# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.



## on

# Object Oriented Java Programming (23CS3PCOOJ)

Submitted by

MONISHA H L (24BECS411)

in partial fulfillment for the award of the degree of BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



**B.M.S. COLLEGE OF ENGINEERING** 

(Autonomous Institution under VTU) BENGALURU-560019 Sep-2024 to Jan-2025

### **B.M.S.** College of Engineering,

**Bull Temple Road, Bangalore 560019** 

(Affiliated To Visvesvaraya Technological University, Belgaum)

#### **Department of Computer Science and Engineering**



#### **CERTIFICATE**

This is to certify that the Lab work entitled "Object Oriented Java Programming (23CS3PCOOJ)" carried out by MONISHA H L (24BECS411), who is bonafide student of B.M.S. College of Engineering. It is in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements in respect of an Object Oriented Java Programming (23CS3PCOOJ) work prescribed for the said degree.

SRUSHTI C S Assistant Professor Department of CSE, BMSCE Dr. JYOTHI S NAYAK Professor & HOD Department of CSE, BMSCE

## Index

Sl. No.	Date	Experiment Title	Page No.
1	30-9-24	QUADRATIC	1-4
2	7-10-24	STUDENT_SGPA	5-8
3	14-10-24	BOOK_DETAILS	9-12
4	21-10-24	AREA_OF_SHAPE	13-15
5	28-10-24	BANK	16-22
6	11-11-24	FINAL MARKS	23-28
7	28-11-24	EXCEPTION	29-31
8	28-11-24	THREADS	32-33
9	28-11-24	INTEGER DIVISION	34-36
10	28-11-24	IPC	37-41
	28-11-24	DEADLOCK	42-45

#### **GITHUB LINK:**

https://github.com/Monisha-HL/Java Lab Programs

#### **PROGRAM 1**

Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c=0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.

```
Lab Program 1
 Develop a Java program that prints all real solutions
 to the quadratic equation azat bx+c=0. Read in a, b, c and use the quadratic formula. It has discontinued
 62- hac is regaine, display a message stating mat there
are no real bolutions.
Gode:
import java util Scanner;
bublic class Quadrake {
public static void main (Sking augs []) {
Scanner scanner = new Scanner (System.in);
double a, b, c, d, x1, x2;
System out brint ("Enter the co-efficient of a:");
a = Scanner. rext Double (); while (a=0) {
System-out. println ("Not a qualitatic equation. Please enter a non-zero value");
a. Scanner. nextDouble ();
System. out paint ( "Enter the co-efficient of 6:");
b= scanner. next Double ();
System. out. print ("Enter the co-efficient of c:");
    scanner. next Double ();
d= b*b-4*a*c;
 System out println ("Roots are real and equal");
 System out println ("Roots and Roots are " + >1);
Else if (dro) {
```

```
71 - (-6+ Marn. Egrt (d)) ((2+a);
22 = (-6+ Marn. Egrt (d)) ((2+a);
System. out frintln ("Roots are exal and distinct");
5.0.P ("Root1:" +>1);
S.O.P ( * Root 2: "+912);
5.0. P ("Roots are Imaginary");
511 = - b/(2*a);
512 = Math. squt (-d) /(2*a);
5.0.P("Real Post: " + 271);
5.0.P("Imaginary Past: " + 272);
Name Input came Input = new Name Input ();
name Input. enter Details ();
scannes. close ();
class Name Input &
 bublic void enterDetails () {
Scamer s= new Scanner (system.in);
System but. print ("Enter your Name:");
String name . S. next Line ();
System. out. faint ("Enter your USN: ");
Sking usn + s. nextline();
5.0.P("Name: " +name);
5.0.P("USN: " + ven);
 Outputs:
```

```
D Enter the coefficient of a . 2 Enter the coefficient of b : 6
   Enter the coefficient of c: 3
   Rooks are real and distinct
   Root 1: -0-6339745
   Root 2: -2.3660254
  Enter your name: Monisha
  ENER YOUR USN: 24BECS411
          Monisha
 USN: DYBECSYII
@ Enter the co-efficient of a=0
  Not a quadratic equation. Please enter a non
  value.
  Enter the coefficient of 6:3
  Enter the coefficient of c: 5
  Roots are imaginary
  Real Part: -0.3
  Imaginary part = 0-9539392
  Enter your name: Monisha
  Enter your USN: 24BECS411
  Name: Monisha
  USN: 24BECS411
```

```
import java.util.Scanner;
public class Quadratic{
public static void main(String args[]){
Scanner scanner=new Scanner(System.in);
double a, b, c, d, r1, r2;
System.out.print("Enter the coefficient of a:");
a=scanner.nextDouble();
while (a==0){
System.out.println("Not a quadratic equation. Please enter a non-zero value");
a=scanner.nextDouble();
}
System.out.print("Enter the coefficient of b:");
b=scanner.nextDouble();
System.out.print("Enter the coefficient of c:");
c=scanner.nextDouble();
d=b*b-4*a*c;
```

```
if(d==0){
r1=-b/2*a;
System.out.println("Root are real and equal");
System.out.println("Root1 and Root2 are:" +r1);
}
else if (d>0){
r1=(-b+Math.sqrt(d))/(2*a);
r2=(-b-Math.sqrt(d))/(2*a);
System.out.println("Roots are real and distinct");
System.out.println("Root1: " +r1);
System.out.println("Root2: " +r2);
else{
System.out.println("Roots are imaginary");
r1 = -b / (2 * a);
r2 = Math.sqrt(-d) / (2 * a);
System.out.println("Real part: " + r1);
System.out.println("Imaginary part: " + r2);
}
NameInput nameInput=new NameInput();
nameInput.enterDetails();
scanner.close();
}
}
class NameInput{
public void enterDetails(){
Scanner s=new Scanner(System.in);
System.out.print("Enter your Name: ");
String name=s.nextLine();
System.out.print("Enter your USN: ");
String usn=s.nextLine();
System.out.println("Name: "+name);
System.out.println("USN: " +usn);
}
}
```

D:\Sample>javac Quadratic.java D:\Sample>java Quadratic Enter the coefficient of a:2 Enter the coefficient of b:6 Enter the coefficient of c:3 Roots are real and distinct Root1: -0.6339745962155614 Root2: -2.3660254037844384 Enter your Name: Monisha H L Enter your USN: 24BECS411 Name: Monisha H L USN: 24BECS411 D:\Sample>java Quadratic Enter the coefficient of a:0 Not a quadratic equation. Please enter a non-zero value 5 Enter the coefficient of b:3 Enter the coefficient of c:5 Roots are imaginary Real part: -0.3 Imaginary part: 0.9539392014169457 Enter your Name: Monisha H L Enter your USN: 24BECS411 Name: Monisha H L USN: 24BECS411 D:\Sample>java Quadratic

D:\Sample>java Quadratic
Enter the coefficient of a:2
Enter the coefficient of b:3
Enter the coefficient of c:5
Roots are imaginary
Real part: -0.75
Imaginary part: 1.3919410907075054
Enter your Name: Monisha H L
Enter your USN: 24BECS411
Name: Monisha H L
USN: 24BECS411

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.

```
WEEK-2
                   Student SGPA
import java util Scanner;
class Student &
   Sking name:
   sking usni
  double SLAPA:
   Subject [] Subject;
   Scanner 5:
   Student () {
       subject = new subject (9);
for (int i=0; i=9; i+t) {
           Subject (it = new Subject ();
        S= new Scanner (System.in);
    Void gershudent Details () {
         System out , print ("Enter name:");
         name = S. nextline ();
         System-out-print ("Enter Usin: "),
          usn = s. nextline();
    void germontellig
      for (int 100; 168; 1++) {
             S.O. O("Enter coeditis makes for subject"+(i+));
              Subject [i] - subject marks = 5 - next9 at();
              S.O.P ("Enter Chedits for subject" + (i+)+":");
              Subject [i] . credits = 5 . rextAnt() i
              il ( Subject [i). subject Marks >= 90) {
                   Subject (i). grade = 10;
```

```
Jetse i6 (subject E) - Subject Marks >= 80) ]
                                               subject(1) - grade = 9;
                              Felse if (Rubject (: ) subject towards >= 70);
                                                   subject [] - grade = 8;
                               3 else ib (subject [i]. subject mostes >= 60)
                                                  Subject (i) grade = 7;
                                    Belse ib (subject (i) . Aubject Markes >= 50) {
                                                   subject (i) grade = 6;
                                     Fellery ( Rubject (i) subject Marks >= 40) 9
                                                    subject [i] grade = 5;
                                       3 else {
                                               subject (1) grade = 0;
void compareSGPA() {
               int total Gredits =0;
                  int total GradePoints = 0;
                  tos (intieo; ic3; i++) {
                                total outist = subject [i]. Gredits:
                                   total Grade Points += 80bject i J grade # 50bject i grade # 50bjec
              SGPA = (double) to talgonade Points / total Credits;
void display Details () {
               S.O. P("Name: "+ name);
S.O. P("USN: "+ USN);
               5.0 P( * SGPA: + SGPA);
```

```
bublic static void main (String () augu) &
           Shident x12 new Shident();
              5) get Shudent Details ();
51. get Marks ();
              St. compute scapac);
              SI. display Details ();
  class Subject &
       int subject monks;
       int credits;
       int grade;
 Enter name: Monisha
 Enter USN: 24BECS 411
 Enter marks for subject 1: 95
 Enter medin for subject 1: 5
Enter marks for subject 2:96
Enter gardin for subject 2:7
 Enter marks for Rubject 3: 93
 Enter ments for subject 3: 3
Enter marks for subject 4: 90
Enter marks for subject 4: 4
Enter marks for subject 5: 93
Enter crudits for subject 5: 3
Enter marks for subject 5: 3
Enter marks
 Enter marks for subject 6:92
Enter credits for subject 6:3
 Enter marks for subject 7: 95
                                                        Nome: Monisha
Ucn: 24BECS411
SGPA: 10.0
ence makes for subject 7: 2. Ence makes for subject 8: 90 Enter challs for subject 8: 2
```

```
import java.util.Scanner;
class Student {
  String name;
  String usn;
  double SGPA;
  Subject[] subject;
  Scanner s;
  Student() {
     subject = new Subject[9];
     for (int i = 0; i < 9; i++) {
       subject[i] = new Subject();
     }
     s = new Scanner(System.in);
  }
  void getStudentDetails() {
     System.out.print("Enter name: ");
     name = s.nextLine();
     System.out.print("Enter USN: ");
     usn = s.nextLine();
  }
  void getMarks() {
     for (int i = 0; i < 8; i++) {
       System.out.print("Enter marks for subject " + (i + 1) + ": ");
       subject[i].subjectMarks = s.nextInt();
       System.out.print("Enter credits for subject " + (i + 1) + ": ");
```

```
subject[i].credits = s.nextInt();
       if (subject[i].subjectMarks >= 90) {
          subject[i].grade = 10;
       } else if (subject[i].subjectMarks >= 80) {
          subject[i].grade = 9;
       } else if (subject[i].subjectMarks >= 70) {
          subject[i].grade = 8;
       } else if (subject[i].subjectMarks >= 60) {
          subject[i].grade = 7;
       } else if (subject[i].subjectMarks >= 50) {
          subject[i].grade = 6;
       } else if (subject[i].subjectMarks >= 40) {
          subject[i].grade = 5;
       } else {
          subject[i].grade = 0;
     }
  }
  void computeSGPA() {
     int totalCredits = 0;
     int totalGradePoints = 0;
     for (int i = 0; i < 8; i++) {
       totalCredits += subject[i].credits;
       totalGradePoints += subject[i].grade * subject[i].credits;
     SGPA = (double) totalGradePoints / totalCredits;
  }
  void displayDetails() {
     System.out.println("Name: " + name);
     System.out.println("USN: " + usn);
     System.out.println("SGPA: " + SGPA);
  }
  public static void main(String[] args) {
     Student s1 = new Student();
     s1.getStudentDetails();
     s1.getMarks();
     s1.computeSGPA();
     s1.displayDetails();
  }
class Subject {
  int subjectMarks;
  int credits;
  int grade;
```

}

## D:\Sample>javac Student.java D:\Sample>java Student Enter name: Monisha Enter USN: 24BECS411 Enter marks for subject 1: 95 Enter credits for subject 1: 5 Enter marks for subject 2: 96 Enter credits for subject 2: 7 Enter marks for subject 3: 93 Enter credits for subject 3: 3 Enter marks for subject 4: 90 Enter credits for subject 4: 4 Enter marks for subject 5: 92 Enter credits for subject 5: 2 Enter marks for subject 6: 92 Enter credits for subject 6: 3 Enter marks for subject 7: 95 Enter credits for subject 7: 2 Enter marks for subject 8: 90 Enter credits for subject 8: 2 Name: Monisha USN: 24BECS411 SGPA: 10.0 D:\Sample>javac Student.java

D:\Sample>java Student Enter name: Priva Enter USN: 24XXZZ Enter marks for subject 1: 90 Enter credits for subject 1: 2 Enter marks for subject 2: 85 Enter credits for subject 2: 5 Enter marks for subject 3: 87 Enter credits for subject 3: 3 Enter marks for subject 4: 89 Enter credits for subject 4: 3 Enter marks for subject 5: 90 Enter credits for subject 5: 2 Enter marks for subject 6: 92 Enter credits for subject 6: 3 Enter marks for subject 7: 91 Enter credits for subject 7: 4 Enter marks for subject 8: 89 Enter credits for subject 8: 2 Name: Priva

USN: 24XXZZ

SGPA: 9.458333333333334

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.

```
WEEK 3
Greate a class Book which contains four members non
author, brice, num-poges. Include a constructive to 24
the values for the members. Include methods to let
and get the details of the objects. Include a toping
method that could display the complete details of he
book. Develop a Java program to create a book object
impost java util. Scannes;
Class Book &
sking name;
String author;
int baice;
int numPages;
Book (String Name, String author, int beice, the numbages) {
this name = name;
This - author = author;
this - paice = paice;
this numPages = numPages :/
Public String to String() &

String name, author, beice, numbages;
name: "Book name: "this name + "h";
author = "Awhor name: "+ this author + " In";
brice = "Price : " + this price + " In";
numbages: "Number of Pages: " + Mis numbages + "In";
Suturn name + aumor + parce + numbages;
```

```
public static void main (thing asgres) &
System. out paintle ("Name: Monishath");
System. out paintle ("UEN: ZUBECS 41");
Scanner &= new Scanner (Syetem-in);
System. out. print (" Enter the number of books: ");
n= s.nextAntl);
Book[] be new Book[n];
for (int-1=0; i=n; i++){
System.out. paint (BEntes the name of the book: ");
String name = 8 next (),
System out faint ( Finter the name of the authors: ");
String author = s. next();
System out paint ("Enter the paice of the book: ");
int price = & next9nt();
System out paint ("takes he number of pages of the book
int numPages = s. next9nt();
s. nextline();
b[i] = new Book (name, author, price, numPages);
System. out brintln ("Book Details: ");
boe (int 1=0; icn; i++) €
System out println (bEII):
8-close ();
Output :
```

```
Name: Montsha HL
 USW: PUBECSUIT
 Enter the number of books: 2
Enter the name of the book: Hassy-Potter
Enter the author of the book: J.K. Rowling
Enter the price of the book: 250
Enter the number of pages: 325
 Enter the name of the book: The-Alchemist
 Enter me author of the book : Paulo-Coetho
 Enter the farce of the book: 400 Enter the number of Pages: 230
 Book name: Harry-Potter
 Author name: J.K. Rowling
 Parice : 200
Number of Pages: 305
 Author name: Paulo-Coelho
 Parice: 400
Number of Pages: 236
```

```
SOURCE CODE:
import java.util.Scanner;
class Book{
String name;
String author;
int price;
int numPages;
Book(String name, String author, int price, int numPages){
this.name=name;
this.author=author;
this.price=price;
this.numPages=numPages;
public String toString(){
String name, author, price, numPages;
name="Book name: "+this.name+ "\n";
author="Author name: "+this.author+ "\n";
price="Price: "+this.price+ "\n";
numPages="Number of pages: "+this.numPages+ "\n";
return name + author + price + numPages;
}
}
class Main{
public static void main(String args[]){
System.out.println("Name: Monisha H L");
```

```
System.out.println("USN: 24BECS411");
Scanner s=new Scanner(System.in);
System.out.print("Enter the number of books: ");
n=s.nextInt();
Book[] b=new Book[n];
for(int i=0; i<n; i++){
System.out.print("Enter the name of the book: ");
String name=s.next();
System.out.print("Enter the name of the author: ");
String author=s.next();
System.out.print("Enter the price of the book: ");
int price=s.nextInt();
System.out.println("Enter the number of pages of the book: ");
int numPages=s.nextInt();
s.nextLine();
b[i]=new Book(name, author, price, numPages);
}
System.out.println("Book Details: ");
for(int i=0; i<n; i++){
System.out.println(b[i]);
}
s.close();
}
```

```
D:\Sample>java Main
Name: Monisha H L
USN: 24BECS411
Enter the number of books: 2
Enter the name of the book: Harry_Potter
Enter the name of the author: J_K_Rowling
Enter the price of the book: 250
Enter the number of pages of the book:
325
Enter the name of the book: The_Alchemist
Enter the name of the author: Paulo_Coelho
Enter the price of the book: 400
Enter the number of pages of the book:
230
Book Details:
Book name: Harry_Potter
Author name: J_K_Rowling
Price: 250
Number of pages: 325
Book name: The_Alchemist
Author name: Paulo_Coelho
Price: 400
Number of pages: 230
```

D:\Sample>java Main Name: Monisha H L USN: 24BECS411

Enter the number of books: 1

Enter the name of the book: Alice\_in\_Wonderland Enter the name of the author: Lewiss\_Caroll

Enter the price of the book: 200

Enter the number of pages of the book:

150

Book Details:

Book name: Alice\_in\_Wonderland Author name: Lewiss\_Caroll

Price: 200

Number of pages: 150

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.

```
WEEK-4
 Develop a java peogram to create an abstract class
 named Shape that contains two integers and an
 empty method named paintAtreal) - Provide there
Classes named Rectangle, Toplangle and Circle such
 that each one of the classes extends the class Shape
Each one of the classes contain only the method beinflived that beints the over of the given shape.
import java util Scanner;
obstract class shape &
int x, y;
abstract void brintAssal);
class Rectangle extends Shape {
Rectangle (int length, int breadth) {
this. 2 = length;
this - y = baladh;
void print Area () {
System out println ("Axea of Rectangle: "+(xxy));
class Taiangle extens Shape &
Talongle (int base, int height) {
this x - base;
this y height;
void baintAsea(){
System. our prints ("Area of Talongle: "+ (0.5 * a "y));
```

```
class Circle extends Shape &
CARCLE (int ladius) &
this . z = hadius;
void printAnea() {
System out printin ("Asua of Circle: "+ (Math.PI * x * x));
public class Asea of Shape &
Public Rhatic void main (exting augus (1) {
System-our beintly ("Nome: Manisha");
System. out. println ("USN: 24BECS411");
Scanner &= new Scanner (System.in);
System out paint ("Enter me value of length: ");
int length: s. next9nt();
System-out print ("Enter me value of berandth: ");
int breadth = s. nextAnt();
Shape a = new Rectangle (length, beeadth);
9- print Amalli
System. out. paint ("Enter the value of base: ");
int base = s next9 ntc);
System- outprint (" Enter the value of higher: ");
Int height = 5. next9n+();
Shape t = new Talongle (base, height);
t. printAsea ();
System. our - print ("Enter me value of nadius:");
int sadius = Enextants;
Shape c= new circle (sodius);
c. paint Areacl
3. dosell.
```

```
Duthert:

Nome: Monisher

Us n: 24GECS41

Enter the value of breath: 5

Area of Rectangle: 10

Enter the value of base: 4

Enter the value of height: 5

Area of Triangle: 10.0

Enter the value of radius: 5

Area of tricle: 78.53981
```

```
import java.util.Scanner;
abstract class Shape {
int x, y;
abstract void printArea();
}
class Rectangle extends Shape {
Rectangle(int length, int breadth) {
this.x = length;
this.y = breadth;
}
void printArea() {
System.out.println("Area of Rectangle: " + (x * y));
}
class Triangle extends Shape {
Triangle(int base, int height) {
this.x = base;
this.y = height;
}
void printArea() {
System.out.println("Area of Triangle: " + (0.5 * x * y));
}
class Circle extends Shape {
Circle(int radius) {
this.x = radius;
}
void printArea() {
System.out.println("Area of Circle: " + (Math.PI * x * x));
}
```

```
}
public class AreaOfShape {
public static void main(String[] args) {
System.out.println("Name: Monisha");
System.out.println("USN: 24BECS411");
Scanner s = new Scanner(System.in);
System.out.print("Enter the value of length: ");
int length = s.nextInt();
System.out.print("Enter the value of breadth: ");
int breadth = s.nextInt();
Shape r = new Rectangle(length, breadth);
r.printArea();
System.out.print("Enter the value of base: ");
int base = s.nextInt();
System.out.print("Enter the value of height: ");
int height = s.nextInt();
Shape t = new Triangle(base, height);
t.printArea();
System.out.print("Enter the value of radius: ");
int radius = s.nextInt();
Shape c = new Circle(radius);
c.printArea();
s.close();
}
}
```

```
D:\Sample>javac AreaOfShape.java

D:\Sample>java AreaOfShape
Name: Monisha
USN: 24BECS411
Enter the value of length: 2
Enter the value of breadth: 5
Area of Rectangle: 10
Enter the value of base: 4
Enter the value of height: 5
Area of Triangle: 10.0
Enter the value of radius: 5
Area of Circle: 78.53981633974483

D:\Sample>
```

```
D:\Sample>java AreaOfShape
Name: Monisha
USN: 24BECS411
Enter the value of length: 6
Enter the value of breadth: 7
Area of Rectangle: 42
Enter the value of base: 8
Enter the value of height: 7
Area of Triangle: 28.0
Enter the value of radius: 12
Area of Circle: 452.3893421169302
```

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 Cur-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
- e) Check for the minimum balance, impose penalty if necessary and update the balance.

```
WEEK 5
Develop a Java peogram to create a class Bank hos
maintains two kinds of account for its customers, one
called savings account and the other current account
The savings account provides compound interest and
windrawal facilities but no cheque book facility
The current account provides chaque book facility
but no interest.
Custer account holders should also maintain a mini
- mum balance and if the balance falls below this
level, a service charge is imposed.
* Create a class Account that stones customer name
account number and type of account. From this da
-ive the classes curacct and Savacct to make
them more specific to their sequisements.
Include the necessary methods in order to achieve
the following tasks:
a) Accept deposit from customer and update the balance
WDisplay the balance
1) Compute the deposit interest
a) Permit withdrawal and update the balance
e) Check for the minimum balance, impose beneatly it
necessary and update the balance.
import java util - Scanner;
class Account {
String Euromenome;
String Accountnumber;
String Account Tyles
```

```
double Balance;
public Account (String Curhomentians, String Accountshinks, String Accountstyle, clouble Relance) &
this. Customer Name : Customer Name;
this. Account Number = Account Number;
this. Account Type = Account Type;
this . Balance = Balance ;
public void deposit (double amount) &
Balance + = amount;
S-0.7 ("Deposit successful. Updated Balance = " + Balance);
public void display Balance () {
S.O. P ("Current Balance: "+ Balance);
class Sav-acct extends Account {
double interest Rate = 0.05;
bublic Saw-acct (String CustomerName, String Account number, double Balance) {
Super (Customer Name, Account Number, "Savings", Balance);
bublic void combuteAndDepositInterest() {
couble interest > Balance interestRate;
Brown = interest;
S.O.P ("Anterest Deposited Updated Balance: "+ Balance);
Public void withdraw (double amount) f
it (amount < = Balance) {
5.00 (" windlews successful. Updated Balance: "+ Balance);
```

```
else & S. O. P ( "Ansufficient Balance! ");
class Curacct extends Account &
double minimumBalance= 500.0;
double serviceCharge= 50.0;
public Curacet (String Customer Name, String Accountlymbe
                double Balance) {
Super (Customer Name, Accounthumber, "Custent", Balance).
bublic void checkminimum Balance () {
 it (Balance = minimum Balance) {
Balance -= service (harge;
 S.O.7( "Balance below minimum. Service change imposed.
          Updated Balance: "+ Balance);
else E
S.O. P ("Minimum Balance maintained");
public void windsaw (double amount) ?
 if (amount <= Balance) {
 S.O.P ("Windrawal successful. Upharea Balance: "+ Balance);
 CheckMinimumBalance ();
S. O.P ("Insufficient Balance!");
```

```
public class Bonk &
public static void main ( sking anga ( )) {
Scanner 8 = new Scanner (System. in);
 S. OP (" EATER Customer name: ");
String (ustomerName = s. nextlinel);
$ 0.0 ("Enter account number");
String AccountNumber = 8-nextline();
S.OP ("Enter initial Balance:");
double initial Balance = 8-nextDouble();
5.0 P ("Enter account type (Savings (wount):");
Stelling Accountingle = s. next();
Account account;
 it ( Accounttype . equals Ignone Case ("Savings")) {
 account = new Sou acct (CustomerName, AccountNum)
account = new Cus_acct (CustomerName, AccountNumber,
while (true) &
S.O.P("1. Deposit 2. Display 3. Compute Street (Sovings) 4. Wilhdraw S. Exit")
S. D.P ("ENTER your choice: ");
Int choice = & next In();
case 1: SOP ( " Entre amount to deposit: ");
        double deposit Amount = s-next Double ();
```

```
account deposit (deposit Amount);
         bseak;
Case 2: account disping Balancel);
        becak:
case 3 : if ( account historica of Saw-acet) {
        ((Savacet) account) · computer And Deposit Intocess().
        else &
        5.0.7( Antexest computation is not applicable
                for Current accounts: ");
        break;
case 4: 5.0.P("Enter amount to withdraw: ");
        double withdrawAmount = 8. nextDoublely;
        if ( account instance of Sav-Acct) &
        ((Sov-acct) account). withdraw (withdraw Amount);
        ((Curacet) account). withdraw (withdraw Amount);
       1
        break;
case 5: S.O.P("Exited .");
  System exit(0);
default: S.O.P ("Anvalid Choia!");
```

```
Output: Enter Customer name: Nomina
Enter account number: 123456
Enter initial balance: 5000
Enter account Type (sowings (corners): Sowings
1. Deposit 2. Display 3. Compute Interest (Sovings) 4 winds
s. Exit .
Enter your choice: 1
Enter amount to deposit: 500
Deposit successful . Updated bolonce = 5500.0
Enter your choice: 2
Cussent Balance: 5500.0
EARCH your choice: 3
Interest Deposited. Updated Balance: 5775.0
Enter your choice: 4
Enter your amount to winders: 500
Window successful. Updated Balance: 5275.0
Enter your choice: 5
Exited
Chasent Account:
Enter customer name: Monisha
Enter account number: 123456
Enter initial balance: 5000
Enter account Type: Current
Enter your choice: 1
Enter amount to deposit: 200
Deposit successful. Updated Balance: 5200.0
Enter your choice: 3
Interest computation is not applicable for current accounts.
taken your choice: 4
Enter amount to windraw: 4,000.0
```

```
with draw successful. Updated Bolonce: 400.0
Ratance betwee minimum. Sesures change imposed. Up.
-4 Balance = 350.0
Envex your choice: 1:
Enter amount to deposit : 500
Deposit successful. Updated Balance: 850.0
enter your choiceta:
Cusunt Balance: 850.0
Enter your choice:5
Exited.
      ice > jourc -d ... -cfr. External found
```

```
SOURCE CODE:
import java.util.Scanner;
class Account{
String CustomerName;
String AccountNumber;
String AccountType;
double Balance;
public Account(String CustomerName, String AccountNumber, String AccountType, double
Balance)
{
this.CustomerName=CustomerName;
this.AccountNumber=AccountNumber;
this.AccountType=AccountType;
this.Balance=Balance;
public void deposit(double amount){
Balance+=amount;
System.out.println("Deposit successful. Updated Balance= "+Balance);
}
public void displayBalance(){
System.out.println("Current Balance: "+Balance);
}
}
class Sav_acct extends Account{
double interestRate=0.05;
public Sav_acct(String CustomerName, String AccountNumber, double Balance){
super(CustomerName, AccountNumber, "Savings", Balance);
}
```

```
public void computeAndDepositInterest(){
double interest= Balance*interestRate;
Balance+=interest;
System.out.println("Interest Deposited. Updated Balance: "+Balance);
public void withdraw(double amount) {
if (amount <= Balance) {
Balance -= amount;
System.out.println("Withdrawal successful. Updated balance: " + Balance);
}
else {
System.out.println("Insufficient balance!");
}
}
class Cur_acct extends Account {
double minimumBalance = 500.0;
double serviceCharge = 50.0;
public Cur_acct(String CustomerName, String AccountNumber, double Balance) {
super(CustomerName, AccountNumber, "Current", Balance);
}
public void checkMinimumBalance() {
if (Balance < minimumBalance) {
Balance -= serviceCharge;
System.out.println("Balance below minimum. Service charge imposed. Updated balance: " +
Balance);
}
else {
System.out.println("Minimum balance maintained.");
}
public void withdraw(double amount) {
if (amount <= Balance) {
Balance -= amount;
System.out.println("Withdrawal successful. Updated balance: " + Balance);
checkMinimumBalance();
else {
System.out.println("Insufficient Balance!");
}
}
public class Bank{
public static void main(String args[]){
Scanner s = new Scanner(System.in);
```

```
System.out.println("Enter customer name:");
String CustomerName = s.nextLine();
System.out.println("Enter account number:");
String AccountNumber = s.nextLine();
System.out.println("Enter initial balance:");
double initialBalance = s.nextDouble();
System.out.println("Enter account type (Savings/Current):");
String AccountType = s.next();
Account account;
if (AccountType.equalsIgnoreCase("Savings")){
account = new Sav_acct(CustomerName, AccountNumber, initialBalance);
}
else {
account = new Cur_acct(CustomerName, AccountNumber, initialBalance);
}
while (true) {
System.out.println("1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5.
System.out.println("Enter your choice:");
int choice = s.nextInt();
switch (choice) {
case 1:
    System.out.println("Enter amount to deposit:");
    double depositAmount = s.nextDouble();
    account.deposit(depositAmount);
    break;
case 2:
    account.displayBalance();
    break;
case 3:
    if (account instanceof Sav_acct) {
    ((Sav acct) account).computeAndDepositInterest();
    } else {
    System.out.println("Interest computation is not applicable for Current accounts.");
    break;
case 4:
    System.out.println("Enter amount to withdraw:");
    double withdrawAmount = s.nextDouble();
    if (account instanceof Sav_acct) {
    ((Sav_acct) account).withdraw(withdrawAmount);
    } else {
    ((Cur_acct) account).withdraw(withdrawAmount);
    break;
```

```
case 5:
    System.out.println("Exited...");
    System.exit(0);
default:
    System.out.println("Invalid choice!");
}
}
}
```

#### SAVINGS ACCOUNT

```
D:\Sample>java Bank
Enter customer name:
Monisha
Enter account number:
123456
Enter initial balance:
5000
Enter account type (Savings/Current):
Savings
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Enter amount to deposit:
Deposit successful. Updated Balance= 5500.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
2
Current Balance: 5500.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Interest Deposited. Updated Balance: 5775.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
4
Enter amount to withdraw:
500
Withdrawal successful. Updated balance: 5275.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
```

```
500
Withdrawal successful. Updated balance: 5275.0

1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit Enter your choice:
5
Exited...
```

#### CURRENT ACCOUNT

```
D:\Sample>java Bank
Enter customer name:
Monisha
Enter account number:
123456
Enter initial balance:
Enter account type (Savings/Current):
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Enter amount to deposit:
Deposit successful. Updated Balance= 5200.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Current Balance: 5200.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Interest computation is not applicable for Current accounts.
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Enter amount to withdraw:
Withdrawal successful. Updated balance: 5000.0
Withdrawal successful. Updated balance: 5000.0
Minimum balance maintained.
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
Ц
Enter amount to withdraw:
4800
Withdrawal successful. Updated balance: 200.0
Balance below minimum. Service charge imposed. Updated balance: 150.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
2
Current Balance: 150.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
1
Enter amount to deposit:
Deposit successful. Updated Balance= 650.0
1. Deposit 2. Display Balance 3. Compute Interest (Savings) 4. Withdraw 5. Exit
Enter your choice:
5
Exited...
```

Create a package CIE which has two classes- Student and Internals. The class Student has members like usn, name, sem. The class Internals derived from Student 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.

puckage CIE

Public class Internals extends Student E

this internalments = internal Marks.

System out beinom ( " Internal marks : ");

for (int most : internal marks) }

system our beint (max: " ");

for (int j=0; j <5; j++) {

for ( int mark: finalmarks) &

System out printin();

System. Dut. pant (mark + " ");

final Markes [j] = meanal students [i]. Internal Markes [j] + external Students [i]. external markes [j];

System-out-faint ("Final Monks for Student "+(ight)+":");

Super ( uso, name, sem);

public vold display () {

System-out Wath 1);

super-display ();

Estimas = removed (3)

frubil - Internals (String usn, String name, int sem, into internal marks)?

```
WEEK 6
Grate a package CIE which has two classer- Student
and Internals. The Blass Student has members like
USN, name, sem. The class governals derived from
Student has an array that I three the internal morales
scored in five courses of the current semester of
the Shident. This chars
Corate another backage SEE which has the class External
which is a desired class of student. This class has
an array that stones the SEE marks scored in fire
countered of the cuspent exmester of the student.
Importation packages in a file that declares the final
marks of it students in all five courses.
package CIE;
public class Student &
   bublic sking usn;
   public int sem;
   public Student (String wan, String name, int sem) &
   this wan = wan ;
   this name = name;
   this sem = sem;
 public void display () {
 System out printly ("USN: " +USN);
 System. out-beintln ("Nome: "+ nome);
 system. out. paintin ( * Semester: " + sem);
```

for (int 1=0; icn; i++) {

String usy s-nextline ();

String name: s-nextline(); System out - brint (" Enter Sem : ");

int sem = 8. next9n+(); THE COTTO I REMAIND AND THE CONTEST

System-out. first ("Enter USN: ");

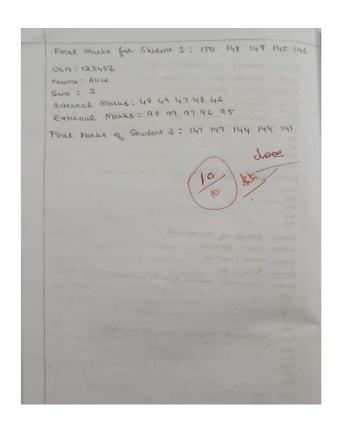
System out print ("Enter name :");

System our frinth ("Enter addits for Student +(i+)

; (":"+

System out , println ("Enter intomal marks for 5

```
External, jova package SEE;
                                                                             CIE. Internals;
                                                                     public class Enternal extends goternals &
                                                                    public int () caternal mostes of new Int (5);
                                                                     bublic External (sking usn, sking name, int sum, int C)
                                                                     externalmates) &
                                                                     Super (uen, name, sem);
                                                                     this external masks = external masks;
                                                                    bublic wid display () {
                                                                    System out paint ("Entranal Makes: ");
                                                                    for (101)=0; jes; j++) &
                                                                    Total masks []] = 15. next Ant ();
Sycken out frontly (mark + " ");
                                                                    internal Students [: ] = new Internals ( wen, name, sem,
                                                                    int[] external Marks = now int (5):
                                                                     System out printin ("Enter external marks for 5
impost CIE. Internals;
impost SEE . External;
                                                                     tox (nt j=0; Jes; J++) {
impost java. util Scanner;
                                                                     external masks [j] = s. next 9nt();
public class FinalMarks &
public storic void main (String angs []) {
                                                                     enternal Students [i]=new External (usn, name, sem
Scanner &= new Scanner (System-in);
System-out-baint ("Enter number of students:");
                                                                    8-next Linell;
                                                                    for (int i=0; icn; i++) {
int n = s-next ant ();
                                                                     internal students [i] edisplay ();
s-restline();
Internals [] internal Students - new Internals [n];
                                                                     external Students [i] · distray();
External [] external Students = now External [n];
                                                                    int [] that Masks = new int [5]:
```



```
Student.java
package CIE;
public class Student {
  public String usn;
  public String name;
  public int sem;
  public Student(String usn, String name, int sem) {
     this.usn = usn;
     this.name = name;
     this.sem = sem;
  }
  public void display() {
     System.out.println("USN: " + usn);
     System.out.println("Name: " + name);
     System.out.println("Semester: " + sem);
  }
}
Internals.java
package CIE;
public class Internals extends Student {
  public int[] internalMarks = new int[5];
```

```
public Internals(String usn, String name, int sem, int[] internalMarks) {
     super(usn, name, sem);
     this.internalMarks = internalMarks;
  }
  public void display() {
     super.display();
     System.out.print("Internal Marks: ");
     for (int mark : internalMarks) {
       System.out.print(mark + " ");
     System.out.println();
  }
}
External.java
package SEE;
import CIE.Student;
public class External extends Student {
  public int[] externalMarks = new int[5];
  public External(String usn, String name, int sem, int[] externalMarks) {
     super(usn, name, sem);
     this.externalMarks = externalMarks;
  }
   @Override
  public void display() {
     super.display();
     System.out.print("External Marks: ");
     for (int mark : externalMarks) {
       System.out.print(mark + " ");
     System.out.println();
  }
}
FinalMarks.java
import CIE.Internals;
import SEE.External;
import java.util.Scanner;
public class FinalMarks {
  public static void main(String[] args) {
     Scanner scanner = new Scanner(System.in);
     System.out.print("Enter number of students: ");
     int n = scanner.nextInt();
```

```
scanner.nextLine(); // Consume newline
     Internals[] internalStudents = new Internals[n];
     External[] externalStudents = new External[n];
     for (int i = 0; i < n; i++) {
       System.out.println("Enter details for student " + (i + 1) + ":");
       System.out.print("Enter USN: ");
       String usn = scanner.nextLine();
       System.out.print("Enter name: ");
       String name = scanner.nextLine();
       System.out.print("Enter semester: ");
       int sem = scanner.nextInt();
       int[] internalMarks = new int[5];
       System.out.println("Enter internal marks for 5 courses: ");
       for (int j = 0; j < 5; j++) {
          internalMarks[j] = scanner.nextInt();
       }
       internalStudents[i] = new Internals(usn, name, sem, internalMarks);
       int[] externalMarks = new int[5];
       System.out.println("Enter external marks for 5 courses: ");
       for (int j = 0; j < 5; j++) {
          externalMarks[j] = scanner.nextInt();
       }
       externalStudents[i] = new External(usn, name, sem, externalMarks);
       scanner.nextLine(); // Consume newline
     }
     // Display student details and final marks
     for (int i = 0; i < n; i++) {
       internalStudents[i].display();
       externalStudents[i].display();
       int[] finalMarks = new int[5];
       for (int j = 0; j < 5; j++) {
          finalMarks[j] = internalStudents[i].internalMarks[j] +
externalStudents[i].externalMarks[j];
       System.out.print("Final Marks for student " + (i + 1) + ": ");
       for (int mark : finalMarks) {
          System.out.print(mark + " ");
       }
```

```
System.out.println();
}
scanner.close();
}
```

```
Enter number of students: 2
Enter details for student 1:
Enter USN: 1MS20CS001
Enter name: Alice
Enter semester: 5
Enter internal marks for 5 courses:
20
25
22
18
24
Enter external marks for 5 courses:
60
55
58
62
57
Enter details for student 2:
Enter USN: 1MS20CS002
Enter name: Bob
Enter semester: 5
Enter internal marks for 5 courses:
21
23
20
19
25
Enter external marks for 5 courses:
61
```

Enter external marks for 5 courses:

61

56

59

63

58

USN: 1MS20CS001

Name: Alice Semester: 5

Internal Marks: 20 25 22 18 24 External Marks: 60 55 58 62 57

Final Marks for student 1: 80 80 80 80 81

USN: 1MS20CS002

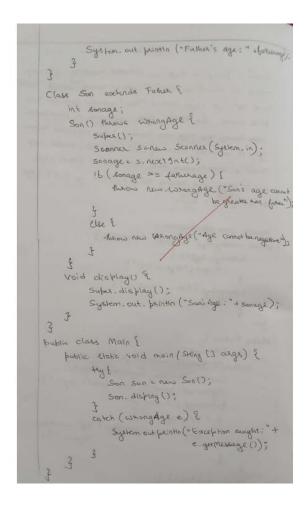
Name: Bob Semester: 5

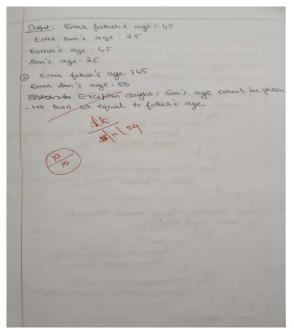
Internal Marks: 21 23 20 19 25 External Marks: 61 56 59 63 58

Final Marks for student 2: 82 79 79 82 83

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.

HEEK T Write a program that demonstrates handling of exceptions in inhesitance tree. Exect a base called "Faths and derived class called "son" which eatends the base class. In Father class, implement a constructor which takes the age and thorows the exception when -gage() when the input age <0. In Son, class, imp -tomesta constructor that cases born father and son's age and twoos on exception it son's age is >= famer's age. impost java. Util. Scanner; class wrong Age extends Exception E Whong - Age () {
Super ("Age Enor"); Wang Age (String message) { super (message); class Futher & int fatherage; Father() thoows whongAge { Scanner S=new Scanner (System.in); System. out. besintly ("Enter Garner's age: "); fatherage = s. nest 9 nt(1; ib (fatherage <0) { those her enoughge ("Age connorbe void distray () {



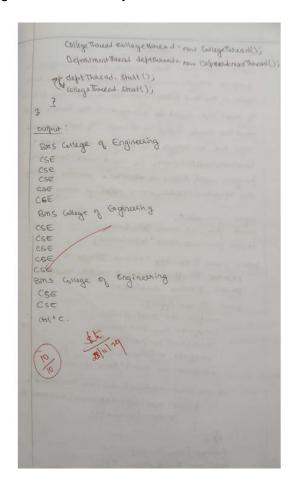


```
import java.util.Scanner;
class WrongAge extends Exception {
  WrongAge() {
     super("Age Error");
  }
  WrongAge(String message) {
     super(message);
  }
}
class Father {
  int fatherAge;
  Father() throws WrongAge {
     Scanner s = new Scanner(System.in);
     System.out.print("Enter Father's Age: ");
     fatherAge = s.nextInt();
     if (fatherAge < 0) {
       throw new WrongAge("Age cannot be negative");
     }
  }
  void display() {
     System.out.println("Father's Age: " + fatherAge);
  }
}
class Son extends Father {
  int sonAge;
  Son() throws WrongAge {
     super(); // Calling the constructor of Father class
     Scanner s = new Scanner(System.in);
     System.out.print("Enter Son's Age: ");
     sonAge = s.nextInt();
     if (sonAge >= fatherAge) {
       throw new WrongAge("Son's age cannot be greater than or equal to father's age");
     } else if (sonAge < 0) {
       throw new WrongAge("Age cannot be negative");
     }
  }
  void display() {
     super.display(); // Display father's age
     System.out.println("Son's Age: " + sonAge);
```

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4391]
(c) Microsoft Corporation. All rights reserved.
C:\Users\KLB-ES>cd ..
C:\Users>cd..
C:\>d:
D:\>cd MONISHA2024
D:\MONISHA2024>javac Main.java
D:\MONISHA2024>java Main
Enter Father's Age: 45
Enter Son's Age: 25
Father's Age: 45
Son's Age: 25
D:\MONISHA2024>java Main
Enter Father's Age: 45
Enter Son's Age: 50
Exception caught: Son's age cannot be greater than or equal to father's age
D:\MONISHA2024>
```

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

```
WEEK-8
White 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 College Thread extends Thread &
    & ( ) nucl bios silded
        try ? while (time) {
                 System out beinth ("BMS college of Engineerie
                 Therad. Sleet (10000);
         Catch (Intersupred Exception e) &
            System. out. paintin ("Therand in tesseup ted");
class Department Thread extends Thread {
    L'Onux brow sidual
         try {
                   System. out. frintin ("CSE");
                    Thread. selep (2000);
         Carch (Interrupted Exception e) &
               System out - beintin ( "Thread Intersupted");
bublic class Maint &
   public state void main (String augests)
```



```
class CollegeThread extends Thread {
  public void run() {
     try {
       while (true) {
          System.out.println("BMS College of Engineering");
          Thread.sleep(10000); // Sleep for 10 seconds
       }
     } catch (InterruptedException e) {
       System.out.println("Thread interrupted");
     }
  }
}
class DepartmentThread extends Thread {
  public void run() {
     try {
       while (true) {
          System.out.println("CSE");
          Thread.sleep(2000); // Sleep for 2 seconds
       }
     }
```

```
D:\MONISHA2024>java Main1
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
```

#### **PROGRAM 9**

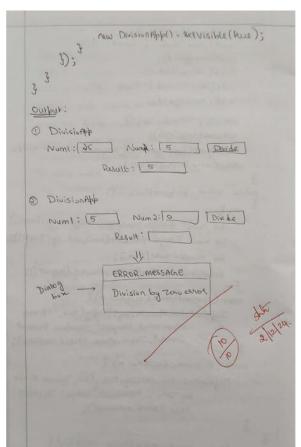
Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

```
WEEK-9
white a program that creates a user introduce to
beeform integer divisions. The user enters two num
bees in the text fields, Numl and Numa. The division
of Nums and Numa is displayed in the Result field
 when the divide button is dicked. If wom! a num?
were not an integer, the program would there a
NumberFormatException. It was were zero, the
program would those an Asithmetic Exception Display
 The exception meetings in a meetings dialog box.
 impose java.aut. ";
impose java aut event. Achientrent; impose java aut event. Achientesnes;
public class DivisionApp extends JF. Some ?
    pairate Transferd numbriel, numbriel, resourfield;
     purolic Division Appl ) &

number - new structed (10);

number - new struct Field (10);
            result feld : 1000 These Field (10)
           Presultigialid. set Editable ( Gales);
            diversione in same ( " Bride );
          divide Button. addActing tistace (new Actualisation )
               tublic void action Performed (Action Evente) 2 beatenm Division ();
         Setlagant (new Flowdayput (1));
add (new Tlabel ("Num: "));
```

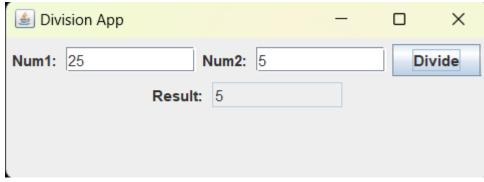
```
add (new Trabel ("Numd:"));
     add (numafield);
     add (divide Button);
     add (new Trabel ("Result: "));
     add (Presultfield);
    set Title ( " Division App ");
     setSize (400, 150);
    set Defaut Close operation (JERAME - EXIT_ON_CLOSE);
    Set Location Relative To (null);
public void perform Division ()
        int num! = Integer - panse Int (num! field-getText);
        int numb - ortegen. proceedors (numbbeld-getTexto);
        int result = num / num 2;
        resultfield. SetText (String value of (result));
    earch (Number Format Exception e) {
        Joshion Pane. Show Mexenge Dialog ( this, " Decese
                 enter valid integers", "Number Format
Grand", JophianPane. BRROR_MESSAGE);
  catch ( Anithmetic Exception e) {
       Johnsone. Show Message Dialog (Mis, "Cannot divide
by Zeno", "Asithretic esonos", Johnson
                 Pane EMPOR MESSAGE);
public Shakic void main (sking aggs (1)) {
Swing Diffities. invoke Later (new Rumable () }
            3 (new biov sildery
```

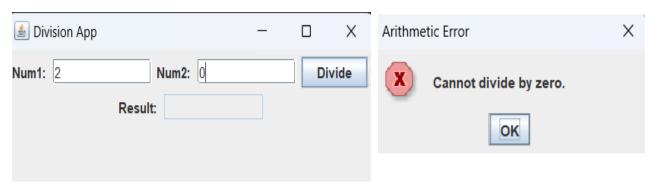


### **SOURCE CODE:**

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class DivisionApp extends JFrame {
  private JTextField num1Field, num2Field, resultField;
  private JButton divideButton;
  public DivisionApp() {
    // Create UI elements
     num1Field = new JTextField(10);
     num2Field = new JTextField(10);
     resultField = new JTextField(10);
     resultField.setEditable(false); // Make result field non-editable
     divideButton = new JButton("Divide");
    // Add action listener to the button
     divideButton.addActionListener(new ActionListener() {
       @Override
       public void actionPerformed(ActionEvent e) {
          performDivision();
       }
    });
    // Set up the layout
     setLayout(new FlowLayout());
     add(new JLabel("Num1: "));
     add(num1Field);
     add(new JLabel("Num2: "));
     add(num2Field);
     add(divideButton);
     add(new JLabel("Result: "));
     add(resultField);
    // Set up the frame
     setTitle("Division App");
     setSize(400, 150);
     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
     setLocationRelativeTo(null); // Center the frame
  }
  private void performDivision() {
    try {
       // Get the input values
       int num1 = Integer.parseInt(num1Field.getText());
       int num2 = Integer.parseInt(num2Field.getText());
```

```
// Perform the division
       int result = num1 / num2;
       // Display the result
       resultField.setText(String.valueOf(result));
    } catch (NumberFormatException e) {
       JOptionPane.showMessageDialog(this, "Please enter valid integers.", "Number Format
Error", JOptionPane.ERROR MESSAGE);
     } catch (ArithmeticException e) {
       JOptionPane.showMessageDialog(this, "Cannot divide by zero.", "Arithmetic Error",
JOptionPane.ERROR_MESSAGE);
  }
  public static void main(String[] args) {
     SwingUtilities.invokeLater(new Runnable() {
       @Override
       public void run() {
         new DivisionApp().setVisible(true);
       }
    });
  }
}
OUTPUTS:
                                                         Division App
                             Num2: 5
 Num1: 25
                                                            Divide
```





### **PROGRAM 10**

Demonstrate Inter process Communication and deadlock

## INTER PROCESS COMMUNICATION

```
WEEK-10
Demeronskate Inter Process Communication and Dea
 -dlock
SPC: Implementation of a Paducea and Consumer
  int ni
  boolean valueSet= false;
  synchronized int get() 2
      while (!valueSel)
       tey & System.out. println ("In Consumer Waiting In");
         wart();
      catch (Interrupted Exception e) ?
         System out printin ("Intertupled Exception cought");
     System out paintin ("Got: "+0);
     valueSet = false;
     Systemout parotto (" In Intimate Paroduces In");
      CO Mitton
     Suhon n;
 Synchronized void put (int n) {
      while (value Set)
         System out paintin ("In Produces waiting In");
     Catch (Intersupted Exception c) {
```

```
System out brinkin ( "Interrupted Exception congress
        this.n=n;
        value Set = Kue;
        System out printin ("Put: " +n);
        System out println ("In Intimate Consumer In");
        notity();
class Phoduces implements Runnable &
    Q 9;
     Panduces (Qq) {
         this q=q;
          new Thread (this, " Produces"). Stort();
     public void aunil &
         int i=o;
         while (icis) &
         9-pur (i++);
 3
class consumer implements Rumable &
    993
     consumer (a 9) &
        this 9 = 9 ;
        new Theread (this, "Consumer") (Start !);
     public void sun() {
int i=0;
while (icis) {
            int n=q-gett);
              System. out. println ("consumed: "+n);
```

```
14+3
dass PCFixed &
   & (ITages public Static void main (string cages) &
         Q q = new Q(1);
          new Producer(9);
          new Consumer (2);
          System out printhn ("Pares Control-( to Stop."))
output: Peress Control-C to Stop
                         Panduces waiting
Put: 0
guinate Consumer
                           Gnot: 3
                       Inhmate Produces consumed: 3
Produces waiting
gatimate Paroduces
                          Put: 4
Put: 1
90 Antimate Consumer
Producer working
Consumed: 0
90 Antimate Consumer
Producer working
Goot: 4
90 Antimate Producer
                          consumed : 4
Crot:1
                          Pux:5
Intimate Parduces Intimate Consumer
consumed:1
                       Producer walking
Grat : 5
Intimate Consumer grimate Parducel
                      Consumed: 5
 Producestaiting
6/01/2
                          Put : 6
                         Intimate Consumer
Intimate Produces
                         Bonduces worthing
Consumed: 2
                         Grot: 6
Antimate Phoduces
Put:3
```

```
SOURCE CODE:
class Q {
int n;
boolean valueSet = false;
synchronized int get() {
while(!valueSet)
try {
System.out.println("\nConsumer waiting\n");
wait();
} catch(InterruptedException e) {
System.out.println("InterruptedException caught");
System.out.println("Got: " + n);
valueSet = false;
System.out.println("\nIntimate Producer\n");
notify();
return n;
synchronized void put(int n) {
while(valueSet)
try {
System.out.println("\nProducer waiting\n");
wait();
} catch(InterruptedException e) {
System.out.println("InterruptedException caught");
}
this.n = n;
```

```
valueSet = true;
System.out.println("Put:"+n);\\
System.out.println("\nIntimate Consumer\n");
notify();
}
class Producer implements Runnable {
Q q;
Producer(Q q) {
this.q = q;
new Thread(this, "Producer").start();
}
public void run() {
int i = 0;
while(i<15) {
q.put(i++);
}
}
class Consumer implements Runnable {
Qq;
Consumer(Q q) {
this.q = q;
new Thread(this, "Consumer").start();
}
public void run() {
int i=0;
```

```
while(i<15) {
int r=q.get();
System.out.println("consumed:"+r);
i++;
}
}
class PCFixed {
public static void main(String args[]) {
Q q = new Q();
new Producer(q);
new Consumer(q);
System.out.println("Press Control-C to stop.");
}
</pre>
```

# **OUTPUTS:**

```
D:\MONISHA2024>java PCFixed
Press Control-C to stop.
Put: 0
Intimate Consumer
Producer waiting
Got: 0
Intimate Producer
Put: 1
Intimate Consumer
Producer waiting
consumed:0
Got: 1
Intimate Producer
consumed:1
Put: 2
Intimate Consumer
```

```
Intimate Consumer

Producer waiting

Got: 2

Intimate Producer

consumed:2
Put: 3

Intimate Consumer

Producer waiting

Got: 3

Intimate Producer

consumed:3
Put: 4

Intimate Consumer

Producer waiting
```

Got: 4

Intimate Producer

consumed:4

Put: 5

Intimate Consumer

**Producer** waiting

Got: 5

Intimate Producer

consumed:5

Put: 6

Intimate Consumer

**Producer** waiting

Got: 6

Intimate Producer

consumed:6

### **DEADLOCK**

```
DEADLOCK
class A &
 synchronized void too (B 6) {
      String name = Thread - current Thread () , get Nome ();
       System out . brintln (name + " enteredA . too");
       try { Thread. Slep(1000);
      catch (Exception e) &
           System out paintln ("A Interrupted");
      System out beintln (name + 'trying to call B. last ()')
     b-last();
  3
 void last () }
      System out beintly ("Inside A. (ast");
class B &
   Synchronized void box (A a) {
       String name = Thread. current Thread(). getName();
        Systemour println (name + " entered B. box");
        try & Thread. sleep (1000);
        catch (Exception e) {
             System out paintin (" B Intersuppled ");
        System-out-paintln (nome + " typing to call A-last()),
      a-last ();
   void last 17 %
        System out freintly ( " Anside A. last")
```

```
class Deadlock implements Runnable &
     A a = new A();
    B b = nu B();
    Deablock!) &
        Thread. current Thread(). Set Name ("Main Thread"):
        Thread to new Thread (this, "Racing Threads);
        t- start();
         a - too (b);
         System out brinth ("Back in main thread");
     Public void sun() ?
         6. bus (a);
          System out paintln ("Book in other thread");
    public Static void main (String ang []) {
         new Deadlock ();
confut: Racing Thread entered B. book
MainThread entered A-foo
Mointhread trying to call B-last 1)
Racing thread trying to call A-last()
Irvide A. last
Inside Alast
Back in owner thread Back in main thread
```

# SOURCE CODE:

```
class A {
synchronized void foo(B b) {
String name =
Thread.currentThread().getName();
System.out.println(name + " enteredA.foo");
try {
Thread.sleep(1000);
} catch(Exception e) {
System.out.println("A Interrupted");
System.out.println(name + " trying to call B.last()");
b.last();
}
void last() {
System.out.println("Inside A.last");
}
class B {
synchronized void bar(A a) {
String name = Thread.currentThread().getName();
System.out.println(name + " entered B.bar");
try {
Thread.sleep(1000);
} catch(Exception e) {
System.out.println("B Interrupted");
}
```

```
System.out.println(name + " trying to call A.last()");
a.last();
void last() {
System.out.println("Inside A.last");
}
}
class Deadlock implements Runnable
A a = new A();
Bb = new B();
Deadlock() {
Thread.current Thread (). set Name ("Main Thread");\\
Thread t = new Thread(this, "RacingThread");
t.start();
a.foo(b); // get lock on a in this thread.
System.out.println("Back in main thread");
public void run() {
b.bar(a); // get lock on b in other thread.
System.out.println("Back in other thread");
}
public static void main(String args[]) {
new Deadlock();
}
}
```

## **OUTPUTS:**

D:\MONISHA2024>javac Deadlock.java

D:\MONISHA2024>java Deadlock
RacingThread entered B.bar
MainThread enteredA.foo
MainThread trying to call B.last()
RacingThread trying to call A.last()
Inside A.last
Inside A.last
Back in other thread
Back in main thread