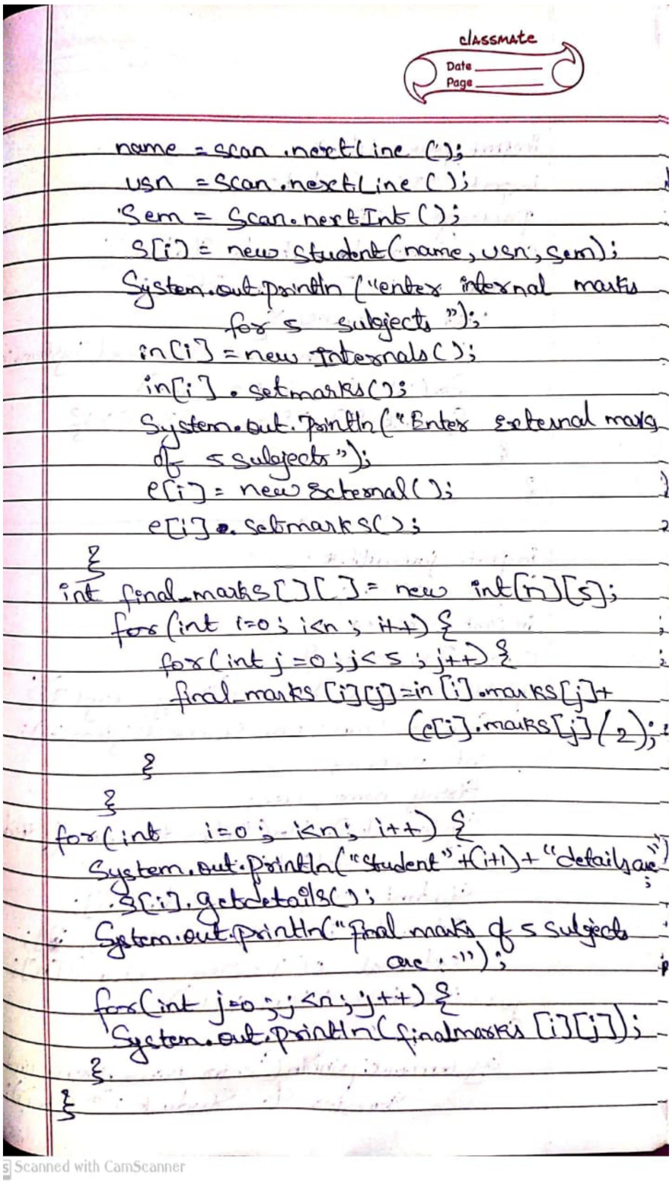
OOJ LAB REPORT

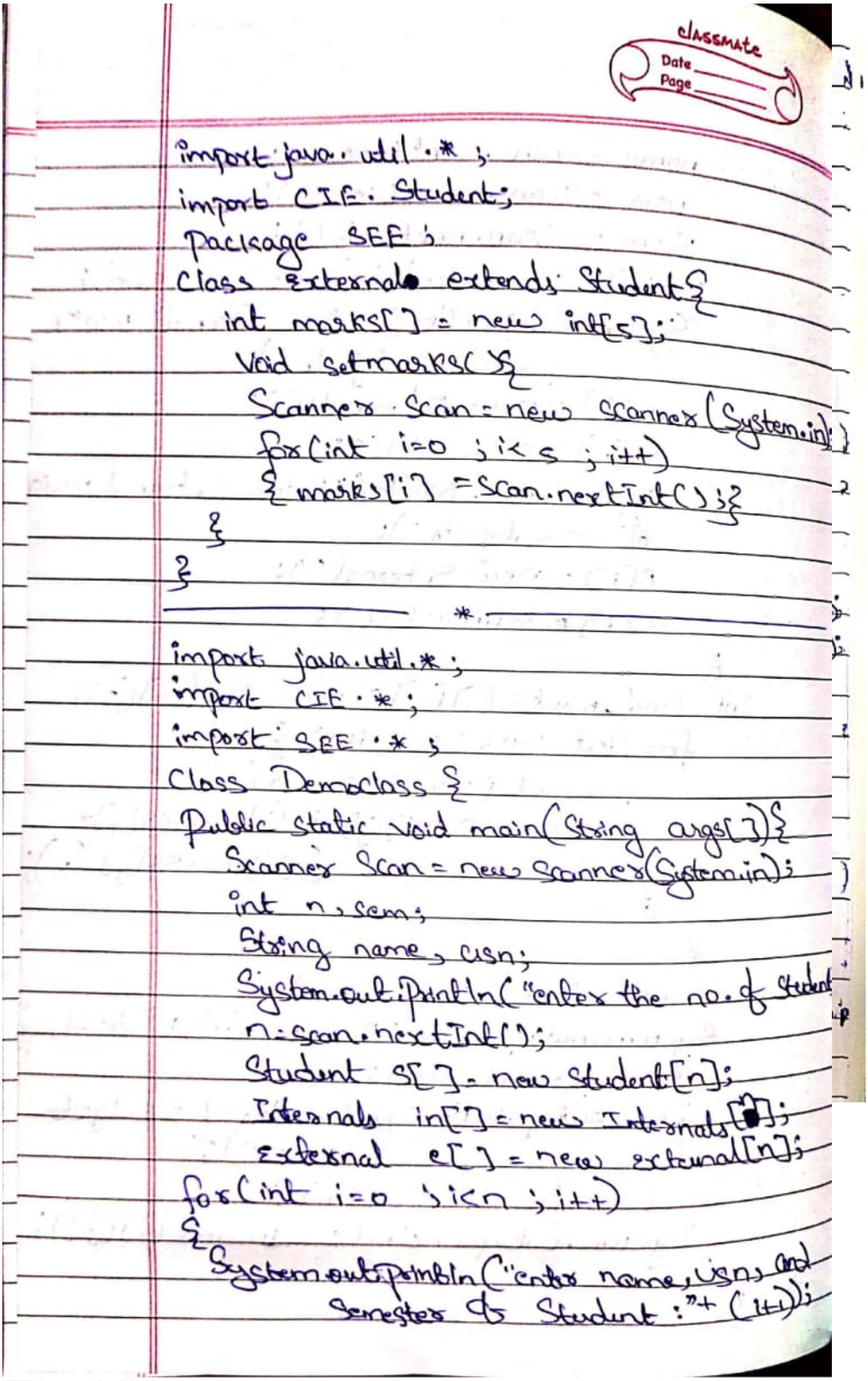
NAME : VASANTH KUMAR T
USN : 1BM19CS180
SECTION : D
SEMESTER : 3rd

LAB 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.

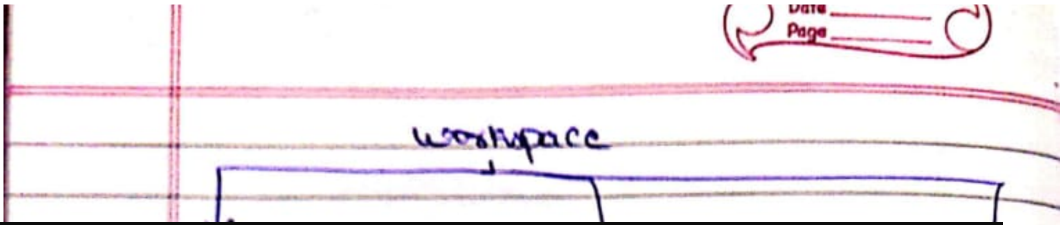


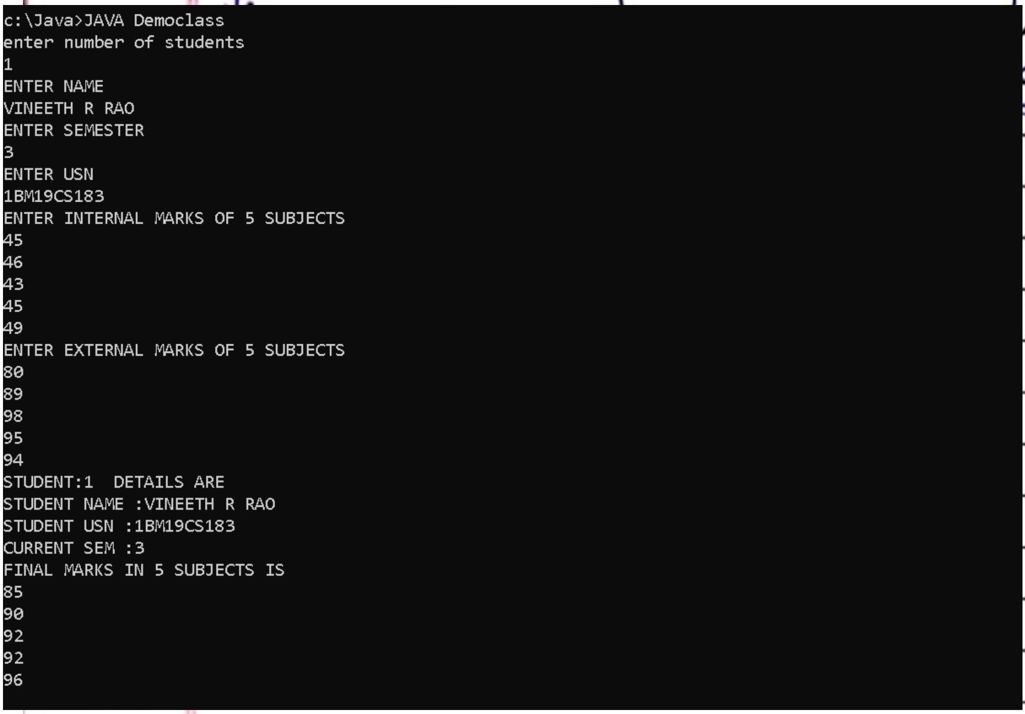
```
      name =scan.nextLine ();
      usn =Scan.nextLine ();
      Sem = Scan.nextInt ();
       S[i] = new Student(name, usn, Sem);
      System.out.println ("enter internal marks
                for 5  subjects ");
       in[i] =new Internals();
       in[i] . setmarks();
       System.out.println("Enter external marks
       of  5 subjects ");
       e[i] = new External();
       e[i]. setmarks();
  }
int final_marks[][]= new int[n][5];
   for(int i=0; i<n; i++) {
      for(int j=0; j<5; j++) {
       final_marks[i][j]=in[i].marks[j]+
                           (e[i].marks[j]/2);
      }
   }
for(int  i=0; i<n; i++) {
  System.out.println("Student"+(i+1)+"details are");
   S[i].getdetails();
  System.out.println("Final marks of 5 subjects
                         are :");
  for(int j=0; j<n; j++) {
   System.out.println(finalmarks [i][j]);
  }.
}
```



```
import java.util.* ;
import CIE. Student;
Package SEE ;
Class External extends Student{
   int marks[] = new int[5];
     Void setmarks( ){
      Scanner Scan = new scanner(System.in);
       for(int i=0 ; i< 5 ; i++)
       { marks[i] =Scan.nextInt();}
    }
}
                          *
import java.util.* ;
import CIE . * ;
import SEE . * ;
Class Democlass {
Public static void main(String args[]){
   Scanner Scan = new Scanner(System.in);
   int n, sem;
   String name, usn;
   System.out.Println("enter the no. of Student
    n=scan.nextInt();
    Student s[]= new Student[n];
    Internals in[] = new Internals[[illegible]];
     External e[] = new external[n];
for(int i=0 ;i<n ; i++)
{
  System.out.println("enter name, usn, and
           Semester of Student :"+ (i+1));
```

Caption





```
c:\Java>JAVA Democlass
enter number of students
1
ENTER NAME
VINEETH R RAO
ENTER SEMESTER
3
ENTER USN
1BM19CS183
ENTER INTERNAL MARKS OF 5 SUBJECTS
45
46
43
45
49
ENTER EXTERNAL MARKS OF 5 SUBJECTS
80
89
98
95
94
STUDENT:1  DETAILS ARE
STUDENT NAME :VINEETH R RAO
STUDENT USN :1BM19CS183
CURRENT SEM :3
FINAL MARKS IN 5 SUBJECTS IS
85
90
92
92
96
```

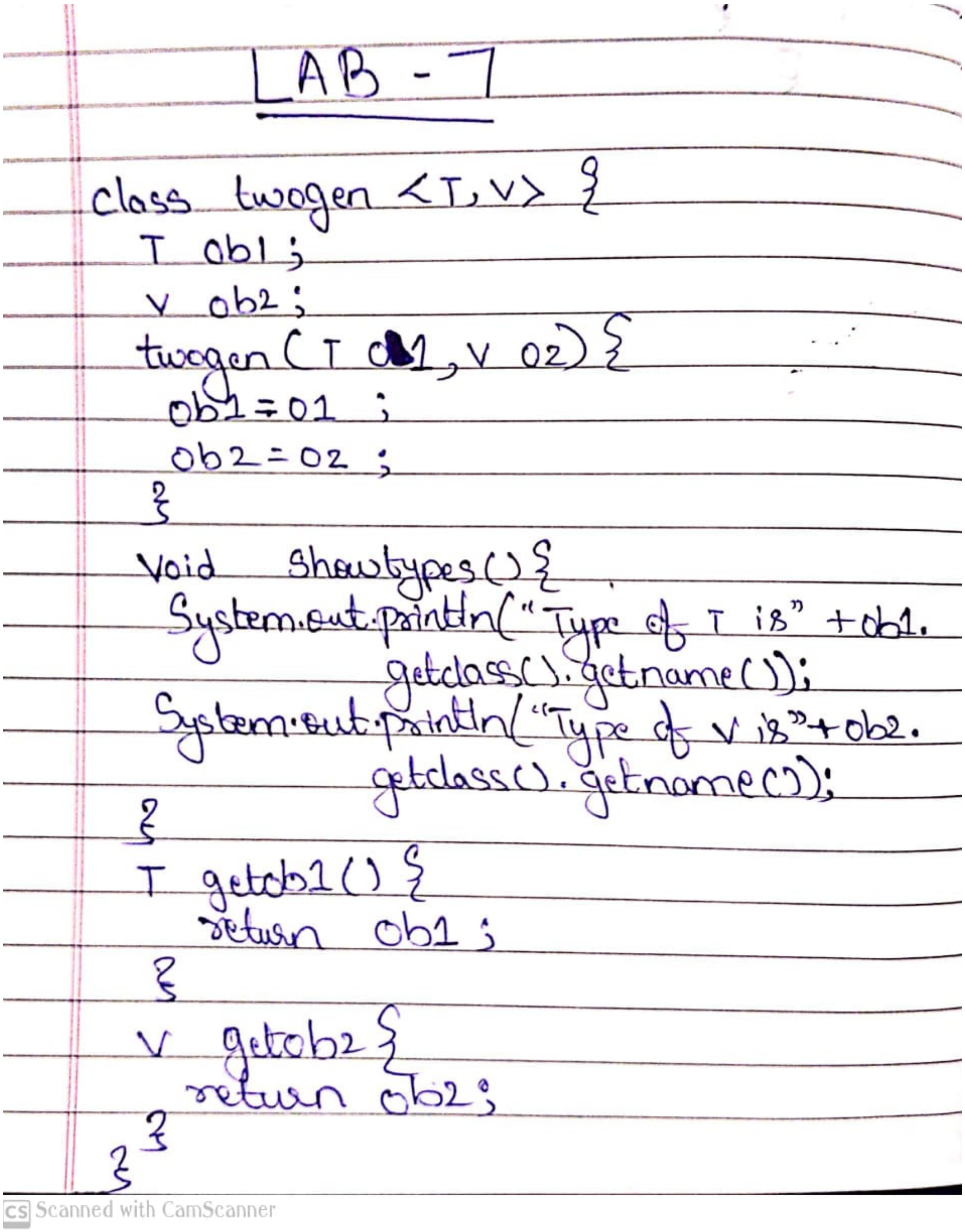
Caption

Caption

OUTPUT

LAB 7

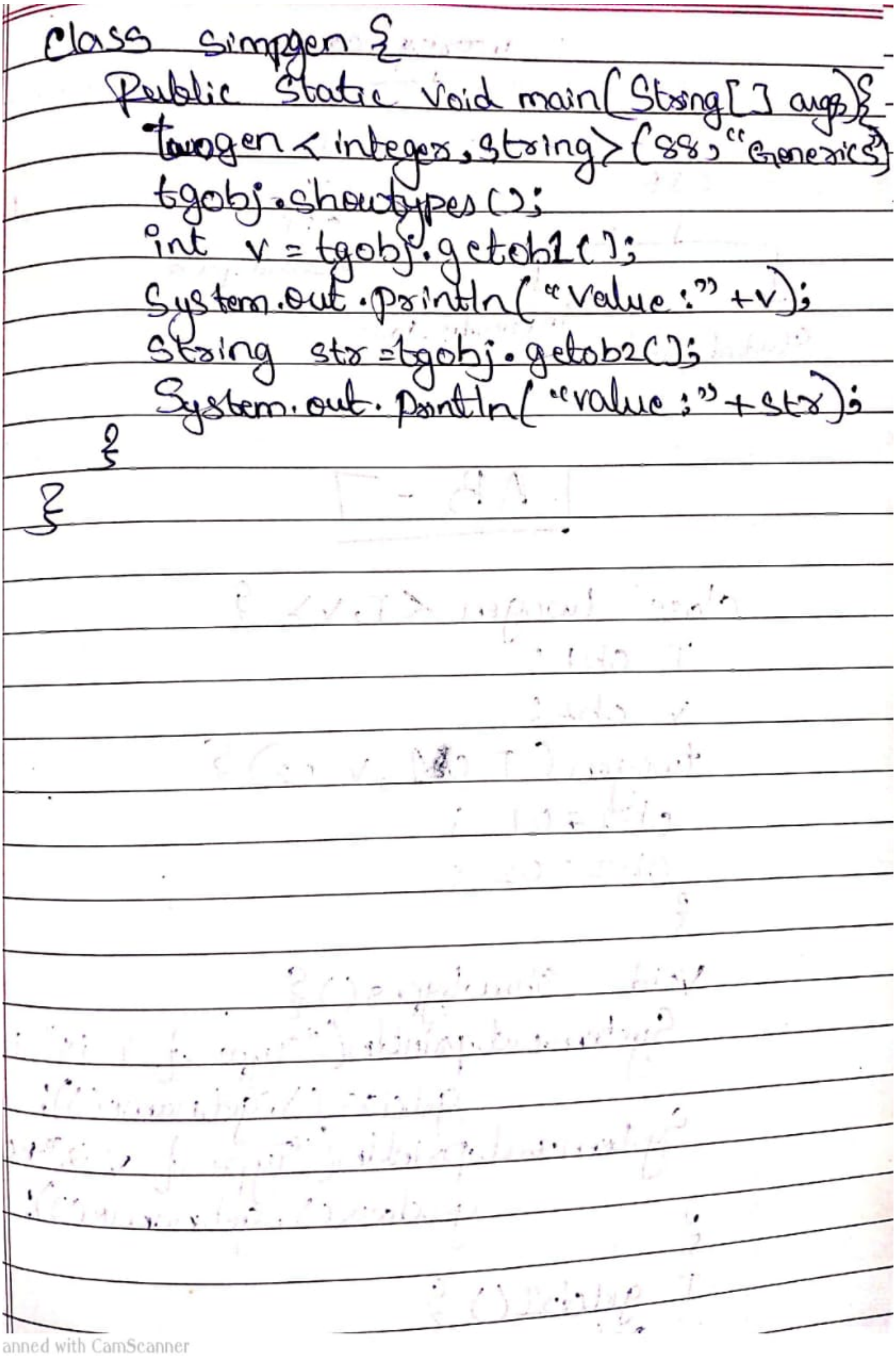
QUESTION : Write a program to demonstrate generics with multiple object parameters.



LAB - 7

```
class twogen <T,V> {
  T ob1;
  V ob2;
  twogen(T o1, V o2){
   ob1=o1 ;
   ob2=o2 ;
  }
  Void Showtypes(){
   System.out.println("Type of T is" +ob1.
              getclass().getname());
   System.out.println("Type of V is"+ob2.
              getclass().getname());
  }
  T getob1() {
    return ob1 ;
  }
  V getob2{
   return ob2;
  }
}
```

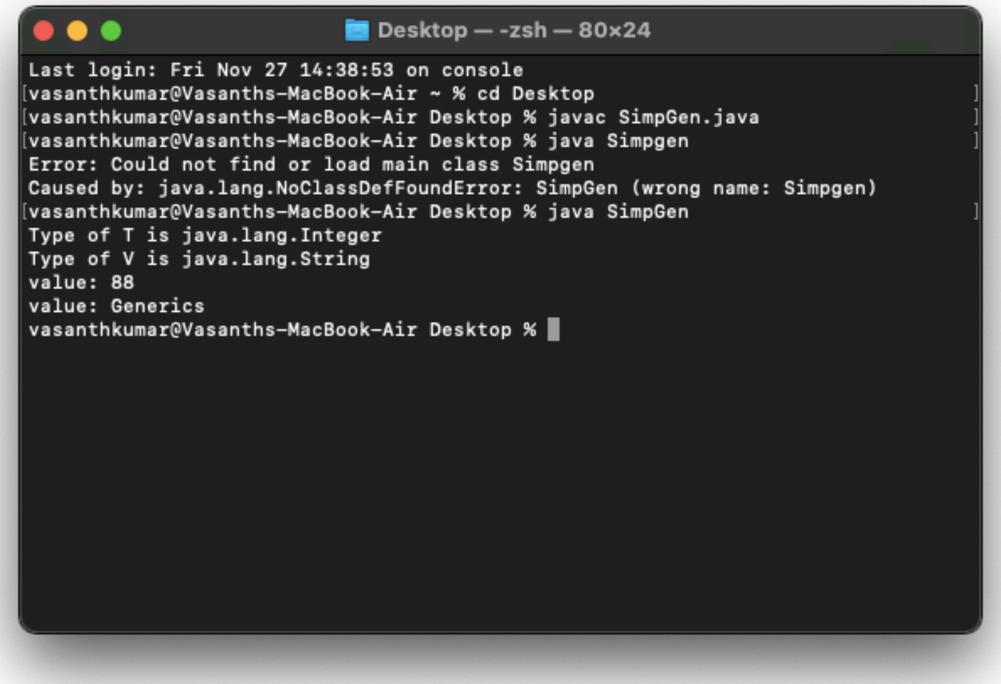
Caption



```
Class simpgen {
   Public Static void main(String[] args){
    twogen< integer, string>(88, "Generics")
    tgobj.showtypes();
    int v = tgobj.getob1();
    System.out.println("value:" +v);
    String str =tgobj.getob2();
    System.out.println("value:"+str);
  }
}
```

Caption

OUTPUT :

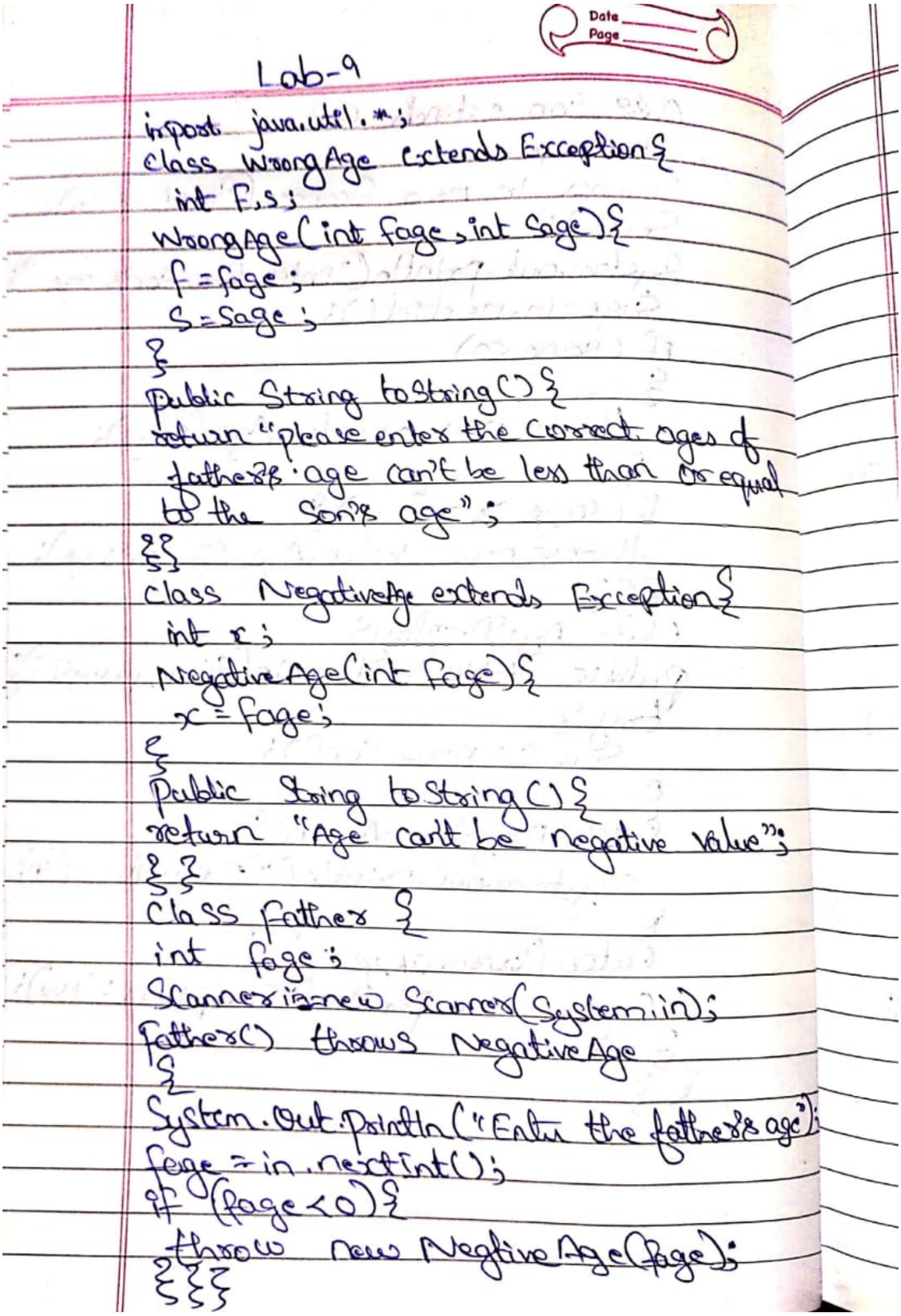


```
Last login: Fri Nov 27 14:38:53 on console
vasanthkumar@Vasanths-MacBook-Air ~ % cd Desktop
vasanthkumar@Vasanths-MacBook-Air Desktop % javac SimpGen.java
vasanthkumar@Vasanths-MacBook-Air Desktop % java Simpgen
Error: Could not find or load main class Simpgen
Caused by: java.lang.NoClassDefFoundError: SimpGen (wrong name: Simpgen)
vasanthkumar@Vasanths-MacBook-Air Desktop % java SimpGen
Type of T is java.lang.Integer
Type of V is java.lang.String
value: 88
value: Generics
vasanthkumar@Vasanths-MacBook-Air Desktop %
```

Caption

LAB 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.



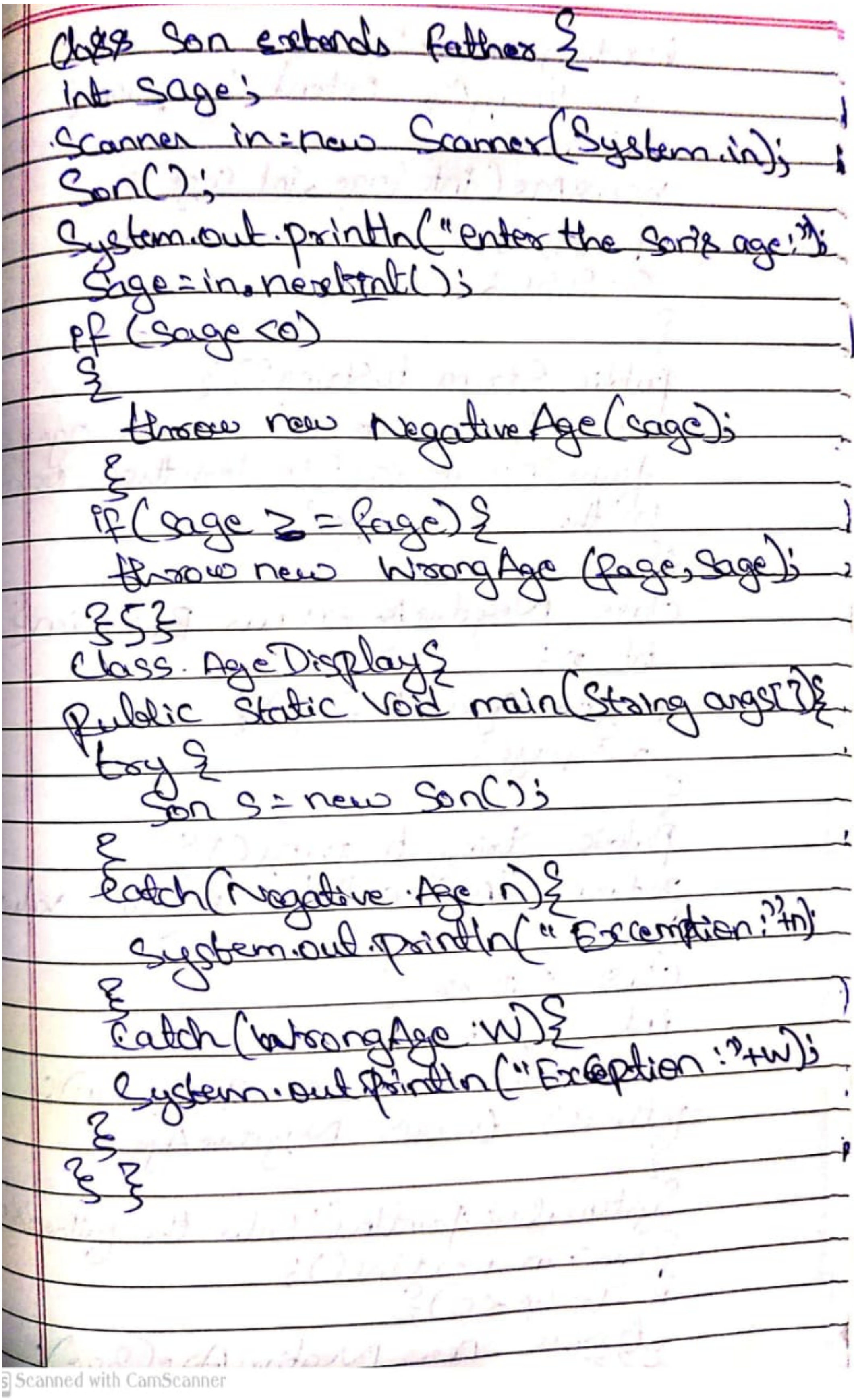
Date

Page

Lab-9

```
import java.util.*;
class WrongAge extends Exception{
 int F,S;
WrongAge(int fage, int sage){
 f=fage;
 S=Sage;
}
Public String toString(){
return "please enter the correct ages of
 father's age can't be less than or equal
to the Son's age";
}}
class NegativeAge extends Exception{
 int x;
NegativeAge(int fage){
 x=fage;
}
Public String toString(){
return "Age can't be negative value";
}}
Class father {
int fage;
Scanner in=new Scanner(System.in);
father() throws NegativeAge
{
System.out.println("Enter the father's age");
fage = in.nextInt();
if (fage<0){
 throw new NegtiveAge(fage);
}}}
```

Caption



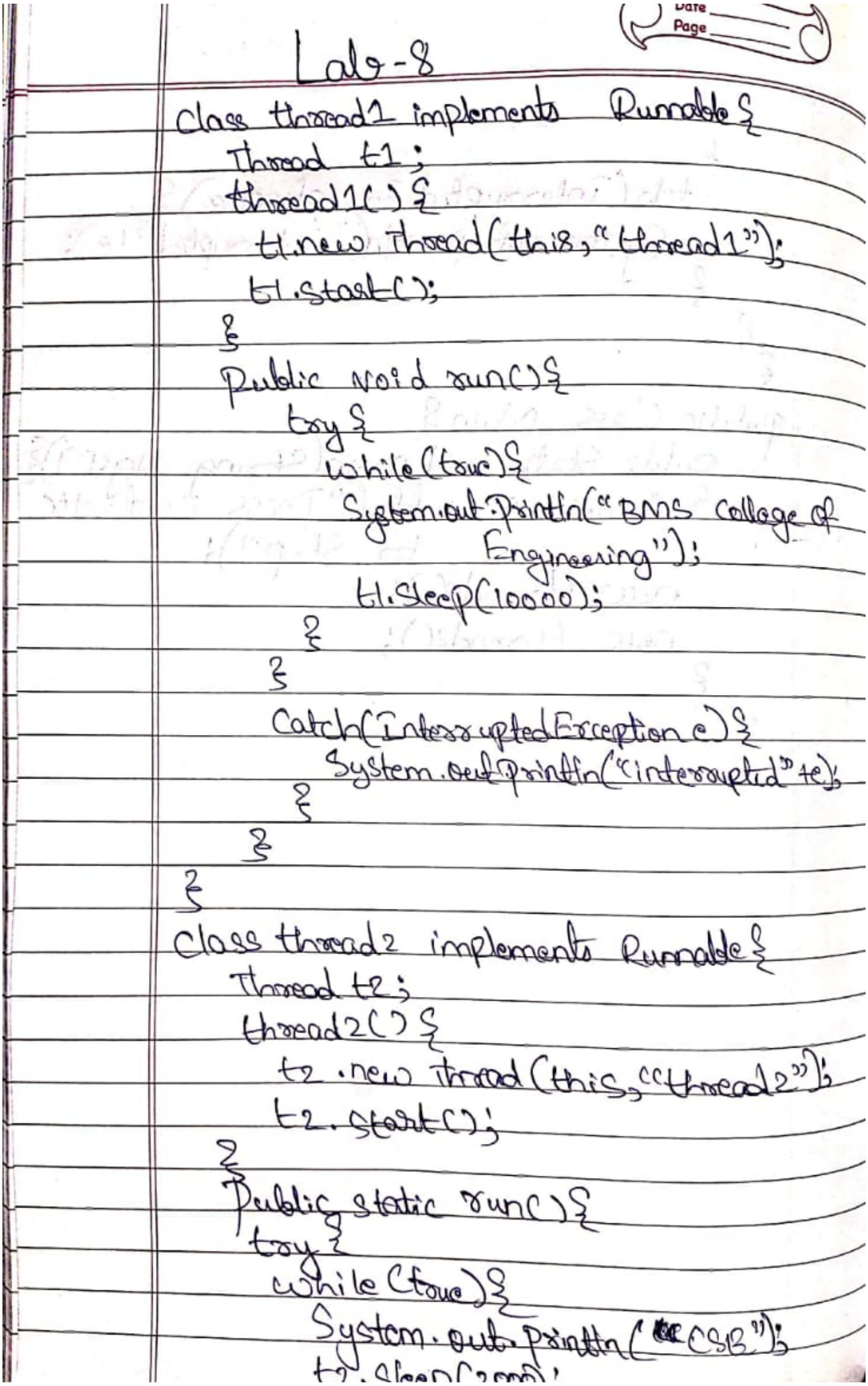
```
Class Son extends father {
int Sage;
Scanner in=new Scanner(System.in);
Son();
System.out.println("enter the son's age:");
 Sage=in.nextInt();
if (sage <0)
{
  throw new NegativeAge(sage);
 }
 if(sage >= fage){
 throw new WrongAge (fage,sage);
}}}
Class. AgeDisplay{
Public static Void main(String args[]){
 try {
   Son s = new Son();
 }
 catch(Negative Age n){
   System.out.println(" Exception:"+n);
 }
 catch(WrongAge w){
 System.out.println("Exception :"+w);
 }
}}
```

Caption

OUTPUT :

LAB 9

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



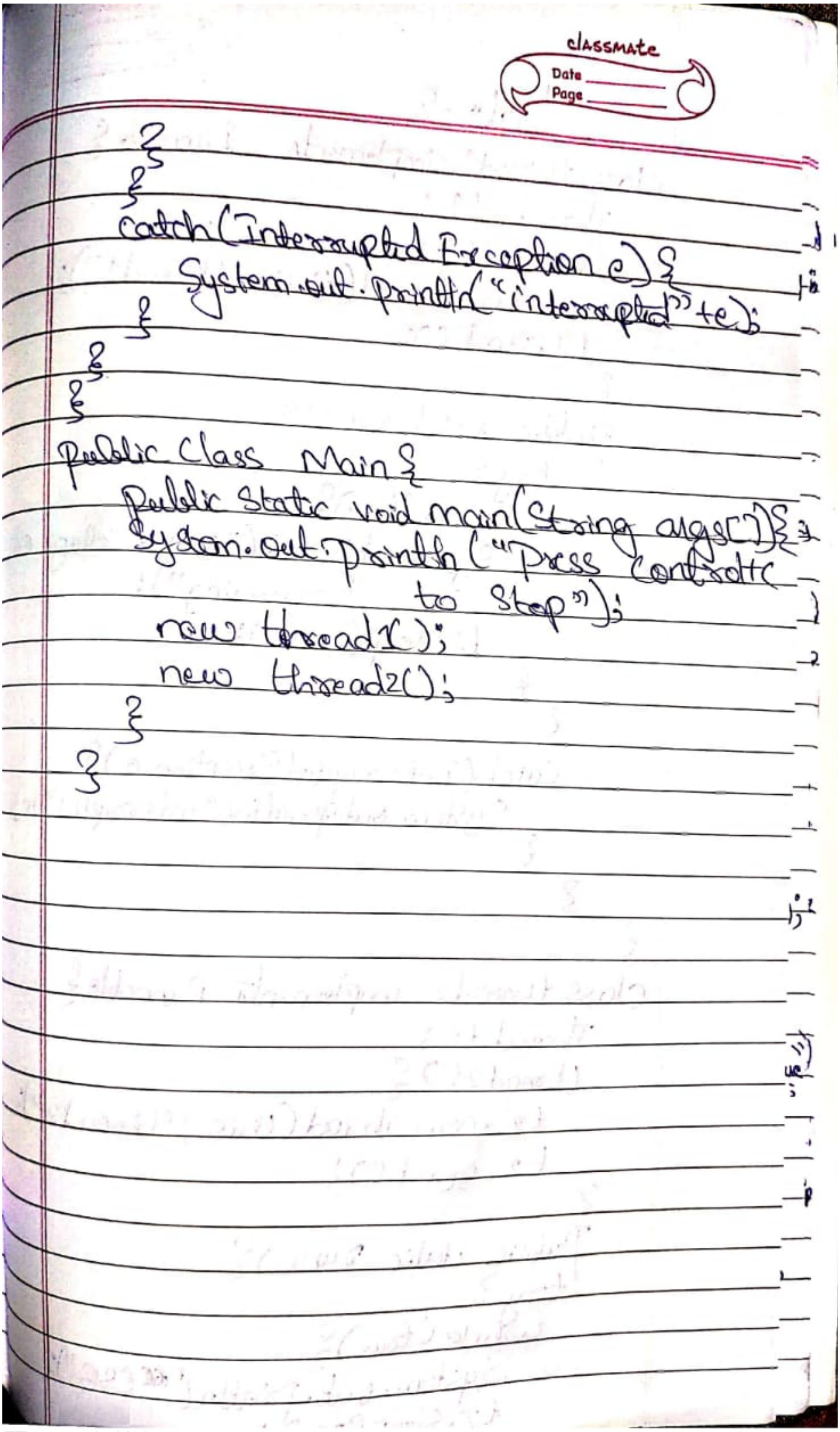
Date

Page

Lab-8

```
Class thread1 implements Runnable{
  Thread t1;
  thread1(){
    t1.new Thread(this,"thread1");
    t1.start();
  }
  Public void run(){
    try{
      while(true){
        System.out.println("BMS College of
                    Engineering");
        t1.sleep(10000);
      }
    }
    Catch(InterruptedException e){
      System.out.println("interrupted"+e);
    }
  }
}
Class thread2 implements Runnable{
  Thread t2;
  thread2(){
    t2.new thread(this,"thread2");
    t2.start();
  }
  Public static run(){
  try{
   while(true){
     System.out.println("CSE");
     t2.sleep(2000);
```

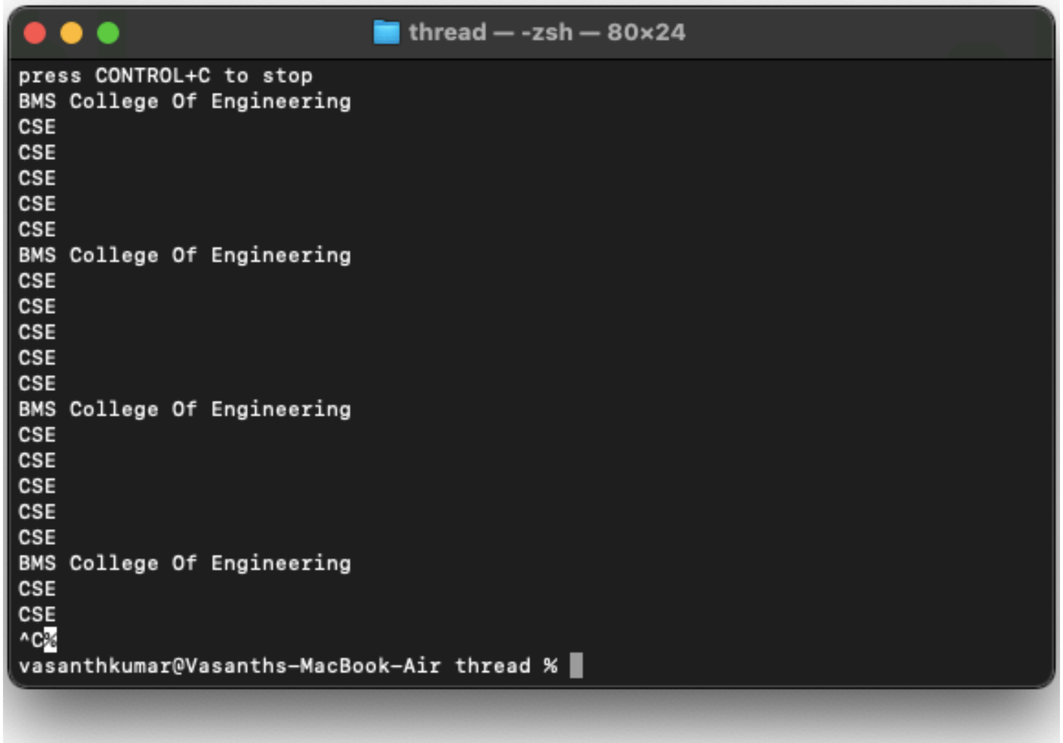
Caption



```
}
}
catch(InterruptedException e){
    System.out.println("interrupted"+e);
}
}
}
public Class Main{
  public static void main(String args[]){
  System.out.println("Press Controlt C
                  to Stop");
   new thread1();
   new thread2();
  }
}
```

Caption

OUTPUT :

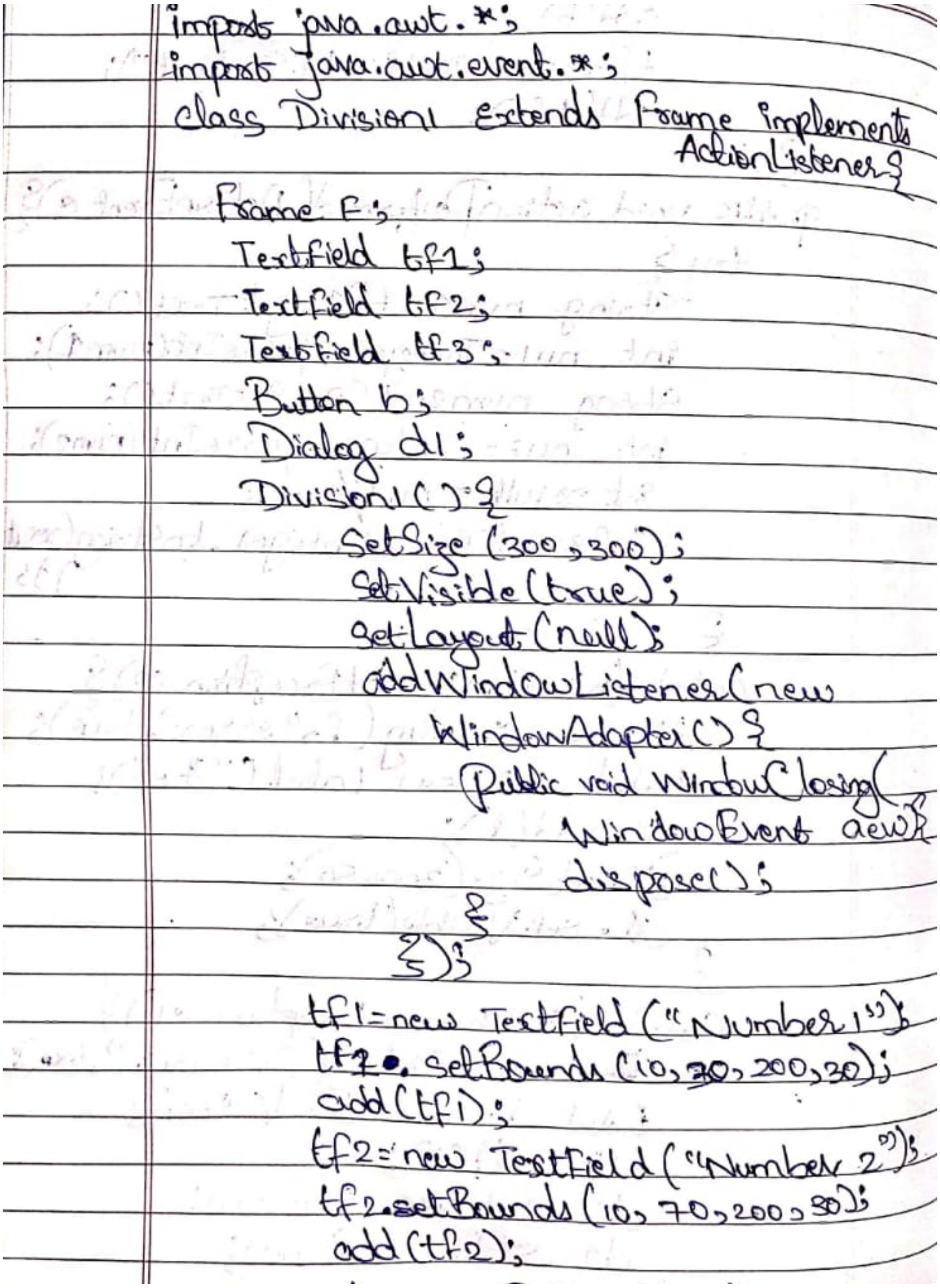


```
press CONTROL+C to stop
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
^C%
vasanthkumar@Vasanths-MacBook-Air thread %
```

Caption

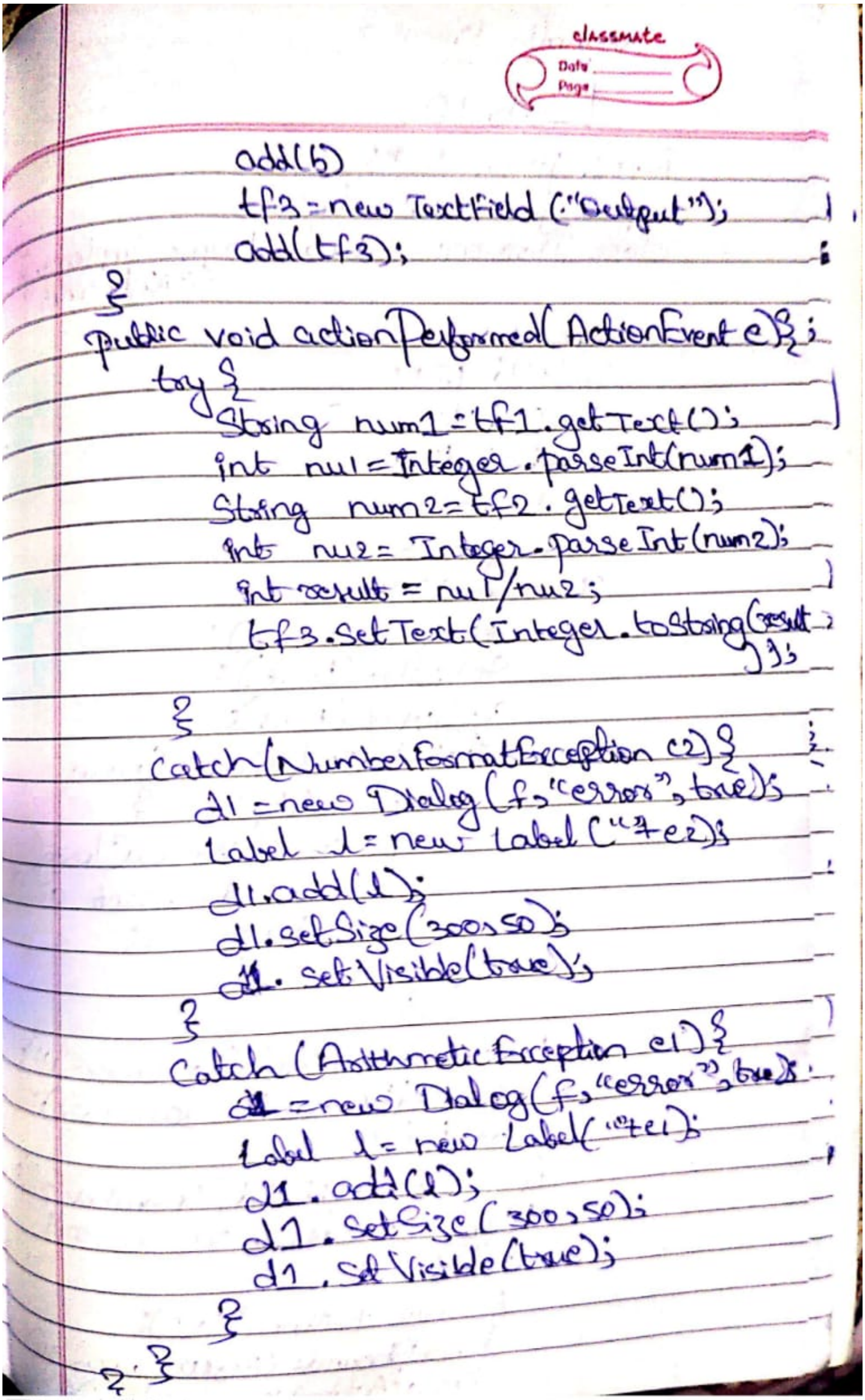
LAB 10

Program : 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.



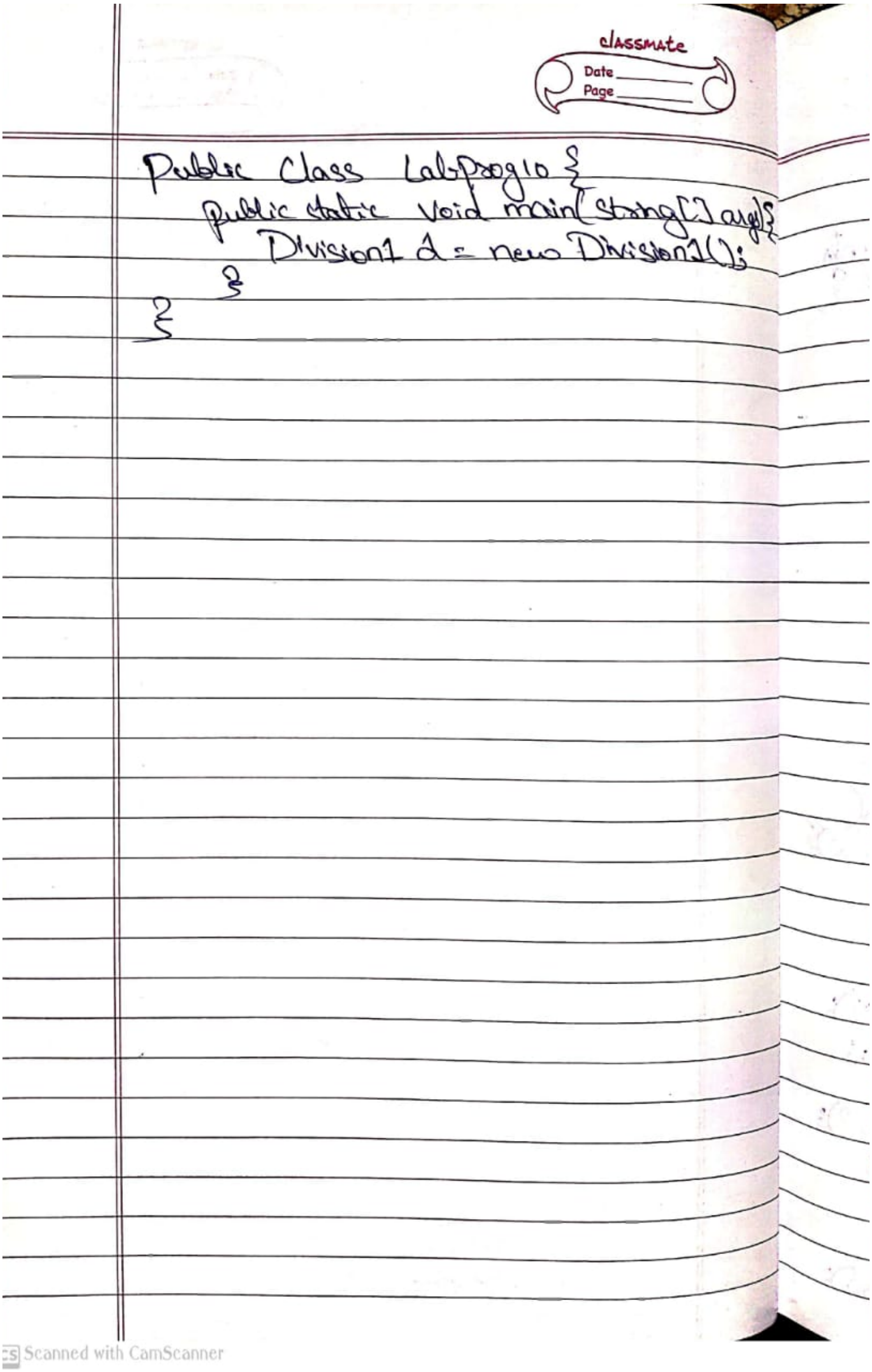
```
import java.awt.*;
import java.awt.event.*;
class Division1 Extends Frame implements
                              ActionListener{
  Frame F;
   Textfield tf1;
   Textfield tf2;
   Textfield tf3;
   Button b;
   Dialog d1;
   Division1(){
         SetSize (300,300);
         SetVisible(true);
         setLayout(null);
          addWindowListener(new
             WindowAdapter(){
               Public void WindowClosing(
                    WindowEvent aew){
                    dispose();
               }
          });
      tf1=new Textfield ("Number1");
      tf2. setBounds (10,30,200,30);
      add(tf1);
      tf2=new TextField ("Number 2");
      tf2.setBounds (10,70,200,30);
      add(tf2);
```

Caption



```
        add(b)
        tf3=new TextField("Output");
        add(tf3);
    }
    public void actionPerformed(ActionEvent e){
        try{
            String num1=tf1.getText();
            int nu1=Integer.parseInt(num1);
            String num2=tf2.getText();
            int nu2= Integer.parseInt(num2);
            int result = nu1/nu2;
            tf3.setText(Integer.toString(result));
        }
        catch(NumberFormatException e2){
            d1=new Dialog(f,"error",true);
            Label l= new Label(""+e2);
            d1.add(l);
            d1.setSize(300,50);
            d1.setVisible(true);
        }
        catch(ArithmeticException e1){
            d1=new Dialog(f,"error",true);
            Label l= new Label(""+e1);
            d1.add(l);
            d1.setSize(300,50);
            d1.setVisible(true);
        }
    }
}
```

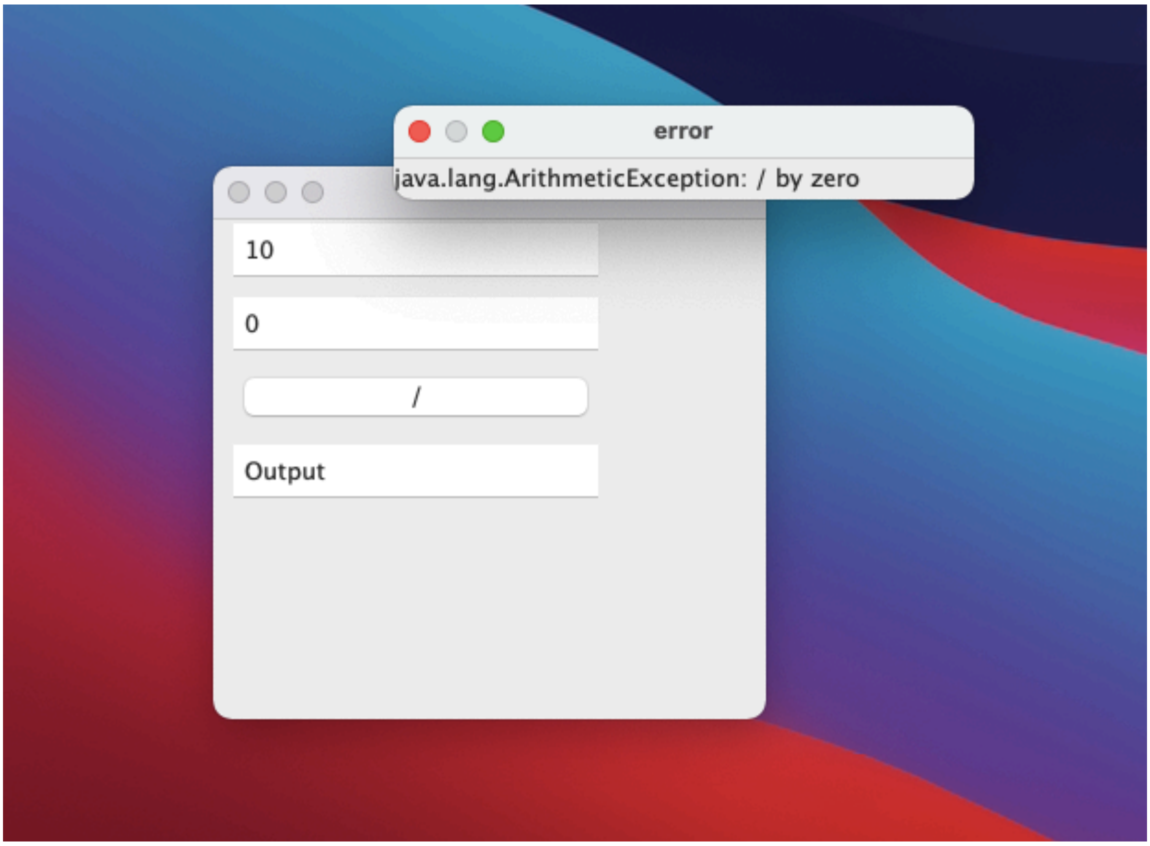
Caption



```
Public Class LabProg10 {
    public static void main(String[] args){
        Division1 d = new Division1();
    }
}
```

Caption

OUTPUT



```
java.lang.ArithmeticException: / by zero
```

Caption