

Name-Vineeth R Rao
USN-1BM19CS183

Lab Program - 1

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 discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

```
import java.util.Scanner;
import java.lang.*;

class Quadratic
{
    public static void main (String args[])
    {
        double a,b,c,d,x1,x2;
        Scanner scan = new Scanner (System.in);
        System.out.println ("enter co-efficients of equation");
        a = scan.nextDouble();
        b = scan.nextDouble();
        c = scan.nextDouble();
        d = b*b - 4*a*c;
        if (d < 0)
        {
            System.out.println ("There are no real solutions");
            System.exit(0);
        }
        x1 = (-b + Math.sqrt(d)) / (2*a);
        x2 = (-b - Math.sqrt(d)) / (2*a);
        if (d == 0)
        {
            System.out.println ("The solutions are real and equal to " + x1);
        }
        else
        {
            System.out.println ("The real solutions are " + x1 + " and " + x2);
        }
    }
}
```

Output-Lab Program 1

```
Command Prompt
NaN and NaN

D:\>javac Quadratic.java

D:\>java Quadratic
enter the co-efficients of the quadratic equation a,b and c

1
2
8
the roots are imaginary:-2.0+(-28.0i),-2.0+(28.0i)

D:\>java Quadratic
enter the co-efficients of the quadratic equation a,b and c

2
8
3
The real solutions are
-0.41886116991581024 and -3.58113883008419

D:\>java Quadratic
enter the co-efficients of the quadratic equation a,b and c

1
2
1
The solutions are real and equal to -1.0

D:\>
```

Lab Program-2

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

```
import java.util.Scanner;  
  
class Student  
{ int usn; int n; String name;  
int credit[];  
int[] credit = new int [100];  
int marks[]; float[] grades = new float[100];  
int[] marks = new marks [100];  
int product[];  
void input() {  
    Scanner scan = new Scanner (System.in);  
    System.out.println ("Enter student USN");  
    usn = scan.nextInt();  
    System.out.println ("Enter student name");  
    name = scan.nextLine();  
    System.out.println ("Enter number of subjects");  
    n = scan.nextInt();  
    System.out.println ("Enter credits for " + n + " subjects");  
    for (int i = 0; i < n; i++)  
    { credit[i] = scan.nextInt(); }  
    System.out.println ("Enter marks of " + n + " subjects");  
    for (int j = 0; j < n; j++)  
    { marks[j] = scan.nextInt(); }  
}  
  
double calculate () {  
    for (int i = 0; i < n; i++) {  
        grades[i] = marks[i] / credit[i]; }  
    int sum = 0;  
    for (int i = 0; i < n; i++)  
    { sum = sum + product[i]; }  
    return sum / n;  
}
```

```
{ int total_credit = 0;  
for ( i=0; i<n; i++ )  
{ total_credit = credit[i] + total_credit; }
```

```
return ( prob Sum / total_credit );  
}
```

```
void display()
```

```
{ System.out.println ("Student id is " + id);  
System.out.println ("student name is " + name);  
System.out.println ("SGPA of student = " + calculate());  
}
```

```
public static void main [String args [ ] ]
```

```
{ student s1;  
s1.input();  
s1.calculate();  
s1.output();  
}
```

Output-Lab Program 2

```
D:\>java student
enter the student usn
1bm19cs183
enter the student name
vineeth r rao
enter number of subjects
5
enter credits for all subjects
5
4
3
2
1
enter marks for all subjects
100
87
56
78
45
student details are
usn: 1bm19cs183
name: vineeth r rao
sgpa : 7.666666507720947

D:\>_
```

Lab Program-3

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

```
import java.util.*;  
SPLASH  
// Lab Program - 3 //  
class Book  
{  
    String name;  
    String author;  
    double price;  
    int num_pages;  
  
    Book (String a, String b, double c, int d)  
    {  
        name = a;  
        author = b;  
        price = c;  
        num_pages = d;  
    }  
  
    Book ()  
    {  
        name = "name";  
        author = "name";  
        price = 0.0;  
        num_pages = 0;  
    }  
  
    void setdata ()  
    {  
        System.out.println ("Enter book details");  
        System.out.println ();  
        Scanner scan = new Scanner (System.in);  
        System.out.println ("Enter name of the book");  
        String name = scan.nextLine();  
        System.out.println ("Enter author of the book");  
        String author = scan.nextLine();  
        System.out.println ("Enter price and no. of pages");  
        double price = scan.nextDouble();  
        int num_pages = scan.nextInt();  
    }  
}
```

```
public String toString()
{
    return ("name:" + name + "author:" + author +
           "price:" + price + "number of pages:" + numPage)
}
```

```
class mainclass {
    public static void main (String args[])
    {
        Scanner scan = new Scanner (System.in)
        int n;
        System.out.println ("Enter number of books");
        n = scan.nextInt();
        Book b[] = new Book [n];
        for (i=0; i < n; i++)
        {
            b[i] = new Book();
            System.out.println ("Enter book" + (i+1) + "details");
            b[i].setData();
        }
        System.out.println ("The book details are:");
        for (i=0; i < n; i++)
        {
            System.out.println ("book" + (i+1) + ";");
            System.out.println (b[i]);
        }
    }
}
```

Output-Lab program 3

```
D:\>java mainclass
enter the number of books
1
Enter book1details:
enter the book name
maths easy way
enter the author name
jj dosh
enter the book price
450
enter number of pages
467
The book details are:
book 1 :
name of the book:maths easy way
author of the book:jj dosh
price of the book:450.0
number of pages:467
```

Lab program 4

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

shape.java
import java.util.Scanner;
abstract class Shape
{ int a;
int b;
Shape (int a, int b)
{ this.a = a;
this.b = b; }
Shape (int a)
{ this.a = a; }

Shape()
{ this.a = 0;
this.b = 0; }

abstract void printArea();
}
class triangle extends Shape
{ triangle (int a, int b)
{ super(a, b); }

void printArea()
{ System.out.println ("Area of triangle is = " + (a*b)/2); }
}
class rectangle extends Shape
{ rectangle (int a, int b)
{ super(a, b); }

void printArea()
{ System.out.println ("Area of rectangle is " + (a+b)); }
}

class circle extends shape

VINEETH.R

{ circle (int a)

{ super(a); }

void printArea()

{ System.out.println("Area of circle is " + (3.14 * a * a)); }

}

class shapes {

public static void main(String args [])

{ Scanner sc = new Scanner (System.in)

int ch, a, b;

System.out.println("Enter 1 for triangle, 2 for rectangle, 3 for
circle, 4 for exit");

ch = sc.nextInt();

switch(ch)

{ case 1: System.out.println("enter base, height of Triangle");

a = sc.nextInt();

b = sc.nextInt();

triangle t = new triangle (a,b);

t.printArea();

break;

case 2: System.out.println("enter length and breadth");

a = sc.nextInt();

b = sc.nextInt();

rectangle r = new rectangle (a,b)

r.printArea();

break;

case 3: System.out.println("enter the radius of circle")

~~a~~ a = sc.nextInt()

circle c = new circle(a);

c.printArea();

break;

case 1: scan.close();
break;

default: System.out.println("invalid input");

}

}

}

Output -Lab Program4

```
D:\>java shapes
ENTER 1 FOR TRIANGLE
ENTER 2 FOR RECTANGLE
ENTER 3 FOR CIRCLE
ENTER 4 FOR EXIT
4

D:\>java shapes
ENTER 1 FOR TRIANGLE
ENTER 2 FOR RECTANGLE
ENTER 3 FOR CIRCLE
ENTER 4 FOR EXIT
2
enter the length and breadth of rectangle
2
5
the area of the rectangle is = 10

D:\>java shapes
ENTER 1 FOR TRIANGLE
ENTER 2 FOR RECTANGLE
ENTER 3 FOR CIRCLE
ENTER 4 FOR EXIT
3
enter the radius of the circle
100
the area of the circle is = 31400.0

D:\>
```

Lab program 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes 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.

LAB - 5

```
import java.util.*;  
class account {  
    String customer-name;  
    int account-number;  
    String account-type;  
};  
  
class curr-acct extends account {  
    Scanner s = new Scanner(System.in);  
    double temp = 0.0;  
    double amount = 0.0;  
    double fine = 0.0;  
    double min-amount = 1000.0;  
    void getdetails() {  
        customer-name = s.nextLine();  
        account-number = s.nextInt();  
    };  
    void deposit() {  
        System.out.print("Enter the deposit amount : ");  
        temp = s.nextDouble();  
        amount += temp;  
    };  
    void showbalance()  
    { if (amount >= min-amount) {  
        System.out.println("Balance is :" + amount);  
    }  
    else {  
        fine = (amount * 1.0 + 10) / 100;  
        amount -= fine;  
        System.out.println("The fine imposed : " + fine);  
        System.out.println("Balance is : " + amount);  
    }  
}
```

```
    }  
}  
void withdrawl(){
```

```
    System.out.print("Enter the withdrawal amount : ");  
    temp = n.nextDouble();  
    amount -= temp;
```

```
}
```

```
}
```

```
class Sav-Acc extends Account {
```

```
Scanner n = new Scanner(System.in);
```

```
double temp = 0.0;
```

```
double amount = 0.0;
```

```
double interest = 0.0;
```

```
void getdetails(){
```

```
customer_name = n.nextLine();
```

```
account_number = n.nextInt();
```

```
}
```

```
void deposit(){
```

```
System.out.print("Enter the deposit amount : ");
```

```
temp = n.nextDouble();
```

```
amount += temp;
```

```
}
```

```
void showbalance(){
```

```
System.out.println("Balance is : " + amount);
```

```
}
```

```
void withdrawl(){
```

```
System.out.print("Enter the withdrawal amount : ");
```

```
temp = n.nextDouble();
```

```
amount -= temp;
```

```
}
```

void interest () {

$$\text{interest} = (\text{amount} * 1.0 + 3) / 100;$$

amount += interest;

System.out.println("Interest added : " + interest);

System.out.println("Balance is : " + amount);

}

}

public class Main {

public static void main (String [] args) {

int opt = 0;

String type = null;

Scanner s = new Scanner (System.in);

System.out.println ("Welcome to bank service");

System.out.println ("Enter the account (curr-acct / sav-acct)");

type = s.nextLine();

if [type.equals ("curr-acct")] {

curr-acct a = new curr-acct();

System.out.println ("Enter the customer-name, account-number : ");

a.getDetails();

while (true) {

System.out.println ("Press 1 : Accept deposit and update the balance");

System.out.println ("Press 2 : Display the balance");

System.out.println ("Press 3 : Withdrawal and update balance");

System.out.println ("Enter option : ");

opt = s.nextInt();

switch (opt) {

case 1 : a.deposit();

a.showbalance();

break;

case 2 : a.showbalance();

break;

case 3 : a.withdraw();
a.showbalance();
break;

} }

}

if (type.equals ("sav-acct")) {

Sav-acct a = new Sav-acct();

System.out.println ("Enter customer-name, account-number");
a.getdetails();

switch (opt) {

System.out.println ("Press 1. Accept deposit and update the balance");

System.out.println ("Press 2. Display the balance");

System.out.println ("Press 3. Compute and deposit interest");

System.out.println ("Press 4. Withdrawal and update the balance");

System.out.println ("Enter option : ");
opt = s.nextInt();

switch (opt) {

case 1 : a.deposit();
a.showbalance();
break;

case 2 : a.showbalance();
break;

case 3 : a.interest();
a.showbalance();
break;

Case 4 : a.withdrawal();
a.showbalance();
break;

{

{

{

{

Output-Lab Program-5

```
Enter the customer_name,account_number:  
e  
33  
press 1 : Accept deposit and update the balance  
press 2 : Display the balance  
press 3 : Compute and deposit interest  
press 4 : Withdrawal and update the balance  
Enter option : 1  
Enter the deposit amount : 3344  
Balance is : 3344.0  
press 1 : Accept deposit and update the balance  
press 2 : Display the balance  
press 3 : Compute and deposit interest  
press 4 : Withdrawal and update the balance  
Enter option : 3  
interest added : 100.32  
Balance is : 3444.32  
Balance is : 3444.32  
press 1 : Accept deposit and update the balance  
press 2 : Display the balance  
press 3 : Compute and deposit interest  
press 4 : Withdrawal and update the balance  
Enter option : 4  
Enter the withdrawal amount : 1000  
Balance is : 2444.32  
press 1 : Accept deposit and update the balance  
press 2 : Display the balance  
press 3 : Compute and deposit interest  
press 4 : Withdrawal and update the balance
```

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.

```
import java.util.*;  
package CIE;  
  
import java.util.*;  
import package CIE.*; 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;  
int final_marks[];  
System.out.println ("Enter the number of students");  
n = scan.nextInt();  
}
```

```
sem = scan.nextInt();  
s[i] = new Student(name, user, sem);  
System.out.println("Enter internal marks for 5 subjects");  
in[i] = new Internal();  
in[i].setmarks();  
System.out.println("Enter external marks for 5 subjects");  
e[i] = new External();  
e[i].setmarks();  
}
```

50 50 = 100

100

```
int final_marks[ ][ ] = new int [n][5];
```

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

```
for(j=0;j<5;j++) .
```

```
{ final_marks[i][j] = in[i].marks[j] + (e[i].marks[j]/2); }
```

}

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

```
{ System.out.println("Student +(i+1) + details are ");
```

```
s[i].getdetails();
```

```
System.out.println("Final marks in 5 subjects are = ");
```

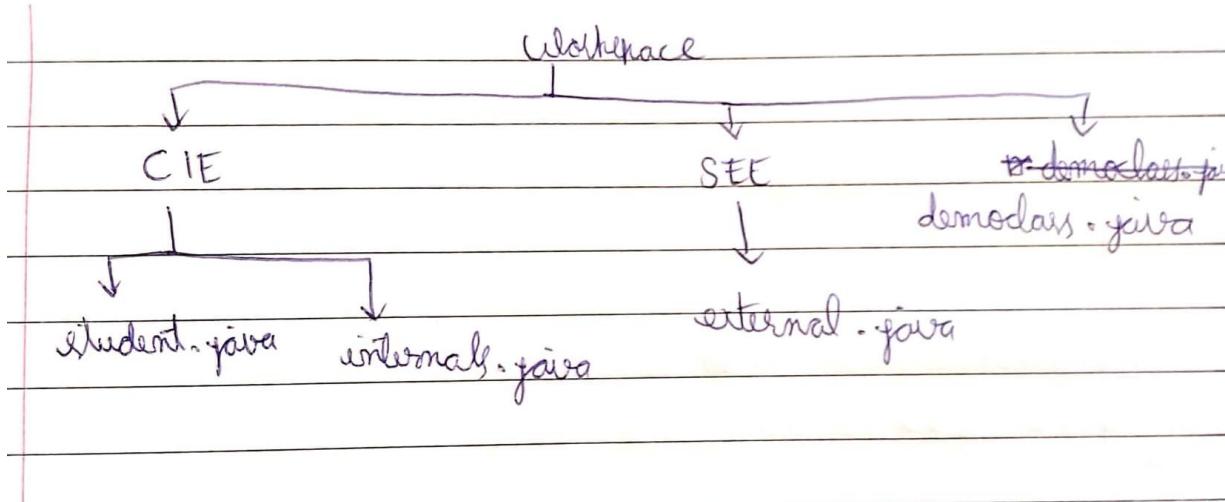
```
for(j=0;j<5;j++).
```

```
{ System.out.println(final_marks[i][j]); for
```

```
System.out.println(final_marks[i][j]); }
```

}

}



Output-Lab Program-6

```

c:\Java>cd cie
c:\Java\cie>javac Student.java
c:\Java\cie>javac Internals.java
c:\Java\cie>cd c:\java\see
c:\Java\see>javac -cp "C:\Java" External.java
c:\Java\see>cd c:\java
c:\Java>javac Democlass.java
c:\Java>java Democlass
enter number of students

```

```
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
```

Lab program 7

Write a program to demonstrate generics with multiple object parameters.

```
class TwoGen<T, V>
{
    T ob1;
    V ob2;
    TwoGen(T ob1, V ob2)
    {
        ob1 = ob1;
        ob2 = ob2;
    }
    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;
    }
}

class Mainclass
{
    public static void main(String args[])
    {
        TwoGen<Integer, String> obj = new
        TwoGen<Integer, String>(99, "Java");
        obj.showTypes();
        int v = obj.getOb1();
        System.out.println("value : " + v);
        String str = obj.getOb2();
        System.out.println("value of string : " + str);
    }
}
```

Output-Lab Program-7

```
c:\Java>javac MainClass.java

c:\Java>java MainClass
Enter a INTEGER
308
Enter a STRING
chris gayle
Type of T is java.lang.Integer
Type of V is java.lang.String
value: 308
value: chris gayle

c:\Java>java MainClass
Enter a INTEGER
99
Enter a STRING
sachin tendulkar
Type of T is java.lang.Integer
Type of V is java.lang.String
value: 99
value: sachin tendulkar
```

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 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.

```
import java.util.Scanner;  
  
class Main {  
    public static void main(String args[]) {  
  
        Scanner in = new Scanner(System.in);  
        System.out.println("Enter father's age:");  
        int fatherAge = in.nextInt();  
        System.out.print("Enter son's age:");  
        int sonAge = in.nextInt();  
        in.close();  
  
        Son son;  
        try {  
            son = new Son(sonAge, fatherAge);  
            System.out.println("No error");  
        } catch (AgeLessThanZeroException exception) {  
            System.out.println(exception.getMessage());  
        } catch (SonOlderThanFatherException exception) {  
            System.out.println(exception.getMessage());  
        }  
    }  
}
```

public class AgeLessThanZeroException extends Exception {

AgeLessThanZeroException (String message) {

super(message);

}

}

public class SonOlderThanFatherException extends Exception {

SonOlderThanFatherException (String message) {

super(message);

}

}

public class Father {

int age = 0;

message = "Father's age cannot be less than zero";

Father () { }

Father (int age) throws ~~Age less than~~ AgeLessThanZeroException

{

if (age < 0)

throw new AgeLessThanZeroException (message);

this.age = age;

}

}

public class Son extends Father {

String message1 = "Son's age cannot be greater than father's age";

String message2 = "Son's age cannot be less than zero";
int age;

Son (int sonAge, int FatherAge) throws AgeLessThanZeroException,
SonOlderThanFatherException.

{

super (fatherAge);

if (sonAge < 0)

throws new AgeLessThanException (message2);

if (sonAge >= fatherAge)

throws new SonOlderThanFatherException (message1);

age = sonAge;

}

}

Output-Lab Program-8

```
D:\>cd Java  
D:\Java>cd prog 8  
D:\Java\prog 8>javac Main.java  
  
D:\Java\prog 8>java Main  
Enter father's age : 38  
Enter son's age : 49  
Age of son can't be greater than age of father  
  
D:\Java\prog 8>javac Main.java  
  
D:\Java\prog 8>java Main  
Enter father's age : -19  
Enter son's age : 20  
Father's age cannot be less than zero  
  
D:\Java\prog 8>java Main  
Enter father's age : 40  
Enter son's age : -9  
Son's age cannot be less than zero  
  
D:\Java\prog 8>java Main  
Enter father's age : 40  
Enter son's age : 13  
No Errors  
  
D:\Java\prog 8>
```

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.

class Thread1 extends Thread { new Thread ("Thread1"); start(); } public void run() { while(1) { try { System.out.println ("BMS college of Engineering"); Thread.sleep (10000); } catch (InterruptedException ie) { System.out.println ("Thread 1 interrupted"); } } }	VINCENTH.K.PAO IBM19CS183
class Thread2 extends Thread { new Thread ("Thread2"); start(); } public void run() { while(1){ try { System.out.println ("CSE"); Thread.sleep (2000); } catch (InterruptedException ie) { System.out.println ("Thread 2 interrupted"); } } }	

```
class threadmain {
    public static void main (String args[]) {
        System.out.println ("Click control+C to stop");
        thread1 t1 = new thread1();
        thread2 t2 = new thread2();
    }
}
```

Output-Lab Program-9

```
c:\Java>java threadmain
c:\Java>java threadmain
Enter 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
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
```

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 ArithmeticException. 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 e) {  
                dispose();  
            }  
        });  
  
        tf1 = new TextField("Number1");  
        tf1.setBounds(10, 70, 200, 30);  
        add(tf1);  
  
        b = new Button("/");  
        b.setBounds(10, 110, 200, 30);  
    }  
  
    public void actionPerformed(ActionEvent e) {  
        if (e.getSource() == b) {  
            int num1 = Integer.parseInt(tf1.getText());  
            int num2 = Integer.parseInt(tf2.getText());  
            tf3.setText(String.valueOf(num1 / num2));  
        }  
    }  
}
```

```
b.addActionListener(this);  
add(b);  
  
tf3 = new JTextField("Output");  
tf3.setBounds(10, 150, 200, 30);  
add(tf3);  
}
```

```
public void actionPerformed(ActionEvent e) {  
    try {  
        String num1 = tf1.getText();  
        int n1 = Integer.parseInt(num1)  
        String num2 = tf2.getText();  
        int n2 = Integer.parseInt(num2);  
        int result = n1 / n2;  
        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 (ArithmaticException e1) {  
    d1 = new Dialog(f, "error", true);  
    Label l = new Label(" " + e1);  
    d1.add(l);  
    d1.setSize(300, 50);  
    d1.setVisible(true);  
}  
}
```

```
public class Labprog10 {  
    public static void main(String[] args) {  
        Division d = new Division1();  
        ?  
        ?  
    }  
}
```

Output-Lab Program-10

