

NAME:NIHARIKA.B.S

USN:1BM19CS100

**RECORD TEST-2(ALL 10 PROGRAMS)**

**LAB PROGRAM-1**

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

**SOURCE CODE:**

```
import java.util.Scanner;
import static java.lang.Math.sqrt;
class Lab1{
    public static void main(String args[]){
        Scanner ss=new Scanner(System.in);
        double a,b,c,d,r1,r2;
        int temp;
        System.out.println("Enter the three co-efficients a,b,c of the quadratic equation");
        a=ss.nextDouble();
        b=ss.nextDouble();
        c=ss.nextDouble();
        d=((b*b)-(4*a*c));
        if(d==0)
        {
            temp=1;
        }
        else if(d>0)
        {
            temp=2;
        }
        else
```

```
{  
temp=3;  
}  
switch(temp)  
{  
case 1:System.out.println("The roots are real and equal");  
r1=r2=(-b/(2*a));  
System.out.println("The roots are "+r1+" and "+r2+"");  
break;  
case 2:System.out.println("The roots are real and distinct");  
r1=(-b+sqrt(d))/(2*a);  
r2=(-b-sqrt(d))/(2*a);  
System.out.println("The roots are "+r1+" and "+r2+"");  
break;  
case 3:System.out.println("The roots are imaginary,that is there are no real  
solutions to the given quadratic equation");  
break;  
default:System.out.println("Invalid input");  
break;  
}  
}  
}
```

## OBSERVATION:

### OOJ Lab-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.

→ Algorithm:-

1) START

2) Read  $a, b, c$  (coefficients of the quadratic equation)

3) Calculate  $d = b^2 - 4ac$

4) If ( $d == 0$ )

    → Calculate  $r_1(\text{root 1}) = \frac{-b}{2a}$

    → Calculate  $r_2(\text{root 2}) = \frac{-b}{2a}$

    → Display real and equal roots ( $r_1, r_2$ )

5) else if ( $d > 0$ )

    → Calculate  $r_1(\text{root 1}) = \frac{-b + \sqrt{d}}{2a}$

    → Calculate  $r_2(\text{root 2}) = \frac{-b - \sqrt{d}}{2a}$

    → Display real and distinct roots ( $r_1, r_2$ )

6) else if ( $d < 0$ )

    → Display no real solution

7) STOP

→ Program :-

```
import java.util.Scanner;  
import static java.lang.Math.sqrt;  
class Lab1 {  
    public static void main(String args[]) {  
        Scanner ss = new Scanner(System.in);  
        double a, b, c, d, r1, r2;  
        int temp;  
        System.out.println("Enter the three  
coefficients a, b, c of the quadratic equation");  
        a = ss.nextDouble();  
        b = ss.nextDouble();  
        c = ss.nextDouble();  
        d = ((b * b) - (4 * a * c));  
        if (d == 0)  
            temp = 1;  
        }  
        else if (d > 0)  
            temp = 2;  
    }
```

else {  
    ~~System.out.println("The roots are real and equal");~~

    temp = 3;

}

switch (temp)

{

case 1: System.out.println ("The roots are  
real and equal");

    r1=r2=(-b/(2\*a));

    System.out.println ("The roots are "+r1+"  
and "+r2+");

    break;

case 2: System.out.println ("The roots are real  
and distinct");

    r1=(-b+sqrt(d))/(2\*a);

    r2=(-b-sqrt(d))/(2\*a);

    System.out.println ("The roots are "+r1+"  
and "+r2+");

    break;

case 3: System.out.println ("The roots are  
imaginary, that is there are no real solutions to  
given quadratic equation");

    break;

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

break;      {  
  e = good }

}

{good.) return

}

  {  
    if (stack.size() > maxStack) throw new

    ("Large stack size")

    ; ((0^e) | d-) = SR = LR

  ("first" vs stack at "1") at max. two. metype

    ; (0^e + SR^+) and

; good.

  if (stack.size() < minStack) throw new

    ("Initial size")

    ; ((0^e) | ((b) type + d-) = LR

    ; ((0^e) | ((b) type - d-) = SR

  ("first" vs stack at "1") at max. two. metype

    ; ("at first" and

; good.

  if (stack.size() > maxStack) throw new

    ("maximum stack size exceeded")

    ; ("maximum stack size exceeded")

; good.

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>java Lab1
Enter the three co-efficients a,b,c of the quadratic equation
1
2
3
The roots are imaginary,that is there are no real solutions to the given quadratic equation
```

```
C:\Users\win10\Documents\Java lab programs>java Lab1
Enter the three co-efficients a,b,c of the quadratic equation
4
8
4
The roots are real and equal
The roots are -1.0 and -1.0
```

```
C:\Users\win10\Documents\Java lab programs>java Lab1
Enter the three co-efficients a,b,c of the quadratic equation
5
10
2
The roots are real and distinct
The roots are -0.2254033307585166 and -1.7745966692414832
```

## **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.**

### **SOURCE CODE:**

```
import java.util.*;
class Student {
    private String usn;
    private String name;
    private int credits[];
    private int marks[];
    private int n;
    void accept()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter student details");
        System.out.println("USN:");
        usn=s.next();
        System.out.println("Name:");
        name=s.next();
        System.out.println("Enter the number of subjects:");
        n=s.nextInt();
        credits=new int[n];
        marks=new int[n];
        System.out.println("Enter credits and marks attained by the student in each subject");
        for(int i=0;i<n;i++)
    }
```

```
{  
    credits[i]=s.nextInt();  
    marks[i]=s.nextInt();  
}  
}  
  
void display()  
{  
    System.out.println("Student details:");  
    System.out.println("USN:"+usn);  
    System.out.println("Name:"+name);  
    System.out.println("Marks in each subject:");  
    for(int i=0;i<n;i++)  
    {  
        System.out.println("Subject "+(i+1)+":"+marks[i]);  
    }  
}  
  
double calculate()  
{  
    int tcp=0,tc=0;  
    for(int i=0;i<n;i++)  
    {  
        tc=tc+credits[i];  
        if(marks[i]>=50)  
        {  
            tcp=tcp+(((marks[i]/10)+1)*credits[i]);  
        }  
    }  
}
```

```
else if(marks[i]>=40 && marks[i]<50)
{
    tcp=tcp+(4*credits[i]);
}
}
return (double)tcp/tc;
}
}

class StudentMain
{
public static void main(String ss[]) {
    Student s1=new Student();
    s1.accept();
    s1.display();
    System.out.println("SGPA: "+s1.calculate());
}
}
```

## OBSERVATION:

### OOJ Lab program - 2.

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

#### \* Algorithm :-

- 1) START
- 2) READ the members id, name, credits, marks in the method accept()
- 3) DISPLAY the student details in the method display()
- 4) CALCULATE the SGPA of the student in the method calculate() using array of credits and marks
- 5) In Student Main class, the objid is created and the default constructor of class Student is called.
- 6) The other methods of class Student is also called/invoke in the main class.
- 7) SGPA is DISPLAYED from the return value of calculate.
- 8) STOP.

→ CODE :-

```
import java.util.*;  
class Student {  
    private String usn;  
    private String name;  
    private int credits[];  
    private int marks[];  
    private int n;
```

void accept ()

{

```
Scanner s = new Scanner (System.in);  
System.out.println ("Enter student details");  
System.out.println ("USN : ");  
usn = s.next();  
System.out.println ("Name : ");  
name = s.next();  
System.out.println ("Enter the number of  
subjects : ");  
n = s.nextInt();
```

credits = new int [n];

marks = new int [n];



```
double calculate()
```

```
{
```

```
    int tcp = 0, tc = 0;
```

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

```
{
```

```
        tc = tc + credits[i];
```

```
        if (marks[i] >= 40)
```

```
{
```

```
            tcp = tcp + ((marks[i] / 10) + 1) *
```

```
                credits[i];
```

```
        else if (marks[i] >= 40 && marks[i] < 50)
```

```
            tcp = tcp + (4 * credits[i]);
```

```
    return (double) tcp / tc;
```

```
}
```

```
}
```

```
class StudentMain
```

```
{
```

```
    public static void main (String ss[])
```

```
{
```

```
        Student st = new Student();
```

```
    s1.accept();
    s1.display();
    System.out.println("SGPA : " + s1.calculate());
}
```

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>javac StudentMain.java
C:\Users\win10\Documents\Java lab programs>java StudentMain
Enter student details
USN:
1BM19CS100
Name:
Niharika
Enter the number of subjects:
5
Enter credits and marks attained by the student in each subject
4 77
5 78
3 74
4 75
4 77
Student details:
USN:1BM19CS100
Name:Niharika
Marks in each subject:
Subject 1:77
Subject 2:78
Subject 3:74
Subject 4:75
Subject 5:77
SGPA: 8.0
```

### **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.**

#### **SOURCE CODE:**

```
import java.util.*;
class Book {
String name;
String author;
int price;
int num_pages;
Book()
{}
Book(String name,String author,int price,int num_pages)
{
this.name=name;
this.author=author;
this.price=price;
this.num_pages=num_pages;
}
void accept()
{
Scanner s=new Scanner(System.in);
System.out.println("Enter the name of the book");
```

```
name=s.next();
System.out.println("Enter the author of the book");
author=s.next();
System.out.println("Enter the price of the book");
price=s.nextInt();
System.out.println("Enter the number of pages of the book");
num_pages=s.nextInt();
}
public String toString()
{
return ("Name: "+name + "\n" + "Author: "+author + "\n" + "Price: "+price + "\n"
+"Number of pages: "+num_pages );
}
}
}
class BookMain1 {
public static void main(String ss[])
{
Scanner a=new Scanner(System.in);
Book b1=new Book("Heights","Anne",299,345);
System.out.println("Sample input:\n"+b1);
System.out.println("Enter the number of books");
int n=a.nextInt();
Book b[]=new Book[n];
for(int i=0;i<n;i++)
{
b[i]=new Book();
System.out.println("\nEnter the details of book"+(i+1));
```

```
b[i].accept();  
}  
for(int i=0;i<n;i++)  
{  
System.out.println("\nDetails of book "+(i+1));  
System.out.println(b[i]);  
}  
}  
}
```

**OBSERVATION:**

## OOJ Tab Program - 03:

\*) 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 would display the complete details of the book. Develop a java program to create n book objects

### CODE:-

```
import java.util.*;  
class Book {  
    String name;  
    String author;  
    int price;  
    int num-pages;  
    Book()  
    {}  
    Book(String name, String author, int price,  
        int num-pages)
```

{

this.name = name;

this.author = author;

this.price = price;

this.num\_pages = num\_pages;

}

void accept()

{

Scanner s = new Scanner(System.in);

System.out.println("Enter the name of the book");

name = s.next();

System.out.println("Enter the author of the book");

author = s.next();

System.out.println("Enter the price of the book");

price = s.nextInt();

    System.out.println("Enter the number of pages of  
the book");

num\_pages = s.nextInt();

}

public String toString()

{

    return ("Name :" + name + "\n" + "Author :" + author + "\n" +  
"Price :" + price + "\n" + "Number of pages :" + num\_pages);

```
}

class BookMain {
    public static void main (String ss[])
    {
        Scanner a = new Scanner(System.in);
        Book b1 = new Book ("Heights", "Anne",
                           299, 345);
        System.out.println ("Sample input : \n" + b1);
        System.out.println ("Enter the number of books");
        int n = a.nextInt();
        Book b[] = new Book[n];
        for (int i=0; i<n; i++)
        {
            b[i] = new Book();
            System.out.println ("\nEnter the details of book"
                               +(i+1));
            b[i].accept();
        }
        for (int i=0; i<n; i++)
            System.out.println ("\n Details of book " +(i+1));
```

System.out.println(b[i]);

}

}

}

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>javac BookMain1.java
```

```
C:\Users\win10\Documents\Java lab programs>java BookMain1
```

```
Sample input:
```

```
Name: Heights
```

```
Author: Anne
```

```
Price: 299
```

```
Number of pages: 345
```

```
Enter the number of books
```

```
3
```

```
Enter the details of book1
```

```
Enter the name of the book
```

```
Sunshine
```

```
Enter the author of the book
```

```
John
```

```
Enter the price of the book
```

```
100
```

```
Enter the number of pages of the book
```

```
788
```

```
Enter the details of book2
```

```
Enter the name of the book
```

```
Crystal
```

```
Enter the author of the book
```

```
Henry
```

```
Enter the price of the book
```

```
200
```

```
Enter the number of pages of the book
```

```
500
```

```
Enter the details of book3
```

```
Enter the name of the book
```

```
King
```

```
Enter the author of the book
```

```
Rahul
```

```
Enter the price of the book
```

```
900
```

```
Enter the number of pages of the book
```

```
100
```

Details of book 1

Name: Sunshine

Author: John

Price: 100

Number of pages: 788

Details of book 2

Name: Crystal

Author: Henry

Price: 200

Number of pages: 500

Details of book 3

Name: King

Author: Rahul

Price: 900

Number of pages: 100

## **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.**

### **SOURCE CODE:**

```
import java.util.*;
abstract class Shape
{
int a,b;
abstract void printArea();
}
class Rectangle extends Shape
{
void printArea()
{
Scanner ss=new Scanner(System.in);
```

```
System.out.println("Enter length and breadth of the rectangle");
a=ss.nextInt();
b=ss.nextInt();
double area;
area=(double)a*b;
System.out.println("The area of Recatngle is "+area);
}
}
class Triangle extends Shape
{
void printArea()
{
Scanner ss=new Scanner(System.in);
System.out.println("Enter base length and height of the triangle");
a=ss.nextInt();
b=ss.nextInt();
double area;
area=(double)0.5*a*b;
System.out.println("The area of Triangle is "+area);
}
}
class Circle extends Shape
{
void printArea()
{
Scanner ss=new Scanner(System.in);
```

```
System.out.println("Enter radius of the circle");
a=ss.nextInt();
double area;
area=(double)3.14*a*a;
System.out.println("The area of Circle is "+area);
}
}
class Shapemain
{
public static void main(String args[])
{
int ch;
Scanner ss=new Scanner(System.in);
Rectangle r=new Rectangle();
Triangle t=new Triangle();
Circle c=new Circle();
while(true){
System.out.println("Enter the choice of shape whose area has to be calculated");
System.out.println("1.Rectangle\n2.Triangle\n3.Circle\n4.Exit");
ch=ss.nextInt();
switch(ch)
{
case 1:
r.printArea();
break;
```

```
case 2:  
t.printArea();  
break;  
case 3:  
c.printArea();  
break;  
case 4:  
System.exit(0);  
break;  
default:  
System.out.println("Invalid choice!");  
}  
}  
}  
}
```

## OBSERAVATION:

OOJ

OOJ Lab Exercise - 4 and 5

[xler 8]

→) Lab-4 :-

CODE :-

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

class Rectangle extends Shape
{
    void printArea()
    {
        Scanner ss = new Scanner(System.in);
        System.out.println("Enter length and breadth of the
rectangle ");
        a=ss.nextInt();
        b=ss.nextInt();
        double area;
        area=(double) a*b;
        System.out.println("The area of Rectangle is "+area);
    }
}
```

```
}
```

```
}
```

```
class Triangle extends Shape
```

```
{
```

```
    void printArea()
```

```
{
```

```
    Scanner ss = new Scanner (System.in);  
    System.out.println ("Enter base length and  
height of the triangle");
```

```
    a = ss.nextInt();
```

```
    b = ss.nextInt();
```

```
    double area;
```

```
    area = (double) 0.5 * a * b;
```

```
    System.out.println ("The area of triangle is " + area);
```

```
}
```

```
class Circle extends Shape
```

```
{
```

```
    void printArea()
```

```
{
```

```
    Scanner ss = new Scanner (System.in)
```

```
    System.out.println ("Enter radius of the circle");
```

```
    a = ss.nextInt();
```

```
    double area;
```

```
    area = (double) 3.14 * a * a;
```

```
System.out.println ("The area of Circle is "
+ area);
```

```
}
```

```
}
```

```
class ShapeMain
```

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

```
    int ch;
```

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

```
    Rectangle r = new Rectangle ();
```

```
    Triangle t = new Triangle ();
```

```
    Circle c = new Circle ();
```

```
    while (true) {
```

```
        System.out.println ("Enter the choice of shape whose
area has to be calculated");
```

```
        System.out.println ("1. Rectangle\n2. Triangle\n3. Circle
\n4. Exit");
```

```
        ch = ss.nextInt();
```

```
        switch (ch)
```

```
{
```

```
    case 1:
```

```
        r.printArea ();
```

```
        break;
```

case 2:

t.printArea();  
break;

case 3:

c.printArea();  
break;

case 4:  
System.exit(0);

break;

default:

System.out.println("Invalid choice!");

}

}

}

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>javac Shapemain.java

C:\Users\win10\Documents\Java lab programs>java Shapemain
Enter the choice of shape whose area has to be calculated
1.Recatngle
2.Triangle
3.Circle
4.Exit
1
Enter length and breadth of the rectangle
4 5
The area of Recatngle is 20.0
Enter the choice of shape whose area has to be calculated
1.Recatngle
2.Triangle
3.Circle
4.Exit
2
Enter base length and height of the triangle
7 9
The area of Triangle is 31.5
Enter the choice of shape whose area has to be calculated
1.Recatngle
2.Triangle
3.Circle
4.Exit
3
Enter radius of the circle
8
The area of Circle is 200.96
Enter the choice of shape whose area has to be calculated
1.Recatngle
2.Triangle
3.Circle
4.Exit
4
```

## **LAB PROGRAM-5**

**Develop a Java program to create a class Bank that maintains two kinds of account for it 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:**

- Accept deposit from customer and update the balance.**
- Display the balance.**
- Compute and deposit interest**
- Permit withdrawal and update the balance**
- Check for the minimum balance, impose penalty if necessary and update the balance**

### **SOURCE CODE:**

```
import java.util.Scanner;
abstract class Account {
    String cName, accType;
    long accNo;
    double bal;
    final double minBal = 1000.0;
    Account(String cName, long accNo, double bal, String accType) {
        this.accNo = accNo;
        this.cName = cName;
        this.bal = bal;
        this.accType = accType;
    }
    abstract void addBal(double amt);
    abstract void dispBal();
    abstract void withBal(double amt);
}
class Curr_acct extends Account {
```

```
Curr_acct(String cName, long accNo, double bal) {  
super(cName, accNo, bal, "Current");  
System.out.println("Name: "+cName+"\taccno: "+accNo+"\tbl: "+bal+"\tttype:  
"+accType);  
}  
void addBal(double amt){  
this.bal += amt;  
}  
void dispBal(){  
System.out.println("Your balance is: " + this.bal);  
}  
void checkBal() {  
if (this.bal < minBal) {  
System.out.println("Insufficient balance, penalty imposed");  
this.bal -= this.bal * 0.02;  
}  
}  
void withBal(double amt){  
this.bal -= amt;  
checkBal();  
}  
}
```

```
class Sav_acct extends Account {  
    Sav_acct(String cName, long accNo, double bal) {  
        super(cName, accNo, bal, "Savings");  
        System.out.println("name: " + cName + "\taccno: " + accNo + "\tbl: " + bal +  
                           "\tttype: " + accType);  
    }  
    void addBal(double amt){  
        this.bal += amt;  
        addIntr();  
    }  
    void addIntr() {  
        this.bal += this.bal * 0.07;  
    }  
    void dispBal(){  
        System.out.println("Your balance is: " + this.bal);  
    }  
    void withBal(double amt){  
        this.bal -= amt;  
    }  
}
```

```
}

class Bank {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
Double amt;
System.out.println("Enter your details:");
System.out.println("Name:");
String x=sc.next();
System.out.println("Account Number:");
long y=sc.nextLong();
for(;;)
{
System.out.println("Type of account:\n1.Current account\n2.Savings
account\n3.Exit");
int t=sc.nextInt();
if(t==1){
System.out.println("The current account provides cheque book facility but no
interest.");
Curr_acct c = new Curr_acct(x, y, 50000);
for(;;)
{
```

```
System.out.println("1:Deposit\n2:Display Balance\n3:Withdraw\n4:Exit");
int ch = sc.nextInt();
switch (ch) {
case 1:
System.out.println("Enter the amount to be added:");
amt = sc.nextDouble();
c.addBal(amt);
break;
case 2:
c.dispBal();
break;
case 3:
System.out.println("Enter the amount to be withdrawn:");
amt = sc.nextDouble();
c.withBal(amt);
break;
case 4:System.exit(0);
default:System.out.println("Invalid choice! Try again");
}
}
}
```

```
else if(t==2){  
System.out.println("The savings account provides compound interest and  
withdrawal facilities but no cheque book facility.");  
Sav_acct s = new Sav_acct(x, y, 5000);  
for(;;){  
System.out.println("1:Deposit\n2:Display Balance\n3:Withdraw\n4:Exit");  
int ch = sc.nextInt();  
switch (ch) {  
case 1:  
System.out.println("Enter the amount to be added:");  
amt = sc.nextDouble();  
s.addBal(amt);  
break;  
case 2:  
s.dispBal();  
break;  
case 3:  
System.out.println("Enter the amount to be withdrawn:");  
amt = sc.nextDouble();  
s.withBal(amt);  
break;  
case 4:System.exit(0);  
default:System.out.println("Invalid choice! Try again");  
}  
}
```

```
    }
}
else if(t==3)
System.exit(0);
else
System.out.println("Invalid choice! Try again");
}
}
}
```

**OBSERVATION:**

\* Lab-5 :-

```
import java.util.*;  
abstract class Account {  
    String cName, accType;  
    long accNo;  
    double bal;  
    final double minBal = 1000.0;  
    Account (String cName, long accNo, double bal,  
             String accType)  
    {  
        this.accNo = accNo;  
        this.cName = cName;  
        this.bal = bal;  
        this.accType = accType;  
    }  
    abstract void addBal (double amt);  
    abstract void dispBal();  
    abstract void withdraw (double amt);  
}  
class Curr_acct extends Account {  
    Curr_acct (String cName, long accNo,  
               double bal)
```

```
{  
    super(cName, accNo, bal, "current");  
    System.out.println("Name : " + cName + "\t"  
accno : "+ accNo + "\t" bal : "+ bal + "\t" type : "+ accType);  
}  
  
void addBal (double amt) {  
    this.bal += amt;  
}  
  
void dispBal () {  
    System.out.println ("your balance is : "+  
        .this.bal);  
}  
  
void checkBal () {  
    if (this.bal < minBal) {  
        System.out.println ("Insufficient balance, penalty  
imposed");  
        this.bal -= this.bal * 0.02;  
    }  
}  
  
void withdraw (double amt) {  
    this.bal -= amt;  
    checkBal();  
}  
}
```

```
class Savacct extends Account {  
    Savacct (String cName, long accNo, double bal) {  
        super(cName, accNo, bal, "Savings");  
        System.out.println ("name : " + cName + "\t"  
                           "accno : " + accNo + "\t" + bal + "\t" + type : "+  
                           accType);  
    }  
    void addBal (double amt) {  
        this.bal += amt;  
        addIntx();  
    }  
    void addIntx () {  
        this.bal += this.bal * 0.07;  
    }  
    void dispBal () {  
        System.out.println ("Your balance is " + this.bal);  
    }  
    void withBal (double amt) {  
        this.amt -= amt;  
    }  
}
```

```
class Bank {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        Double amt;
        System.out.println ("Enter your details:");
        System.out.println ("Name : ");
        String x = sc.next();
        System.out.println ("Account Number : ");
        long y = sc.nextLong();
        for (;;) {
            System.out.println ("Type of account :\n1. Current account\n2. Savings account\n3. Exit");
            int t = sc.nextInt();
            if (t == 1) {
                System.out.println ("The current account provides cheque book facility but no interest");
                Curr_act c = new Curr_act (x, y, 5000);
                for (;;) {
                    System.out.println ("1. Deposit\n2. Display Balance\n3. Withdraw\n4. Exit");

```

```
int ch = sc.nextInt();
```

```
switch(ch) {
```

case 1 :

```
    system.out.println("Enter the amount to be  
added");
```

```
    amt = sc.nextDouble();
```

```
    c.addBal(amt);
```

```
    break;
```

case 2 :

```
    c.dispBal();
```

```
    break;
```

case 3 :

```
    system.out.println("Enter the amount to be  
withdrawn");
```

```
    amt = sc.nextDouble();
```

```
    c.withBal(amt);
```

```
    break;
```

case 4 :

```
    System.exit(0);
```

default:

```
    system.out.println("Invalid choice! Try again");
```

```
}
```

else if ( $t == 2$ ) {

System.out.println ("The savings account provides compound interest and withdrawal facilities but no cheque book facility");

Sav-act s = new Sav-act (x, y, 5000);

for (;;) {

{

System.out.println ("1. Deposit\n2. Display Balance\n3. Withdrawal\n4. Exit");

int ch = sc.nextInt();

switch (ch) {

case 1:

System.out.println ("Enter the amount to be added");

amt = sc.nextDouble();

s.addBal (amt); break;

case 2: c.dispBal();

break;

case 3: System.out.println ("Enter the amount to be withdrawn");

amt = sc.nextDouble();

c.withBal (amt);

break;

case 4: System.exit(0);

default:

System.out.println("Invalid choice! Try again");

}

}

}

else if (t == 3)

System.exit(0);

else

System.out.println("Invalid choice! Try again");

}

}

}

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>java Bank
Enter your details:
Name:
fgh
Account Number:
789
Type of account:
1.Current account
2.Savings account
3.Exit
2
The savings account provides compound interest and withdrawal facilities but no cheque book facility.
name: fgh      accno: 789      bal: 5000.0      type: Savings
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
1
Enter the amount to be added:
1000
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
2
Your balance is: 6420.0
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
3
Enter the amount to be withdrawn:
100
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
2
Your balance is: 6320.0
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
4
```

```
C:\Users\win10\Documents\Java lab programs>java Bank
Enter your details:
Name:
abc
Account Number:
123
Type of account:
1.Current account
2.Savings account
3.Exit
1
The current account provides cheque book facility but no interest.
Name: abc      accno: 123      bal: 50000.0    type: Current
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
1
Enter the amount to be added:
1000
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
2
Your balance is: 51000.0
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
3
Enter the amount to be withdrawn:
50500
Insufficient balance, penalty imposed
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
2
Your balance is: 490.0
1:Deposit
2:Display Balance
3:Withdraw
4:Exit
4
```

## **LAB PROGRAM-6**

**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.**

### **SOURCE CODE:**

***Below is two programs of class Student and class Internals. Both belong to a package named CIE. Class Internals extends from class Student.***

```
package CIE;
import java.util.Scanner;

public class Student
{
    public String name;
    public String usn;
    public int sem;
    public void display()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Name:");
        name=s.next();
        System.out.println("USN:");
        usn=s.next();
        System.out.println("Semester:");
        sem=s.nextInt();
    }
}
package CIE;
import java.util.Scanner;
```

```

public class Internals extends Student
{
    public double ciem[];

    public void display()
    {
        ciem=new double[5];
        Scanner t=new Scanner(System.in);
        System.out.println("CIE Marks for 5 subjects(out of 50):");
        for(int i=0;i<5;i++)
            ciem[i]=t.nextDouble();
    }
}

```

***Below is a program of class Externals which belongs to a package named SEE. Class Externals extends from class Student.***

```

package SEE;
import java.util.*;
import CIE.*;

public class Externals extends CIE.Student
{
    public double seem[];

    public void display()
    {
        seem=new double[5];
        Scanner s=new Scanner(System.in);
        System.out.println("SEE Marks for 5 subjects(out of 100):");
        for(int i=0;i<5;i++)
            seem[i]=s.nextDouble();
    }
}

```

***Below is a driven program with class Main which imports both the packages CIE and SEE.***

```

import CIE.*;
import SEE.*;
import java.util.Scanner;

```

```
public class Main
{
    public static void main(String args[])
    {
        int n;
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the number of students:");
        n=s.nextInt();
        CIE.Student st[]=new CIE.Student[n];
        CIE.Internals in[]=new CIE.Internals[n];
        SEE.Externals e[]=new SEE.Externals[n];
        for(int i=0;i<n;i++)
        {
            st[i]=new CIE.Student();
            in[i]=new CIE.Internals();
            e[i]=new SEE.Externals();
            st[i].display();
            in[i].display();
            e[i].display();
            System.out.println("Total marks of student "+st[i].name+" in 5
subjects are:");
            for(int j=0;j<5;j++)
            {
                System.out.println(in[i].ciem[j]+(e[i].seem[j]/2));
            }
        }
    }
}
```

## OBSERVATION:

### Week - 9 - Lab Program - 6

\* ) Below is two programs of class Student and class Internals . Both belong to a package named CIE . Class Internals extends from class Student .

①

```
package CIE;  
import java.util.Scanner;  
  
public class Student  
{  
    public String name;  
    public String USN;  
    public int sem;  
    public void display()
```

L

```
Scanner s = new Scanner (System.in);  
System.out.println ("Name : ");  
name = s.next();  
System.out.println ("USN : ");  
USN = s.next();  
System.out.println ("Semester : ");  
sem = s.nextInt();
```

} }

② package CIE;  
import java.util.Scanner;  
public class Internals extends Student  
{  
 public double ciem[];  
 public void display()  
 {  
 ciem = new double [5];  
 Scanner t = new Scanner (System.in);  
 System.out.println ("CIE marks for 5 subjects  
(out of 50):");  
 }  
}

\* Below is a program of class Internals which belongs to package SEE. Class Internals extends from class Student.

③ package SEE;  
import java.util.\*;  
import CIE.\*;

```
public class SEEinternals extends CIE.Student
{
    public double seem[];
    public void display()
    {
        seem = new double [5];
        Scanner s = new Scanner (System.in);
        System.out.println ("SEE Marks for 5 subjects (out of 100):");
        for (int i=0; i<5; i++)
            seem[i] = s.nextDouble();
    }
}
```

→ Below is a driven program with class Main which imports both the packages CIE and SEE.

④

```
import CIE.*;
import SEE.*;
import java.util.Scanner;
public class Main
```

```
public static void main( String args[] )  
{  
    int n;  
    Scanner s = new Scanner( System.in );  
    System.out.println( "Enter the number of  
students: " );  
    n = s.nextInt();  
    CIE.Student st[] = new CIE.Student[n];  
    CIE.Internals in[] = new CIE.Internals[n];  
    SEE.Externals e[] = new SEE.Externals[n];  
    for (int i=0; i<n; i++)  
    {  
        st[i] = new CIE.Student();  
        in[i] = new CIE.Internals();  
        e[i] = new SEE.Externals();  
        st[i].display();  
        in[i].display();  
        e[i].display();  
        System.out.println( "Total marks of  
student " + st[i].name + " in 5 subjects are: " );  
        for (int j=0; j<5; j++)  
            System.out.println( in[i].ciem[j] + ( e[i].sem[j] / 2 ));  
    }  
}
```

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs\Packages>javac Student.java  
C:\Users\win10\Documents\Java lab programs\Packages>javac Internals.java  
C:\Users\win10\Documents\Java lab programs\Packages>javac Externals.java  
C:\Users\win10\Documents\Java lab programs\Packages>javac Main.java  
  
C:\Users\win10\Documents\Java lab programs\Packages>java Main  
Enter the number of students:  
2  
Name:  
Ajay  
USN:  
1BM19CS001  
Semester:  
3  
CIE Marks for 5 subjects(out of 50):  
41  
42  
43  
44  
45  
SEE Marks for 5 subjects(out of 100):  
80  
81  
82  
83  
84  
Total marks of student Ajay in 5 subjects are:  
81.0  
82.5  
84.0  
85.5  
87.0
```

Name:  
Vijay  
USN:  
1BM19CS002  
Semester:  
3  
CIE Marks for 5 subjects(out of 50):  
50  
40  
50  
45  
50  
SEE Marks for 5 subjects(out of 100):  
90  
100  
95  
100  
93  
Total marks of student Vijay in 5 subjects are:  
95.0  
90.0  
97.5  
95.0  
96.5

---

---

## **LAB PROGRAM-7**

**7. Write a program to demonstrate generics with multiple object parameters.**

### **SOURCE CODE:**

```
import java.util.*;
class Gener<T,U,V>
{
    T usn;
    U attendance;
    V cgpa;
    Gener(T n,U a,V c)
    {
        usn = n;
        attendance = a;
        cgpa = c;
    }
    void display()
    {
        System.out.println("=====");
        System.out.println("USN of student: "+usn);
        System.out.println("Attendance = "+attendance);
        System.out.println("CGPA = :" +cgpa);
    }
}
class Generics
{
    public static void main(String args[])
    {
        Scanner in = new Scanner(System.in);
        String USN;
        int attd;
        double cg;
        System.out.println("Enter the USN of the student:");
        USN = in.next();
        System.out.println("Enter the attendance % of the student:");
        attd = in.nextInt();
```

```
        System.out.println("Enter the CGPA of the student:");
        cg = in.nextDouble();
        Gener<String, Integer, Double> ob = new Gener<String, Integer,
Double>(USN, attd, cg);
        ob.display();
    }
}
```

## OBSERVATION:

### Week-10 Lab Programs 7 and 8

#### → Lab program - 7 :-

```
import java.util.*;  
class Gener<T,U,V>  
{  
    T usn;  
    U attendance;  
    V cgpa;  
    Gener(T n,U a,V c)  
    {  
        usn = n;  
        attendance = a;  
        cgpa = c;  
    }
```

```
    void display()  
    {
```

```
        System.out.println("=====");  
        System.out.println("USN of student : "+usn);  
        System.out.println("Attendance : "+attendance);  
        System.out.println("CGPA : "+cgpa);  
    }
```

class Generics

{ public static void main(String args[])

{ Scanner in = new Scanner(System.in);  
String USN;  
int attd;  
double cg;

System.out.println("Enter the USN of the student");

USN = in.nextInt();

System.out.println("Enter the attendance % of the  
student");

attd = in.nextInt();

System.out.println("Enter CGPA of the student");

cg = in.nextDouble();

Generics<String, Integer, Double> ob = new Generics<  
String, Integer, Double>(USN, attd, cg);  
ob.display();

}

}

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>javac Generics.java
C:\Users\win10\Documents\Java lab programs>java Generics
Enter the USN of the student:
1BM19CS001
Enter the attendance % of the student:
90
Enter the CGPA of the student:
9.7
=====
USN of student: 1BM19CS001
Attendance = 90
CGPA = :9.7
```

## **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.**

### **SOURCE CODE:**

```
import java.util.Scanner;
class WrongAge extends Exception {
int age;
WrongAge(int x) {
age = x;
}
public String toString() {
return "AGE OF FATHER=" + age + " IS ENTERED INCORRECTLY";
}
}
class WrongAgeSon extends Exception {
int age;
WrongAgeSon(int x) {
age = x;
}
public String toString() {
return "AGE OF SON=" + age + " IS ENTERED INCORRECTLY";
}
}

class Father {
int a;
Father(int x) {
a = x;
}
void check() throws WrongAge {
if (a<0) {
throw new WrongAge(a);
}
}
```

```
}

}

class Son extends Father {
    int age;
    Son(int fage,int sage) {
        super(fage);
        age = sage;
    }
    void compute() throws WrongAgeSon {
        if (age >= a) {
            throw new WrongAgeSon(age);
        } else {
            System.out.println("THE AGES ARE ENTERED CORECTLY");
            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.check();
        } catch (WrongAgeSon e) {
            System.out.println(e);
        }
        try {
            ss.compute();
        } catch (WrongAge e) {
            System.out.println(e);
        }
    }
}
```

## OBSERVATION:

Week -10

### Modified Lab Program -8.



```
import java.util.Scanner;  
class WrongAge extends Exception {  
    int age;  
    WrongAge(int x) {  
        age = x;  
    }  
    public String toString() {  
        return "Age of Father = "+age+" is entered incorrectly";  
    }  
}  
class WrongAgeSon extends Exception {  
    int age;  
    WrongAgeSon(int x) {  
        age = x;  
    }  
    public String toString() {  
        return "Age of Son = "+age+" is entered incor-  
-tly";  
    }  
}
```

```
class Father {
```

```
    int a;
```

```
    Father (int x) {
```

```
        a = x;
```

```
}
```

```
void check() throws WrongAge {
```

```
    if (a < 0)
```

```
        throw new WrongAge(a);
```

```
} }
```

```
class Son extends Father {
```

```
    int age;
```

```
    Son (int fage, int sage) {
```

```
        super(fage);
```

```
        age = sage;
```

```
}
```

```
void compute() throws WrongAgeSon {
```

```
    if (age >= a) {
```

```
        throw new WrongAgeSon(age);
```

```
}
```

```
    else {
        System.out.println("The ages are entered correctly");
        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.check();
        } catch (WrongAgeSon e) {
            System.out.println(e);
        }
    }
}
```

```
    catch (WrongAge e) {
```

```
        System.out.println (e);
```

```
}
```

```
}
```

```
}
```

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>javac ExceptionsMain.java
C:\Users\win10\Documents\Java lab programs>java ExceptionsMain
ENTER FATHER'S AGE:
40
ENTER SON'S AGE:
20
THE AGES ARE ENTERED CORECTLY
FATHER'S AGE=40 SON'S AGE=20
```

```
C:\Users\win10\Documents\Java lab programs>javac ExceptionsMain.java
C:\Users\win10\Documents\Java lab programs>java ExceptionsMain
ENTER FATHER'S AGE:
-10
ENTER SON'S AGE:
20
AGE OF FATHER=-10 IS ENTERED INCORRECTLY
```

```
C:\Users\win10\Documents\Java lab programs>javac ExceptionsMain.java
C:\Users\win10\Documents\Java lab programs>java ExceptionsMain
ENTER FATHER'S AGE:
10
ENTER SON'S AGE:
20
AGE OF SON=20 IS ENTERED INCORRECTLY
```

## **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.**

### **SOURCE CODE:**

```
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 (InterruptedException e) {  
            System.out.println(name + "Interrupted");  
        }  
        System.out.println(name + " exiting.");  
    }  
}  
class Threadmain {  
    public static void main(String args[]) {  
        Thread1 t1=new Thread1("BMS COLLEGE OF ENGINEERING",10000);  
        Thread1 t2=new Thread1("CSE",2000);  
    }  
}
```

## OBSERVATION:

Wuk - 11

Lab program - 9



```
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 (InterruptedException e) {  
            System.out.println(name + " Interrupted");  
        }  
        System.out.println(name + " exiting.");  
    }  
}  
  
class Threadmain {  
    public static void main (String args[]) {  
        Thread t1 = new Thread ("BMS College  
of Engineering", 10000);  
        Thread t2 = new Thread ("CSE", 2000);  
    }  
}
```

**OUTPUT: In the output, “BMS College of Engineering” gets printed once for every ten seconds and “CSE” once for every two seconds.**

```
C:\Users\win10\Documents\Java lab programs>javac Threadmain.java  
C:\Users\win10\Documents\Java lab programs>java Threadmain  
thread:Thread[BMS COLLEGE OF ENGINEERING,5,main]  
thread:Thread[CSE,5,main]  
BMS COLLEGE OF ENGINEERING  
CSE  
CSE  
CSE  
CSE  
CSE  
BMS COLLEGE OF ENGINEERING  
CSE exiting.  
BMS COLLEGE OF ENGINEERING  
BMS COLLEGE OF ENGINEERING  
BMS COLLEGE OF ENGINEERING  
BMS COLLEGE OF ENGINEERING exiting.
```

## 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.

### SOURCE CODE:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class integerdivision extends Frame implements ActionListener{
    TextField n1,n2,res;
    Label ln1,ln2,lres;
    Button b;
    public integerdivision(){
        setLayout(new FlowLayout());
        Label ln1=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(b);
        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 ){  
        JOptionPane.showMessageDialog(this,ne,"ERROR",  
JOptionPane.ERROR_MESSAGE);  
    }  
    catch(ArithmeticException a){  
        JOptionPane.showMessageDialog(this,a,"ERROR",  
JOptionPane.ERROR_MESSAGE);  
    }  
}  
}  
  
public static void main(String args[]){  
    integerdivision i=new integerdivision();  
    i.setSize(new Dimension(400,400));  
    i.setTitle("INTEGER DIVISION OF TWO NUMBERS");  
    i.setVisible(true);  
}  
  
class WindowAdapter1 extends WindowAdapter{  
    public void windowClosing(WindowEvent we)  
    {  
        System.exit(0);  
    }  
}
```

OBSERVATION:

Week - 12

Lab Program - 10

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class integrdivision extends Frame
implements ActionListener {
    TextField n1, n2, res;
    Label ln1, ln2, lres;
    Button b;
    public integrdivision() {
        setLayout(new FlowLayout());
        Label ln1 = 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(b);
    add(res);
    b.addActionListener(this);
}

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) {
            JOptionPane.showMessageDialog(this, ne, "ERROR",
                JOptionPane.ERROR_MESSAGE);
        }
    }
    catch (ArithmaticException a) {
        JOptionPane.showMessageDialog(this, a, "ERROR",
            JOptionPane.ERROR_MESSAGE);
    }
}
public static void main(String args[])
{
    integerdivision i = new integerdivision();
    i.setSize(new Dimension(400, 400));
    i.setTitle("INTEGER DIVISION OF TWO NUMBERS");
    i.setVisible(true);
}
class WindowAdapter1 extends WindowAdapter {
    public void windowClosing(WindowEvent we)
    {
        System.exit(0);
    }
}
```

## **OUTPUT:**

```
C:\Users\win10\Documents\Java lab programs>javac integerdivision.java  
C:\Users\win10\Documents\Java lab programs>java integerdivision
```



