

Week 3

import java.util.Scanner;  
public class Roots {  
 public static void main (String args [ ]) {  
 double secondroot = 0, firstroot = 0;  
 Scanner sc = new Scanner (System.in);  
 System.out.println ("Enter the values of a,b,c");  
 double a = sc.nextDouble();  
 double b = sc.nextDouble();  
 double c = sc.nextDouble();  
 double determinant = ~~Math.sqrt~~ 
$$[b * b] - [4 * a * c];$$
  
  
 if (determinant > 0) {  
 firstroot = 
$$(-b + \sqrt{det}) / (2 * a);$$
  
 secondroot = 
$$(-b - \sqrt{det}) / (2 * a);$$
  
 System.out.println ("Roots are : " + firstroot + " and " + secondroot);  
 } else if (det == 0)  
 System.out.println ("Roots are : " +  $(-b + \sqrt{det}) / (2 * a)$ );  
 else if (det < 0)  
 System.out.println ("Roots don't exist");  
 }  
}

### Algorithm:

1. Input three nos  $a, b, c$
2.  $\det = \sqrt{b^2 - 4ac}$
3. if  $\det > 0$  then roots =  $(-b \pm \det)/2a$
4. else if  $\det = 0$  then roots =  $-b/2a$
5. else if  $\det < 0$  then no real roots exist.
6. end.

```
Administrator: Command Prompt  
Microsoft Windows [Version 10.0.18363.1082]  
(c) 2019 Microsoft Corporation. All rights reserved.  
C:\Users\Aishwarya V>cd C:\Program Files\Java\jdk1.8.0_261\bin  
C:\Program Files\Java\jdk1.8.0_261\bin>cd ojj  
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>set path="C:\Program Files\Java\jdk1.8.0_261\bin"  
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>javac quadratic.java  
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java quadratic  
enter the values  
2 4 -4  
roots are real and distinct: 0.73 and -2.73  
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>javac student.java  
  
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java std  
enter name, usn, no. of subjects  
aish 123 2  
enter credits and marks for subject1  
5 50  
enter credits and marks for subject2  
6 70  
name= aish  
usn= 123  
sgpa is 7.09  
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>
```

6/10

## Lab Program:

Q) Develop a Java program, class Student  
members → USN, name, an array credits and  
array marks. Methods to accept, display details  
and method to calculate SGPA of student.

A) import java.util.Scanner();

class Student

{

String name;

String usn;

int sum = 0, n; double sgpa;

int[] credits, score;

double[] marks;

void getdata()

{

Scanner ss = new Scanner(System.in);

System.out.println("enter name, usn, no. of subjects");

name = ss.next();

usn = ss.next();

n = ss.nextInt();

credits = new int[n];

marks = new double[n];

Score = new int[n];

int i;

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

{

Date \_\_\_\_\_  
Page \_\_\_\_\_

and marks

```

System.out.println("enter credits for subjects " + (i+1));
credits[i] = ss.nextInt();
System.out.println("enter
marks[i] = ss.nextDouble();
if (marks[i] >= 90)
    score[i] = 10;
else if (marks[i] >= 80 && marks[i] < 90)
    score[i] = 9;
else if (marks[i] >= 70 && marks[i] < 80)
    score[i] = 8;
else if (marks[i] >= 60 && marks[i] < 70)
    score[i] = 7;
else if (marks[i] >= 50 && marks[i] < 60)
    score[i] = 6;
else if (marks[i] >= 40 && marks[i] < 50)
    score[i] = 5;
else
    score[i] = 4;
sum = sum + credits[i];
}
void calc (int[] credits, int score[], int sum)
{
    int i;
    double sum, total = 0.0;
    for (i=0; i<n; i++)
        total = total + credits[i] * score[i];
}

```

Date \_\_\_\_\_  
Page \_\_\_\_\_

```

sgpa = total / sum;
}

void printdata()
{
    System.out.println("name = " + name);
    System.out.println("usn = " + usn);
    System.out.println("sgpa = " + sgpa);
}

class std
{
    public static void main (String args[])
    {
        Student s1 = new Student();
        s1.getdata();
        s1.printdata();
    }
}

```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java books
enter the no. of books
2
book no. 1
enter the name, auhor, price, number of pages of the book
as as 12 12
book no. 2
enter the name, auhor, price, number of pages of the book
sd sd 23 23
book details:
book no. 1
book: as author: as price: 12.0 pages: 12
book no. 2
book: sd author: sd price: 23.0 pages: 23
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>
```

1BM19CS008  
Aishwarya V

Week 5

BookString.java

13/10 1) Create Class Book, members → name, author, pages, price. Constructor to set values, methods to get values. Include `toString()` to return all details of n no. of objects.

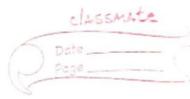
A) import java.util.Scanner;

Class book

```
private String name, author;
private double price;
private int pages;

book()
{
    name = "abcd";
    author = "xyz";
    price = 0.0;
    pages = 0;
}

void get()
{
    Scanner sc = new Scanner(System.in);
    System.out.println("enter name, author, price, pages");
    name = sc.next();
    author = sc.next();
    price = sc.nextDouble();
    pages = sc.nextInt();
}
```



public String toString()
{
 return ("book ; " + name + " author : " + author + " price : " + price + " pages : " + pages);
}

class books
{
 public static void main(String args[])
 {
 Scanner xx = new Scanner(System.in)
 System.out.println("Enter no. of Books");
 int n = xx.nextInt();
 book b[] = new book[n];
 for(int i=0; i < n; i++)
 {
 System.out.println("Book no. " + (i+1));
 b[i] = new book();
 b[i].get();
 }
 System.out.println("Book details : ");
 for(int i=0; i < n; i++)
 {
 System.out.println("book no." + (i+1));
 System.out.print(b[i]);
 }
 }
}

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java bank
```

```
Enter the type of account 1.savings 2.current
```

```
1
```

```
Enter the name , account number, type of bank account, balance
```

```
aishwarya 1234567 savings 30000
```

```
choose option for your savings account :1.deposit money (with compound interest) 2.withdraw money 3.check balance 4.checkbook 5.exit
```

```
1
```

```
enter the deposit amount, rate of interest ,number of years
```

```
20000 30 4
```

```
choose option for your savings account :1.deposit money (with compound interest) 2.withdraw money 3.check balance 4.checkbook 5.exit
```

```
3
```

```
the remaining balance is= Rs. 87121.99
```

```
choose option for your savings account :1.deposit money (with compound interest) 2.withdraw money 3.check balance 4.checkbook 5.exit
```

```
2
```

```
enter the amount you want to withdraw
```

```
2000
```

```
choose option for your savings account :1.deposit money (with compound interest) 2.withdraw money 3.check balance 4.checkbook 5.exit
```

```
3
```

```
the remaining balance is= Rs. 85121.99
```

```
choose option for your savings account :1.deposit money (with compound interest) 2.withdraw money 3.check balance 4.checkbook 5.exit
```

```
4
```

```
check book not provided in savings account
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>javac shapes.java

C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java shapes
Enter the height and base of the triangle
20 30
The area of the triangle= 300
Enter the sides of the rectangle
23 23
The area of the rectangle= 529
Enter the radius of the circle
34
The area of the circle= 3629.84
```

Aishwarya V  
1BM19 CS008

Week 8

Q) Class Bank → two kinds of accs -

Savings - provides C.I., withdrawal facilities, no checkbook.  
Current - provides checkbook, no interest.  
Should maintain minimum balance,  
if not service charge imposed.

Class Account - customer name, accno., type of acc.  
derive → class Curr-acct

→ class Sav-acct

methods - accept deposit from user and update balance.

- display balance.

- Compute and deposit interest

- Permit withdrawal and update balance.

- Check min balance, impose penalty, update balance.

Q) class Account

$$CI = P \left( 1 + \frac{R}{100} \right)^t \text{ annually}$$

String name, accno, type;

float bal;  
import java.lang.Math;

int option;  
Math.pow(a, b);  
a

float deposit, rate, years, withdrawal=0;

; Account()

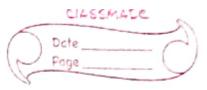
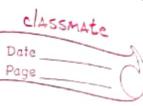
Scanner sc = new Scanner(System.in);

System.out.println("Enter the name, acc no, type, balance");

name = sc.next();

accno = sc.next();

type = sc.next();



bal = sc.nextDouble();

}

void display()

{ System.out.println("The name = " + name + " acc.no = " + accno +  
" type of bank account = " + type + " balance is " + bal);

}

class sav-acct extends account

{

float ci = 0;

boolean checkbookFacility = false;

void savActivity()

{

Scanner sc = new Scanner(System.in);

do {

System.out.println("choose 1. deposit money  
2. withdraw 3. check balance 4. checkbook 5. exit");

option = sc.nextInt();

switch (option)

case 1: System.out.println("Enter the deposit amount, rate of interest  
no. of years");

deposit = sc.nextDouble();

rate = sc.nextDouble();

years = sc.nextDouble();

ci = deposit \* (float) Math.pow((1 + (rate / 100)), years));

bal = bal + ci;

break;

CLASSMATE  
Date \_\_\_\_\_  
Page \_\_\_\_\_

```

Case 2 : { System.out.println ("the enter amount to withdraw");
    withdrawal = sc.nextFloat();
    bal = bal - withdrawal;
    } break;

Case3 : { System.out.println ("the remaining balance is =Rs "+bal);
    break;

Case 4: { if (checkbookfacility)
    System.out.println ("Checkbook provided in savings account");
    else System.out.println (" " " not provided ");
    } break;

Case 5: break;

Default: System.out.println ("invalid option");
} while (option!=5);
}

class curr_acct extends account
{
    boolean checkbookfacility = true;
    void curr_activity()
    {
        Scanner sc=new Scanner (System.in);
        System.out.println ("Enter the minimum balance and penalty");
        float min= sc.nextFloat();
        float penalty = sc.nextFloat();
        do{ System.out.println ("choose 1. deposit 2. withdraw
            3. checkbalance 4. checkbook 5. exit ");
            option = sc.nextInt();
            switch (option):
                Case 1 : { System.out.println ("enter the deposit");
                    deposit = sc.nextFloat();
                    bal = bal + deposit;
                    } break;
                Case 2 : { System.out.println ("enter the amt you want to withdraw");
                    withdrawal = sc.nextFloat();
                    bal = bal - withdrawal;
                    } break;
                Case 3 : { if (bal < min)
                    bal -=penalty ;
                    System.out.println ("penalty = "+penalty );
                    System.out.println ("the remaining balance is =Rs ." +bal);
                    } break;
                Case 4: { if (checkbookfacility)
                    System.out.println ("check book provided ");
                    else System.out.println ("checkbook not provided ");
                    } break;
                Case 5: break;
                default: System.out.println ("invalid option");
            } while (option!=5);
    }
}

```

class bank {  
 public static void main(String args[]){  
 Scanner sc = new Scanner(System.in);  
 System.out.println("Enter the type 1. savings 2. current");  
 int type = sc.nextInt();  
 if (type == 1){  
 Sav acct s = new Sav\_acct();  
 s.sav\_activity();  
 }  
 else if (type == 2){  
 Curr\_acct c = new Curr\_acct();  
 c.curr\_activity();  
 }  
 }

4.3.2) Abstract Class Shape → 2 integers and empty method printArea();  
 3 classes extend Shape → rectangle, triangle and Circle.  
 Each contain only method printArea(), that prints the area  
 of the shape.

a). import java.util.Scanner;  
 abstract class Shape  
 { int a,b;  
 abstract void printArea();

class rectangle extends shape  
 { void printArea(){  
 Scanner sc = new Scanner(System.in);  
 System.out.println("Enter the sides of rectangle");  
 a = sc.nextInt();  
 b = sc.nextInt();  
 System.out.println("The area of rectangle = " + (a \* b));

class triangle extends shape  
 { void printArea(){  
 Scanner sc = new Scanner(System.in);  
 System.out.println("Enter radius of circle");  
 a = sc.nextInt();  
 b = sc.nextInt();  
 System.out.println("The area = " + ((3.14 \* a \* a) / 2));

class circle extends shape  
 { void printArea(){  
 Scanner sc = new Scanner(System.in);  
 System.out.println("Enter Radius");  
 a = sc.nextInt();  
 System.out.println("area = " + (3.14 \* a \* a));

## { Class Shapes

{ public static void main (String args[ ])

{ rectangle r = new rectangle();

r. printarea();

triangle t = new triangle();

t. printarea();

circle c = new circle();

c. printarea();

}

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj\pack\week9lab>java total
enter the number of students
1
details of student 1
enter the usn name and sem
123 aish 3
the internals marks in all 5 courses
20 30 40 50 40
the externals marks in all 5 courses
70 80 90 70 60
total marks of student 1
55
70
85
85
70
C:\Program Files\Java\jdk1.8.0_261\bin\ojj\pack\week9lab>
```

ABM19CS008

Aishwarya.V  
Week 9

classmate

Date \_\_\_\_\_

Page \_\_\_\_\_

1/11/20  
1) Package CIE → Class Student → usn, name, sem  
→ class internally <sup>extends Student</sup> marks in 5 subjects.

Package SEE → class external extends student → marks in "

Import 2 packages in a file compute final marks.

A) Student.java

```
package cie;  
import java.util.Scanner;  
public class Student  
{
```

Folder [week9 Lab]

```
    String usn, name, sem;  
    public void info()  
    {  
        Scanner sc = new Scanner (System.in);  
        System.out.println ("Enter usn name sem");  
        usn = sc.next();  
        name = sc.next();  
        sem = sc.next();  
    }
```

internals.java

```
package cie;  
import java.util.Scanner;  
public class internals extends Student  
{  
    int i;  
    public int a[] = new int[5];  
    public void internalss()  
    {
```

Scanner sc = new Scanner(System.in);  
System.out.println("the internal marks in all 5 courses");  
for(i=0; i<5; i++)  
{  
 a[i] = sc.nextInt();  
}

### external.java

```
package see;  
import java.util.Scanner;  
import cie.students;  
public class external extends Students  
{ int i;  
    public int b[] = new int[5];  
    public void external()  
    { Scanner sc = new Scanner(System.in);  
        System.out.println("the external marks in all 5 courses");  
        for(i=0; i<5; i++)  
        { b[i] = sc.nextInt();  
        }  
    }
```

### total.java

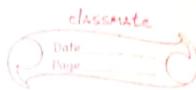
total.java

```
import cie.*;  
import see.*;  
import java.util.Scanner;  
public class total  
{ public static void main(String args[])  
{ int i, j, n;  
    int d[] = new int[5];  
    Scanner sc = new Scanner(System.in);  
    System.out.println("Enter the number of students");  
    n = sc.nextInt();  
    cie.Internal c[] = new cie.Internal();  
    see.External s[] = new see.External();  
    for(i=0; i<n; i++)  
    { c[i] = new cie.Internal();  
        s[i] = new see.External();  
        System.out.println("details of student " + (i+1));  
        c[i].info();  
        c[i].internal();  
        s[i].external();  
        System.out.println("total marks of student " + (i+1));  
        for(j=0; j<5; j++)  
        { d[j] = c[i].a[j] + s[i].b[j] / 2;  
        }  
        System.out.println(d[j]);  
    }  
}
```

1B419CS008  
Aishwarya.V  
Week 10

24/11/20

- 7) Demonstrate generics with multiple object parameters.
- A) import java.util.Scanner;  
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; }  
  
class multiplegen  
{ public static void main(String args[]){  
twoGen<Integer, String> tg = new twoGen<Integer, String>(  
(10, "hello world");  
tg.showTypes();  
int v = tg.getob1();  
System.out.println("value of int : " + v);  
String str = tg.getob2();  
System.out.println("value of string : " + str);  
}



twoGen<Boolean, float> tg1 = new twoGen<Boolean, float>  
(true, 3.14f);  
tg1.showTypes();

boolean b = tg1.getob1();  
System.out.println("value of boolean : " + b);  
float f = tg1.getob2();  
System.out.println("value of float : " + f);

- B) Exceptions in inheritance tree. Father → Son derived  
in Father, constructor takes age and throws exception  
WrongAge() when input age < 0.  
In Son, constructor throws exception if sonage ≥ fatherage.

A) Class WrongAge extends Exception  
private int a;  
WrongAge(int b)  
{ a = b;  
}  
public String toString() { return "WrongAge(" + a + ");"  
}  
  
class father  
{ int fatherage;  
father() throws WrongAge  
Scanner sc = new Scanner(System.in);  
System.out.print("Enter father's age");  
fatherage = sc.nextInt();  
}

```
if (fatherage < 0)
    throw new WrongAge(fatherage);
else System.out.println("correct father's age");
```

```
class Son extends Father
```

```
{ int sonage;
```

```
Son() throws WrongAge
```

```
{ Scanner sc = new Scanner(System.in);
```

```
System.out.println("enter son's age");
```

```
sonage = sc.nextInt();
```

```
if (sonage >= fatherage)
```

```
throw new WrongAge(sonage);
```

```
; else System.out.println("No error");
```

```
class exception_tree
```

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

```
{ try{ Son s = new Son();
```

```
} catch(WrongAge e)
```

```
System.out.println("Caught "+e);
```

```
}
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>javac exception_tree.java
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java exception_tree
enter father's age
-1
Caught WrongAge(-1)
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java exception_tree
enter father's age
50
correct father's age
enter son's age
60
Caught WrongAge(60)
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java exception_tree
enter father's age
50
correct father's age
enter son's age
50
Caught WrongAge(50)
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>javac multiplegen.java
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>java multiplegen
Type of T is java.lang.Integer
Type of V is java.lang.String
Type of X is java.lang.Double
value of int: 10
value of string: hello world
value of double: 20.35
Type of T is java.lang.Boolean
Type of V is java.lang.Float
value of boolean: true
value of float: 3.14
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj>
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj\threads>java bms_cse
New thread: Thread[BMS College Of Engineering,5,main]
New thread: Thread[CSE,5,main]
BMS College Of Engineering:5
CSE:5
CSE:4
CSE:3
CSE:2
CSE:1
BMS College Of Engineering:4
CSE exiting.
BMS College Of Engineering:3
BMS College Of Engineering:2
BMS College Of Engineering:1
BMS College Of Engineering exiting.
```

```
C:\Program Files\Java\jdk1.8.0_261\bin\ojj\threads>
```

Q WAP which creates 2 threads , one displays "BMS College of engineering" once every ten seconds, other "CSE" every two seconds.

A) class NewThread implements Runnable

```
{ String name;
  Thread t;
  int x;
```

```
NewThread (String threadname, int delay)
{ name = threadname;
  x = delay;
```

```
t = new Thread (this, name);
```

```
System.out.println ("New Thread :" + t);
```

```
t.start();
```

```
public void run()
```

```
try { for (int i=5; i>0; i--)
```

```
  { System.out.println (name + ":" + i);
    Thread.sleep(x);
```

```
}
```

```
Catch (InterruptedException e)
```

```
  { System.out.println (name + "Interrupted");
```

```
  System.out.println (name + "Exiting");
```

```
}
```

classmate

Date \_\_\_\_\_  
Page \_\_\_\_\_

```
class bms_CSE {  
    public static void main (String args [] )  
    {  
        new NewThread ("BMS College of Engineering", 10000);  
        new NewThread ("CSE", 2000);  
    }  
}
```

th="C:\Program Files\Java\jdk1.8.0\_261\bin"

divide\_awt

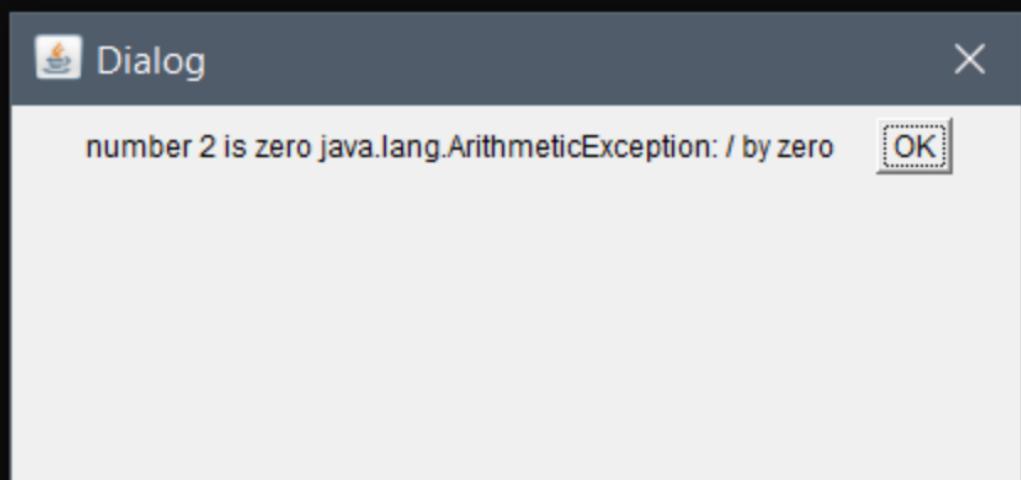
divide

divide\_a

Num1:

Num2:

Result:



divide\_awt

Num1:

Num2:

Result:

```
jj>set path="C:\Program Files\Java\jdk1.8.0_261\bin"
```

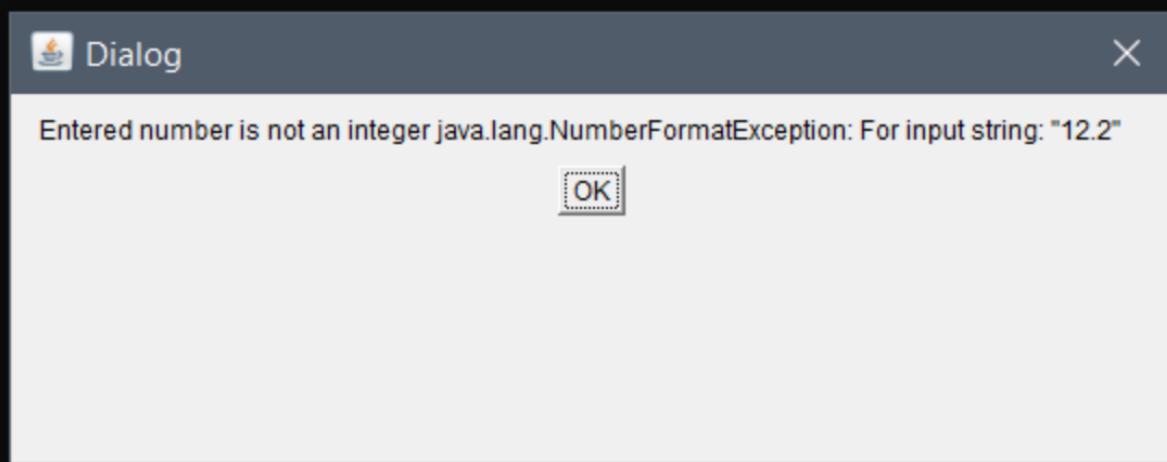
```
jj>javac divide_
```

```
jj>java divide_a
```

divide\_awt

Num1:  Num2:

Result:



15/12/20 Week 12 Lab program 10

Q) WAP that creates a user interface to perform integer divisions. The user enters two numbers in the text field, Num1 Num2, The division is displayed in Result field when divide is clicked. if not integer then NumberFormatException. If Num2 is zero then throw ArithmeticException display error in message dialog box

A) import java.awt.\*;

import java.awt.event.\*;

public class divide\_awt extends Frame implements

{

ActionListener

Textfield num1, num2, result;

int msg;

String msg1 = "";

Button divide;

public divide\_awt()

setLayout(new FlowLayout());

Label num1x = new Label("Num1:", Label.RIGHT);

Label num2x = new Label("Num2:", Label.RIGHT);

Label resultx = new Label("Result:", Label.RIGHT);

Button b = new Button("divide");

num1 = new TextField(8);

num2 = new TextField(8);

result = new Textfield(8);

```

    add(num1x);
    add(num1);
    add(num2x);
    add(num2);
    divide = (Button) add(b);
    add(result);
    add(result);
    num1.addActionListener(this);
    num2.addActionListener(this);
    divide.addActionListener(this);
    addWindowListener(new WindowAdapter()
    {
        public void windowClosing(WindowEvent we)
        {
            System.exit(0);
        }
    });
}

public void actionPerformed(ActionEvent ae)
{
    if (ae.getSource() == divide)
    {
        double a = Double.parseDouble(num1.getText());
        double b = Double.parseDouble(num2.getText());
        if (a%1 == 0 || b%1 != 0)
        {
            try
            {
                throw new NumberFormatException();
            } catch (NumberFormatException e)
            {
                msg1 = "Entered numbers is not an integer" + e;
                SampleDialog d = new SampleDialog(this, "Dialog");
                d.setVisible(true);
            }
        }
    }
}

```

CLASSMATE  
Date \_\_\_\_\_  
Page \_\_\_\_\_

```

else if (b == 0)
{
    try
    {
        throw new ArithmeticException();
    } catch (ArithmeticException e)
    {
        msg1 = "number 2 is zero" + e;
        SampleDialog d = new SampleDialog(this, "Dialog");
        d.setVisible(true);
    }
}
else
{
    msg = (int) a / (int) b;
    String c = "." + msg;
    result.setText(c);
    repaint();
}

public void paint(Graphics g)
{
    g.drawString("Exception: " + msg1, 20, 150);
}

```

```

public static void main(String[] args)
{
    divide.awt aa = new divide.awt();
    aa.setSize(new Dimension(380, 180));
    aa.setTitle("divide.awt");
    aa.setVisible(true);
}

```

classmate  
Date \_\_\_\_\_  
Page \_\_\_\_\_

```

at the beginning:
class SampleDialog extends Dialog implements ActionListener
{
    divide_unt bld;
    SampleDialog(Frame parent, String title)
    {
        Super(parent, title, false);
        bld = (divide_unt) parent;
        setLayout(new FlowLayout());
        setSize(300, 200);
        add(new JLabel(bld.msg1));
        Button b;
        add(b = new button("OK"));
        b.addActionListener(this);
    }
    public void actionPerformed(ActionEvent ae)
    {
        dispose();
    }
}

```

action performed in divide\_unt class

```

public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource() == divide)
    {
        try { msg1 = Integer.parseInt(num1.getText()) / Integer.parseInt(
            num2.getText());
        String c = "" + msg1;
        result.setText(c);
        msg1 = "";
    }
}

```

classmate  
Date \_\_\_\_\_  
Page \_\_\_\_\_

```

3 catch (NumberFormatException e)
{
    msg1 = "Entered number is not an integer " + e;
    SampleDialog d = new SampleDialog(this, "Dialog");
    d.setVisible(true);
}

catch (ArithmeticException e)
{
    msg1 = "number 2 is zero " + e;
    SampleDialog d = new SampleDialog(this, "Dialog");
    d.setVisible(true);
}

```