LAB PROGRAM-6

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.

Week-09

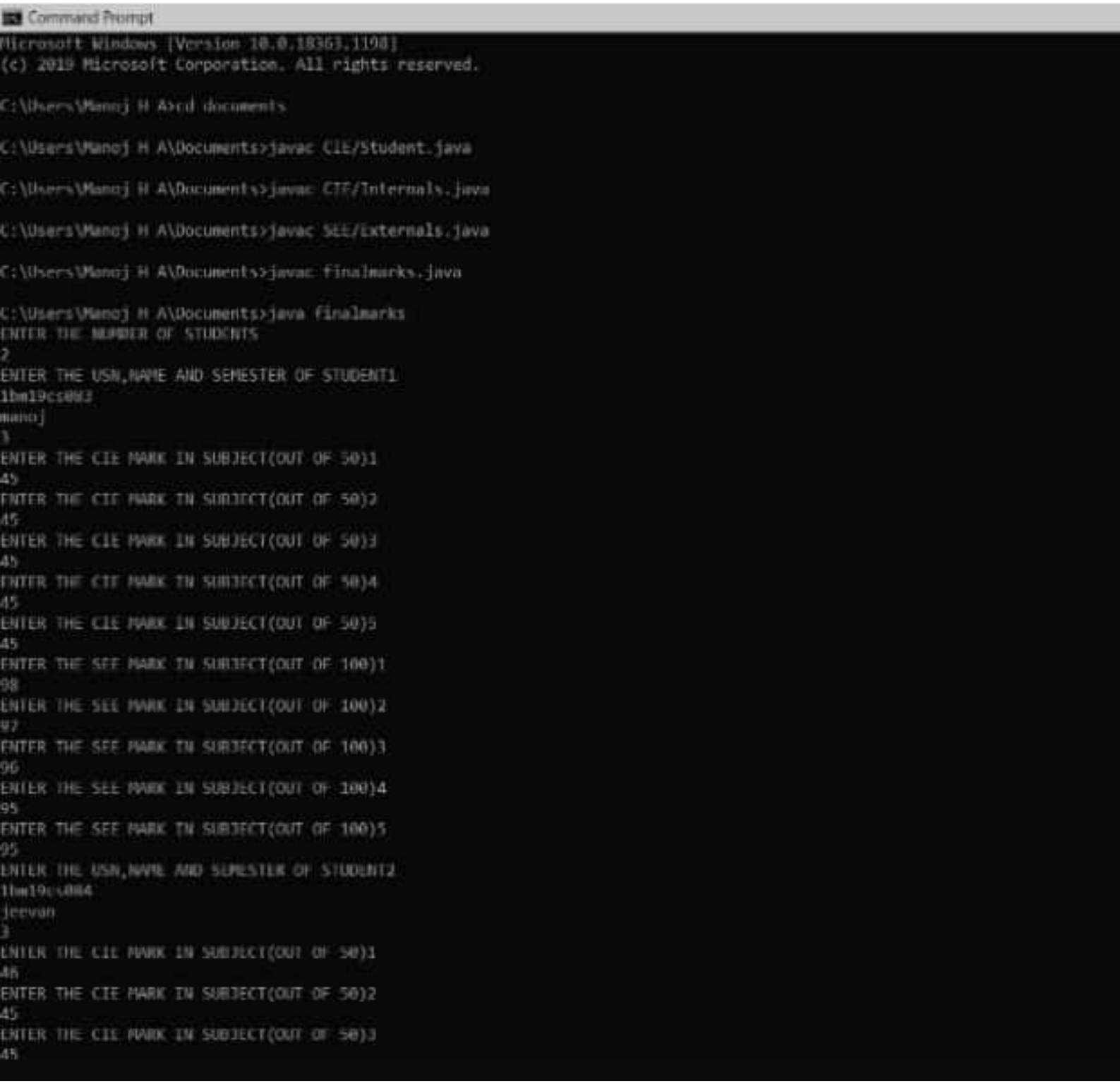
```
package CIE;
public class Student {
public String USN;
public String name;
public int sem;
public Student () {}
public Student (String usn, String name, int sem)
{ this.usn = usn;
  this.name = name;
  this.sem = sem;
 }
}
```

```
package CIE;
import java.util.Scanner;
public class Internals extends Student {
 Scanner s = new Scanner (System.in);
 public int [] cie = new int [5];
  public void get()
 {
   for (int i=0; i<5; i++)
   { System.out.println("Enter CIE marks in
                  subject (out of 50)" + (i+1));
     cie[i] = s.nextInt();
   }
 }
}
```

```
package SEE;
import java.util.Scanner;
public class Externals extends CIE.Student
{
  public Externals (String usn, String name, int sem)
{
  super (usn, name, sem);
}
Scanner s=new Scanner (System.in);
public int[] see = new int[5];
public void get()
{
  for(int i=0; i<5; i++)
{ System.out.println("Enter the SEE marks in
                    subject (out of 100) "+(i+1));
      see[i] = s.nextInt();
    }
  }
}

import CIE.*;
import SEE.*;
import java.util.Scanner;
class final marks {
  public static void main (String args[])
{ Scanner s=new Scanner (System.in);
System.out.println("Enter the no. of students");
int n=s.nextInt();
SEE.Externals obj[] = new SEE.Externals[n];
```

```
CIE.Internals obj[] = new CIE.Internals[n];
for(int i=0 ; i<n; i++)
{ System.out.println("Enter USN, name, sem
                    of student" + (i+1));
  String u = s.next();
  String na = s.next();
  int se = s.nextInt();
  obj[i] = new CIE.Internals();
  obj[i].get();
  obj1[i] = new SEE.Externals();
  obj1.get();
}
for(int i=0 ; i<n ; i++)
{ System.out.println("Name:" + obj1[i].name +
    " USN: " + obj1[i].usn + "Semester: " + obj1[i].sem);
  System.out.println("The total marks of student"
                    + (i+1) + "are");
    for(int j=0 ; j<5; j++)
  { System.out.println("subject " + (j+1) + " = "
                    + (obj[i].cie[j] + (obj1[i].see[j]/2)));
    }
}
}
```



```
ENTER THE USN,NAME AND SEMESTER OF STUDENT2
1bm19cs084
jeevan
3
ENTER THE CIE MARK IN SUBJECT(OUT OF 50)1
46
ENTER THE CIE MARK IN SUBJECT(OUT OF 50)2
45
ENTER THE CIE MARK IN SUBJECT(OUT OF 50)3
45
ENTER THE CIE MARK IN SUBJECT(OUT OF 50)4
48
ENTER THE CIE MARK IN SUBJECT(OUT OF 50)5
45
ENTER THE SEE MARK IN SUBJECT(OUT OF 100)1
98
ENTER THE SEE MARK IN SUBJECT(OUT OF 100)2
97
ENTER THE SEE MARK IN SUBJECT(OUT OF 100)3
96
ENTER THE SEE MARK IN SUBJECT(OUT OF 100)4
95
ENTER THE SEE MARK IN SUBJECT(OUT OF 100)5
95
NAME:manojUSN:1bm19cs083SEMESTER:3
THE TOTAL MARKS OF STUDENT1 ARE
SUBJECT1=94
SUBJECT2=93
SUBJECT3=93
SUBJECT4=92
SUBJECT5=92
NAME:jeevanUSN:1bm19cs084SEMESTER:3
THE TOTAL MARKS OF STUDENT2 ARE
SUBJECT1=95
SUBJECT2=93
SUBJECT3=93
SUBJECT4=95
SUBJECT5=92
```

LAB PROGRAM-7

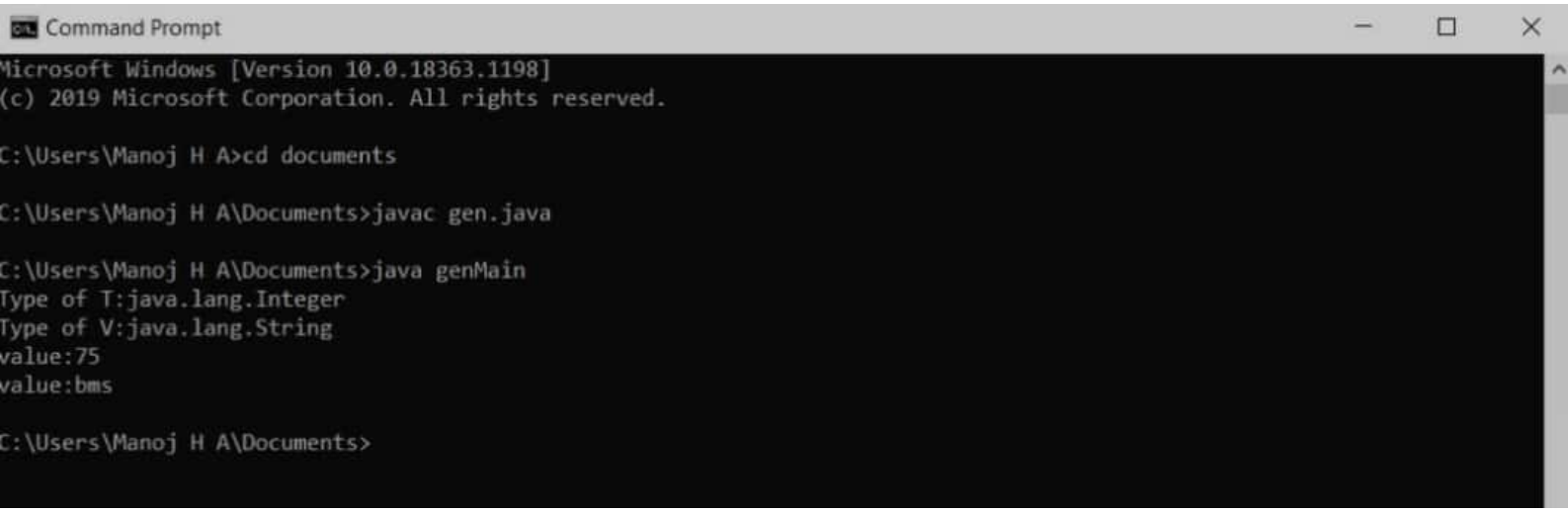
Write a program to demonstrate generics with multiple object parameters.

Week-10

```
class gen<T, V>{
 T ob1;
  V ob2;

 gen(T a, T b){
   ob1 = a;
   ob2 = b;
 }
 void showtypes(){
 System.out.println("Type of T : "+ob1.getClass().getName());
System.out.println("Type of V : "+ob2.getClass().getName());
}
T getob1(){
  return ob1;
 }
 V getob2(){
   return ob2;
 }
}
 class genMain{
 public static void main(String ss[]){
 gen<Integer, String> i=new gen
                          gen<Integer, String>(75, "bms");
 i.showtypes();
int v = i.getob1();
System.out.println("value "+v);
```

```
String str = i.getobj2();
System.out.println("value" + str);
}
}
```



LAB PROGRAM-8

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 Wrong Age( ) 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.

```
import java.util.Scanner;
class WrongAge extends Exception{
  int age;
 WrongAge(int x){
    age = x;
 }
 public String tostring(){
  return "Age of son" + age + "is incorrectly");
 }
}
class father{
   int a;
 father(int x){
   a=x;
 }
}
class son extends father{
   int age;
 son(int fage, int sage){
    super(fage);
    age=sage;
```

```
void compute() throws WrongAge {
    if (age >=a) {
        throw new WrongAge(age);
    } else {
  System.out.println("Ages are correct");
System.out.println("Father's age:" + a + "\t" +
                    "Son's age" + age);
  }
 }
}

class ExceptionsMain{
  public static void main(String args[])
  Scanner s=new Scanner(System.in);
System.out.println("Enter Father's age:");
 int f=s.nextInt();
System.out.println("Enter Son's age:");
 int so=s.nextInt();
 Son ss=new Son(f,so);
  try {
     ss.compute();
  }
  catch (WrongAge e) {
     System.out.println(e);
  }
 }
}
```

Command Prompt

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

C:\Users\Manoj H A>cd documents

C:\Users\Manoj H A\Documents>javac ExceptionsMain.java

C:\Users\Manoj H A\Documents>java ExceptionsMain
ENTER FATHER'S AGE:
45
ENTER SON'S AGE:
45
AGE OF SON=45 IS ENTERED INCORRECTLY

C:\Users\Manoj H A\Documents>java ExceptionsMain
ENTER FATHER'S AGE:
45
ENTER SON'S AGE:
20
THE AGES ARE ENTERED CORECTLY
FATHER'S AGE=45 SON'S AGE=20

C:\Users\Manoj H A\Documents>
```

LAB PROGRAM-9

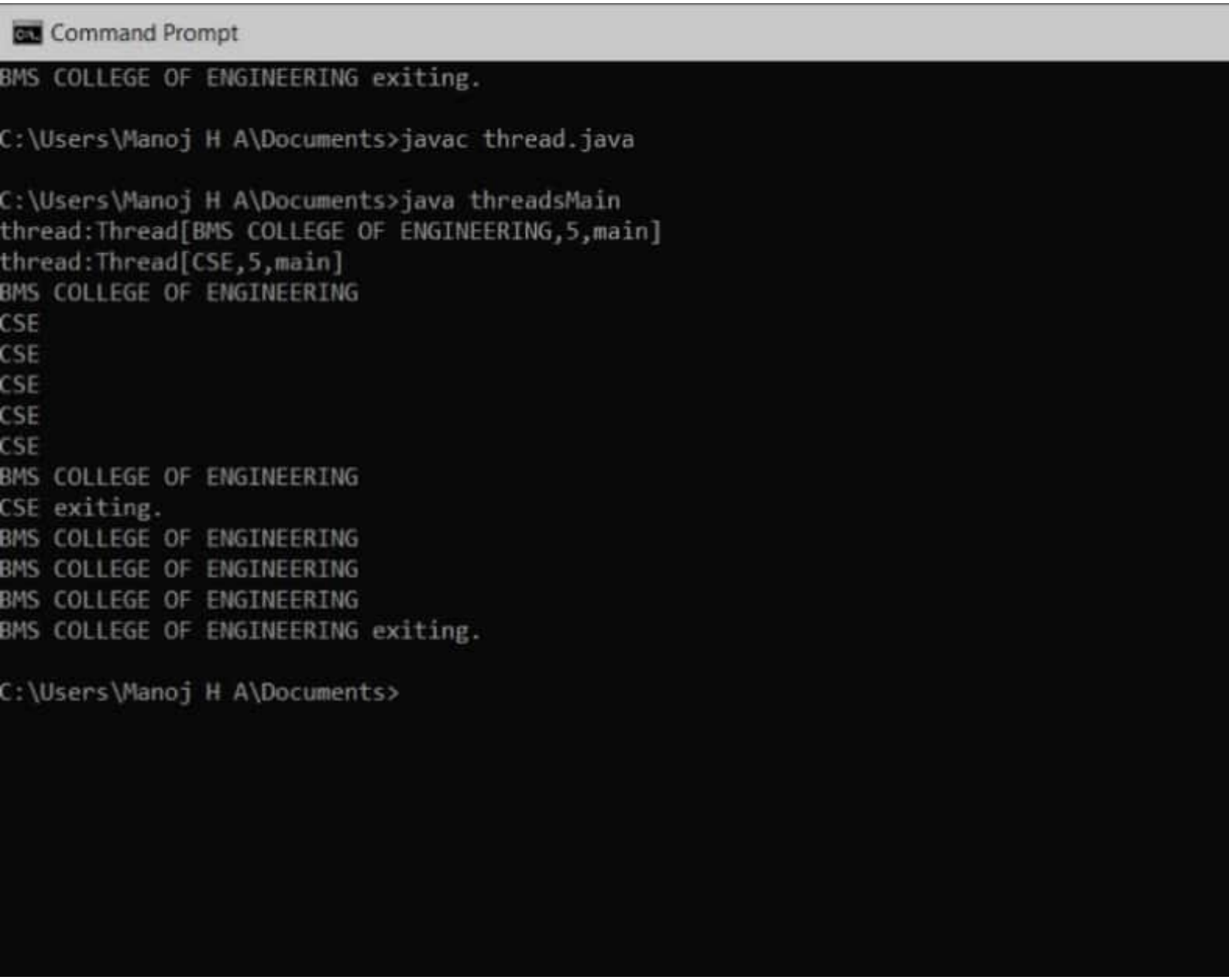
Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

Week-11

Manoj.H.A
1BM19CS083

```
class Thread1 implements Runnable
{ String name;
  Thread t;
   int time;
  Thread1 (String threadname, int time){
    name = threadname;
    this.time = time;
   t= new Thread (this, name);
System.out.println("thread :"+t);
    t.start();
}
  public void run(){
```

```
try {
for (int i=5; i>0; i--) {
  System.out.println("name);
  Thread.sleep (time);
}
catch (InterruptedExecption e) {
  System.out.println (name + " Interrupted");
}
System.out.println(name + "exiting");
}
class threadsMain {
   public static void main (String args[]) {
Thread 1 t1 = new Thread1 ("BMS college of Eng ", 10000);
Thread1. t2 = new Thread2 ("CSE", 2000);
}
}
```



LAB PROGRAM-10

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.

Week - 13

Manoj. H. A
1BM19CS083

```
import java.awt.*;
import. java.awt.event;
class dia extends Dialog implements {
IntegerDivision id;
dia(Frame parent, String title){
super(parent, title, false);
id = (IntegerDivision) parent;
setLayout(new FlowLayout());
setSize(300, 300);
add(new Label(id.msg));
Button b;
add(b = new Button("ok"));
b.addActionListener(this);
}
public void actionPerformed(ActionEvent ae){
  dispose();
}
}
public class IntegerDivision extends Frame implements
                                    ActionListener{
TextField n1, n2, res;
String msg = " ";
Label ln1, ln2, lres;
Button b;
public IntegerDivision(){
setLayout(new FlowLayout());
Label fn1=new Label("Number 1", Label.Right);
```

```
Label ln2 = new Label("Number 2", Label.Right);
Label lres = new Label("Result", Label.Right);
n1=new TextField(12);
n2=new TextField(8);
res = new TextField(10);
b=new Button("Divide");
add(ln1);
add(n1);
add(ln2);
add(n2);
add(lres);
add(res);

b.addActionListener(this);
addWindowListener(new WindowAdapter1());
}
public void actionPerformed(ActionEvent ae)
{
  if(ae.getSource()==b)
  {try{
    int num1=Integer.parseInt(n1.getText());
    int num2= Integer.ParseInt(n2.getText());
    int num3= num1/num2;
    res.setText(String.valueOf(num3));
  } catch(NumberFormatException ne){
    msg = "Number Format Exception";
    dia d=new dia(this, "Exception");
    d.setVisible(true);
  }
  catch(ArithmeticException a){
    msg = "Arithmetic Exception";
    dia d=new dia(this, "Exception");
```

```
        d.setVisible(true);
    }
}

public static void main(String args[])
{
    IntegerDivision i=new IntegerDivision();
    i.setSize(new Dimension(300,300));
    i.setTitle("Integer Division of 2 numbers");
    i.setVisible(true);
}
class WindowAdapter1 extends WindowAdapter{
    public void WindowClosing(WindowEvent we)
    {
        System.exit(0);
    }
}
}
```