
}
```

```
class Thread1
implementsRunnable{
                         Thread t;
                         String name;
                         Thread1(String name){
                             this.name = name;
                             t = new Thread(this,this.name);
                             t.start();
                         }
                         public void run(){
                             try{
                                 for(int i=0;i<20;i++){
                                 System.out.println("CSE dept");
                                     Thread.sleep(2000);
                                 }
                             }
                     catch(InterruptedException e){
                                 System.out.println(e);
                             }
                         }
                     }

                     class Thread2 implements Runnable{
                         Thread t;
                         String name;
                         Thread2(String name){
                             this.name = name;
                             t = new Thread(this,this.name);
                             t.start();
                         }
                         public void run(){
                             try{
                                 for(int i=0;i<5;i++){
                      System.out.println("BMS college of
                     Engineering");
                                     Thread.sleep(10000);
                                 }
                             }
                     catch(InterruptedException e){
                                 System.out.println(e);
                             }
                         }
                     }

                     class labProgram9{
                         public static void main(String[] args){
```

```
           Thread1 obj1 = new Thread1("Dept. name");
         Thread2 obj2 = new Thread2("College name");
        //System.out.println(obj1.name+"
        "+obj1.t.isAlive());
        //System.out.println(obj2.name+"
        "+obj2.t.isAlive());
                try{
                    obj1.t.join();
                    obj2.t.join();
                }
        catch(Exception e){

        System.out.println("Interrupted");
                }
            }
        }
```

Output:

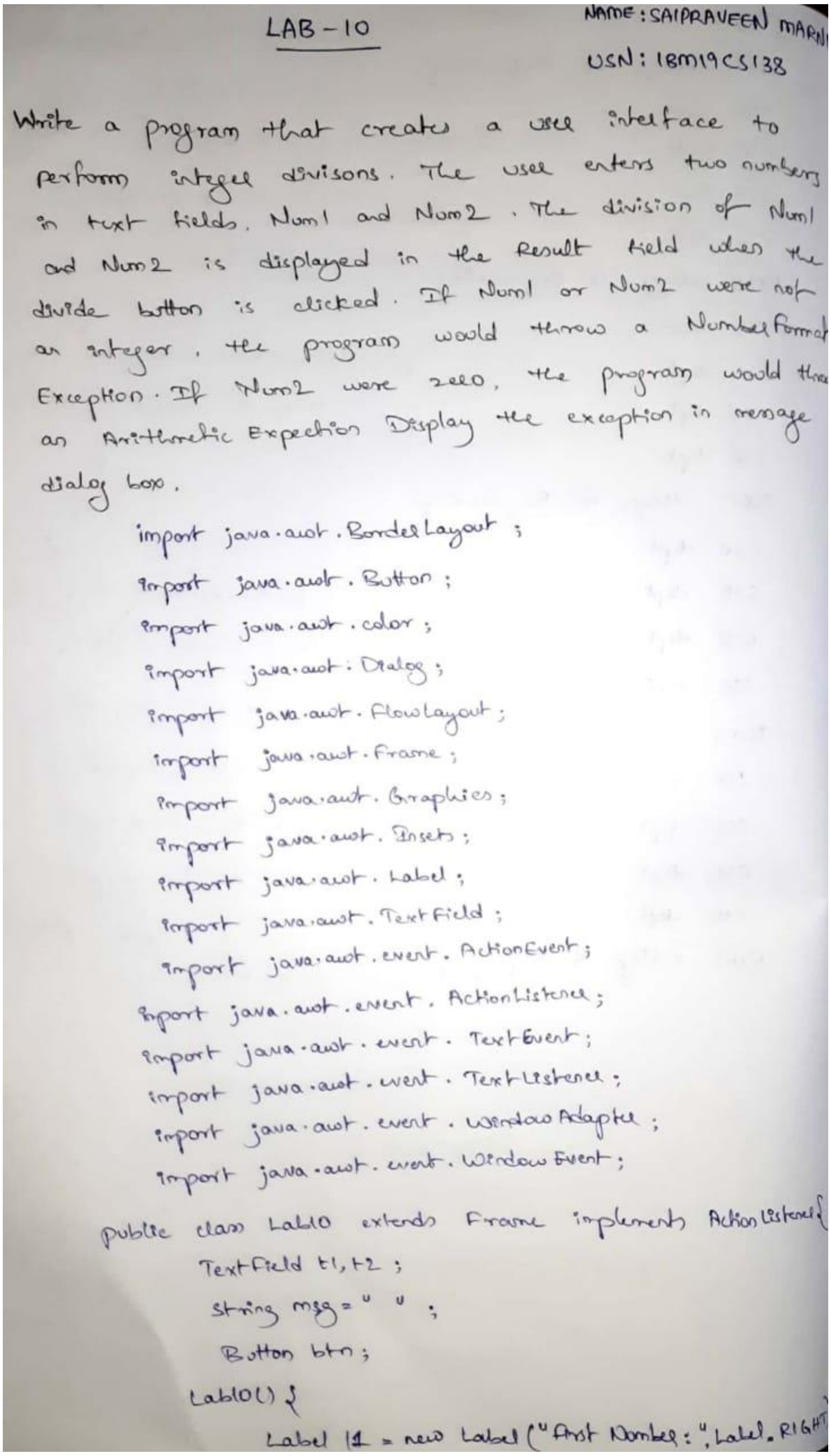


```
C:\Users\Praveen\Desktop\ooj>javac labProgram9.java

C:\Users\Praveen\Desktop\ooj>java labProgram9
CSE dept
BMS college of Engineering
CSE dept
CSE dept
CSE dept
CSE dept
BMS college of Engineering
CSE dept
CSE dept
CSE dept
CSE dept
CSE dept
BMS college of Engineering
CSE dept
CSE dept
CSE dept
CSE dept
CSE dept
BMS college of Engineering
CSE dept
CSE dept
CSE dept
CSE dept
CSE dept
BMS college of Engineering

C:\Users\Praveen\Desktop\ooj>
```

Lab Program-10(*11*/12/2020)

Write a program that creates a user interface to perform integer divisions of Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If

Num1 and 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.



LAB - 10

NAME : SAIPRAVEEN MARNI

USN : 1BM19CS138

Write a program that creates a user interface to perform integer divisons. The user enters two numbers in 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 NumberFormat Exception. If Num2 were zero, the program would throw an Arithmetic Expection Display the exception in message dialog box.

```
        import java.awt.BorderLayout;
        import java.awt.Button;
        import java.awt.color;
        import java.awt.Dialog;
        import java.awt.FlowLayout;
        import java.awt.Frame;
        import java.awt.Graphics;
        import java.awt.Insets;
        import java.awt.Label;
        import java.awt.TextField;
        import java.awt.event.ActionEvent;
        import java.awt.event.ActionListener;
        import java.awt.event.TextEvent;
        import java.awt.event.TextListener;
        import java.awt.event.WindowAdapter;
        import java.awt.event.WindowEvent;
public class Lab10 extends Frame implements ActionListener{
        TextField t1,t2;
        string msg = " ";
        Button btn;
    Lab10(){
            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 -set Background (color.Yellow);
      l2.set Background (color.YELLOW);
     // this.set Risizable (false);
        this.add(l1);
        this.add(t1);
        this.add(l2);
        this.add(t2);
// the following command will make sure that input char is
                  not visible to the user.
// (it has been added just to demostrate). can be used
 for passwoords.
  //t1.set Echochar('*');
  //t2.set EchoChar('#');
    this.add (btn, Borderlayout.CENTER);
     this.set Visible (true);
     this.set Size (600, 300);
 this.setlayout (new Flowlayout (Flowlayout.CENTER,20,10))
    //t1. add ActionListener (this);
    btn. add Action Listener (this);
   add WindowListener (new myWindow());
     set Background (color.YELLOW);
  // System.out.println (Borderlayout.CENTER);
 }
    @Override
     public Insets getInsets() {
            return new Insets (50,10,10,20);
     }
```

```
@ Overside
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 = "You cannot leave the text elements blank"
   } else {
      try {
          n1 = Double.parseDouble (st1) ;
          n2 = Double.parseDouble (st2) ;
        try {
              double res = n1/n2 ;
            msg = "Result of devison :"+res;
        } catch(ArthimeticExpection e1) {
                msg = e1.toString() ;
        }
    } catch (NumberFormatException e2) {
          msg = "Enter only numbers and not other things";
    }
  }
new myDialog( this, "Result Dialog", false, msg, n1, n2);
 }
    public static void main(String[] args) {
            new Lab10();
    }
```

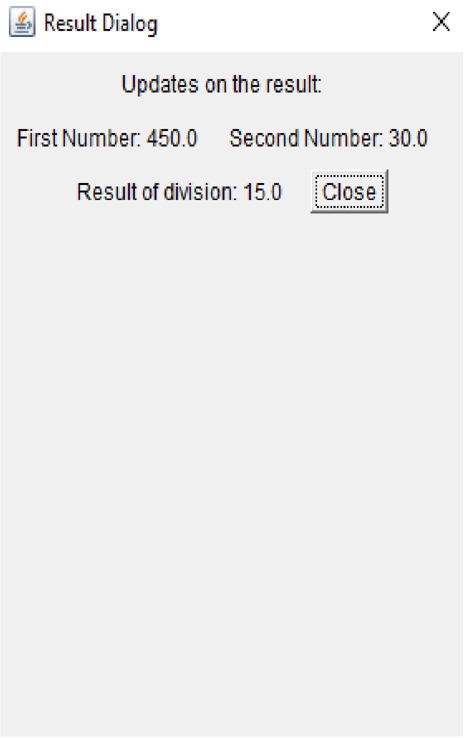
```
class myDialog extends Dialog implements ActionListener {
 public myDialog (Frame owner, String title , boolean modal,
                              String msg , double n1, double n2)
 {
      super (owner , title, modal);
       this . setVisible (true);
        this.set Size (300, 400) ;
      this.set Layout (new FlowLayout ());
    // System.out.println (owner) ;
   Label l1 = new Label (" Updates on the result : ");
      // l1 . set Size (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);
      }
```

```
import
java.awt.BorderLa
yout;
                    import java.awt.Button;
                                   import java.awt.Color;
                                   import java.awt.Dialog;
                                   import java.awt.FlowLayout;
                                   import java.awt.Frame;
                                   import java.awt.Graphics;
                                   import java.awt.Insets;
                                   import java.awt.Label;
                                   import java.awt.TextField;
                                   import
                                   java.awt.event.ActionEvent;
                                   import
                                   java.awt.event.ActionListener;
                                   import java.awt.event.TextEvent;
                                   import
                                   java.awt.event.TextListener;
                                   import
                                   java.awt.event.WindowAdapter;
                                   import
                                   java.awt.event.WindowEvent;

                                   public class Lab10 extends Frame
                                   implements ActionListener{
                                   TextField t1,t2;
                                   String msg="";
                                   Button btn;
                                   Lab10(){
                                   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(BorderLayou
t.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="You cannot leave the text
elements 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 = "Enter only numbers and
not other things";
}
}
New MyDialog(this,"Result
Dialog",false,msg,n1,n2);
}
public static void main(String[]
args) {
new Lab10();
}
}

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(“Updates on
the 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:



Updates on the result:

First Number: 450.0 Second Number: 30.0

Result of division: 15.0