

# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



## LAB REPORT on **Object Oriented Java Programming** **(23CS3PCOOJ)**

*Submitted by*

**Deepthi M (1BM23CS088)**

*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 **Deepthi M (1BM23CS088)**, who is bonafide student of **B.M.S. College of Engineering**. It is in partial fulfilment 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.

Lab faculty Incharge Name Assistant Professor Department of CSE, BMSCE	Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE
--	---

# Index

<b>Sl. No.</b>	<b>Date</b>	<b>Experiment Title</b>	<b>Page No.</b>
1	01.10.2024	Implementing Quadratic Equation	7-10
2	08-10-2024	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.	11-15
3	15.10.24	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 <code>toString()</code> method that could display the complete details of the book.  Develop a Java program to create n book objects	16-21
4	22.10.2024	Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named <code>printArea()</code> . 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 <code>printArea()</code> that prints the area of the given shape.	22-25

5	<p>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.</p> <p>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:</p> <ul style="list-style-type: none"> <li>a) Accept deposit from customer and update the balance.</li> <li>b) Display the balance.</li> <li>c) Compute and deposit interest</li> <li>d) Permit withdrawal and update the balance</li> </ul> <p>Check for the minimum balance, impose penalty if necessary and update the balance.</p>	26-35
6	<p>12.11.2024</p> <p>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.</p>	35-40

7	19.11.2024	We have created an interface named Polygon. It includes a default method getPerimeter() and an abstract method getArea (). We can calculate the perimeter of all polygons in the same manner so we implemented the body of get Perimeter() in Polygon. Now, all polygons that implement Polygon can use getPerimeter () to calculate perimeter. However, the rule for calculating the area is different for different polygons. Hence, getArea() is included without implementation. Any class that implements Polygon must provide an implementation of getArea ()	41-43
---	------------	---	-------

8	26.11.2024	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 uses both father and son's age and throws an exception if son's age is >=father's age.	44-46
9	03.12.2024	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	47-49

10.	03.12.2024	Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box	50-53
-----	------------	---	-------

GitHub Link : [https://github.com/Deepthi-1534/JAVA\\_LAB\\_PROGRAMS.git](https://github.com/Deepthi-1534/JAVA_LAB_PROGRAMS.git)

## PROGRAM 1 :

### CODE:

```
import java.util.Scanner;

class Quadratic {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the coefficient a:");
        double a = sc.nextDouble();

        System.out.println("Enter the coefficient b:");
        double b = sc.nextDouble();

        System.out.println("Enter the coefficient c:");
        double c = sc.nextDouble();

        double d = b * b - 4 * a * c;

        if (d > 0)
        {
            double root1 = (-b + Math.sqrt(d)) / (2 * a);
            double root2 = (-b - Math.sqrt(d)) / (2 * a);
            System.out.println("The roots are real and distinct:");
            System.out.println("Root 1: " + root1);
            System.out.println("Root 2: " + root2);
        }
    }
}
```

```
else if (d == 0)
{
    double root = -b / (2 * a);
    System.out.println("The roots are real and equal:");
    System.out.println("Root: " + root);
}

else
{
    double realPart = -b / (2 * a);
    double imaginaryPart = Math.sqrt(-d) / (2 * a);
    System.out.println("The roots are complex and imaginary:");
    System.out.println("Root 1: " + realPart + " + " + imaginaryPart + "i");
    System.out.println("Root 2: " + realPart + " - " + imaginaryPart + "i");
}

System.out.println("Name : Deepthi M ");
System.out.println("USN : 1BM23CS088");

sc.close();
}
```

}

}

OUTPUT:

```
F:\JAVA LAB\Quadratic>javac Quadratic.java
```

```
F:\JAVA LAB\Quadratic>java Quadratic
```

```
Enter the coefficient a:
```

```
1
```

```
Enter the coefficient b:
```

```
3
```

```
Enter the coefficient c:
```

```
3
```

```
The roots are complex and imaginary:
```

```
Root 1: -1.5 + 0.8660254037844386i
```

```
Root 2: -1.5 - 0.8660254037844386i
```

```
Name : Deepthi M
```

```
USN : 1BM23CS088
```

```
F:\JAVA LAB\Quadratic>
```

```
F:\JAVA LAB\Quadratic>javac Quadratic.java
```

```
F:\JAVA LAB\Quadratic>java Quadratic
```

```
Enter the coefficient a:
```

```
2
```

```
Enter the coefficient b:
```

```
6
```

```
Enter the coefficient c:
```

```
1
```

```
The roots are real and distinct:
```

```
Root 1: -0.17712434446770464
```

```
Root 2: -2.8228756555322954
```

```
Name : Deepthi M
```

```
USN : 1BM23CS088
```

```
F:\JAVA LAB\Quadratic>javac Quadratic.java
```

```
F:\JAVA LAB\Quadratic>java Quadratic
```

```
Enter the coefficient a:
```

```
1
```

```
Enter the coefficient b:
```

```
2
```

```
Enter the coefficient c:
```

```
1
```

```
The roots are real and equal:
```

```
Root: -1.0
```

```
Name : Deepthi M
```

```
USN : 1BM23CS088
```

## OBSERVATION :

else  
    {  
        double realpart = -b / (2\*a);  
        double imagpart = Math.sqrt(-d) / (2\*a);  
        System.out.println("Roots are complex & different");  
        System.out.println("Root 1 : " + realpart + " + " + imagpart + "i");  
        System.out.println("Root 2 : " + realpart - imagpart + "i");  
    }  
}  
sc.close();  
}  
}  
  
Enters coefficient of a:  
2  
Enters coefficient of b:  
5  
Enters coefficient of c:  
6  
Roots are complex & different  
Root 1 : -0.11 + 0.50i  
Root 2 : -0.11 - 0.50i

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

```
(3) import java.util.Scanner;  
public class Q3  
{  
    public static void main(String args[])  
    {  
        Scanner s = new Scanner(System.in);  
        System.out.print("Coefficient of a : ");  
        double a = nextDouble();  
        System.out.print("Coefficient of b : ");  
        double b = nextDouble();  
        System.out.print("Coefficient of c : ");  
        double c = nextDouble();  
        double d = (b*b) - 4*a*c;  
        if(d>0)  
        {  
            double m = (-b + Math.sqrt(d)) / (2*a);  
            double r1 = (-b - Math.sqrt(d)) / (2*a);  
            System.out.println("Root 1 : " + r1);  
            System.out.println("Root 2 : " + m);  
        }  
        else if(d==0)  
        {  
            double root = -b / (2*a);  
            System.out.println("Roots are real & same");  
            System.out.println("Root : " + root);  
        }  
    }  
}
```

## PROGRAM 2 :

### CODE :

```
import java.util.Scanner;

class Students {
    String usn;
    String name;
    int marks[];
    int credits[];
    int subjects;

    Students(int subjects) {
        this.subjects = subjects;
        credits = new int[subjects];
        marks = new int[subjects];
    }

    void input() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the student's USN: ");
        usn = sc.nextLine();
        System.out.println("Enter the student's name: ");
        name = sc.nextLine();

        for (int i = 0; i < subjects; i++) {
            System.out.println("Enter the credits of subject : ");
        }
    }
}
```

```
    credits[i] = sc.nextInt();

    System.out.println("Enter the marks of subject : ");

    marks[i] = sc.nextInt();

}

}

double SGPA_calc() {

    int totalCredits = 0;

    double weightedPoints = 0.0;

    for (int i = 0; i < subjects; i++) {

        totalCredits += credits[i];

        double gradePoint = (marks[i] / 10.0);

        weightedPoints += gradePoint * credits[i];

    }

    return weightedPoints / totalCredits;

}

void display() {

    System.out.println("Student Name: " + name);

    System.out.println("Student USN: " + usn);

    for (int i = 0; i < subjects; i++) {

        System.out.println( "Credits: " + credits[i] + "and Marks: " + marks[i]);

    }

    System.out.printf("SGPA: %.2f%n", SGPA_calc());

}
```

```
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the number of subjects: ");
    int subjects = sc.nextInt();

    Students student1 = new Students(subjects);
    Students student2 = new Students(subjects);

    System.out.println("Enter details for Student 1:");
    student1.input();
    System.out.println("Details of Student 1:");
    student1.display();

    System.out.println("\nEnter details for Student 2:");
    student2.input();
    System.out.println("Details of Student 2:");
    student2.display();

    System.out.println("Name : Deepthi M");
    System.out.println("USN : 1BM23CS088");

    sc.close();
}

}
```

## OUTPUT :

```
55
Enter the credits of subject :
4
Enter the marks of subject :
67
Enter the credits of subject :
3
Enter the marks of subject :
88
Details of Student 1:
Student Name: asd
Student USN: 23de
Credits: 2and Marks: 55
Credits: 4and Marks: 67
Credits: 3and Marks: 88
SGPA: 7.13

Enter details for Student 2:
Enter the student's USN:
33fr
Enter the student's name:
def
Enter the credits of subject :
3
Enter the marks of subject :
56
Enter the credits of subject :
4
Enter the marks of subject :
77
Enter the credits of subject :
2
Enter the marks of subject :
89
Details of Student 2:
Student Name: def
Student USN: 33fr
Credits: 3and Marks: 56
Credits: 4and Marks: 77
Credits: 2and Marks: 89
SGPA: 7.27
Name : Deepthi M
USN : 1BM23CS088
```

# OBSERVATION :

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

```

2. import java.util.Scanner;
public class Students {
    String USN;
    String name;
    int marks[];
    int credits[];
    int total;
    int subjects;
    students (int subjects) {
        this.subjects = subjects;
        credits = new int [subjects];
        marks = new int [marks];
        for (int i=0; i < subjects; i++) {
            credits[i] = 2;
            marks[i] = 20;
        }
    }
    void input() {
        Scanner s = new Scanner (System.in);
        System.out.println ("Enter the USN of the student : ");
        USN = s.nextLine();
        System.out.println ("Enter the marks of the student : ");
        name = s.nextLine();
    }
}

```

classmate (All rights reserved)
 21/01/2023 21:00

```

System.out.println ("Enter the no of subjects : ");
for (int i=0; i < subjects; i++) {
    System.out.println ("Enter the no of subjects : ");
    subjects = s.nextInt();
    for (int i=0; i < subjects; i++) {
        System.out.println ("Enter the credits for subjects : ");
        credits[i] = s.nextInt();
        System.out.println ("Enter the marks : ");
        marks[i] = s.nextInt();
    }
}
void display () {
    System.out.println ("Student's name : " + name);
    System.out.println ("Student's USN : " + USN);
    System.out.println ("The credits and marks are as follows : ");
    for (int i=0; i < subjects; i++) {
        System.out.println ("Credits : " + credits[i] + " and Marks : " +
                           marks[i]);
    }
    System.out.println ("SGPA of the student : " + sgpa_calc());
}

```

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

```

void sgpa_calc () {
    int totalcredits = 0;
    int points = 0;
    for (int i=0; i < subjects; i++) {
        totalcredits += credits[i];
        points += (marks[i] * 10) / totalcredits;
    }
    return (points / totalcredits);
}

public static void main (String args[]) {
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter the no of subjects : ");
    subjects = sc.nextInt();
    students a = new Students (System.in);
    students b = new Students (System.in);
    a.input();
    a.display();
    b.input();
    b.display();
}

```

*(Handwritten note: Exercised)*

(1) 2100 - 09/01/2023

Enter the number of subjects  
3  
Enter the student name : asd  
Enter the student USN : 23dl  
23dl  
Enter the credits : 2  
2  
Enter the marks : 55  
55  
Enter the credits : 4  
4  
Enter the marks : 67  
67  
Enter the credits : 3  
3  
Enter the marks : 88  
88  
student's name : asd  
student's USN : 23dl  
credits : 2 and marks : 55  
credits : 4 and marks : 67  
credits : 3 and marks : 88  
SGPA : 7.13

## PROGRAM 3 :

CODE:

```
import java.util.Scanner;
```

```
class Books {
```

```
    private String name;
```

```
    private String author;
```

```
    private int price;
```

```
    private int numPages;
```

```
Books(String name, String author, int price, int numPages) {
```

```
    this.name = name;
```

```
    this.author = author;
```

```
    this.price = price;
```

```
    this.numPages = numPages;
```

```
}
```

```
public String getName() {
```

```
    return name;
```

```
}
```

```
public void setName(String name) {
```

```
    this.name = name;
```

```
}
```

```
public String getAuthor() {
```

```
    return author;
```

```
}
```

```
public void setAuthor(String author) {
```

```
    this.author = author;
```

```
}
```

```
public int getPrice() {
```

```
    return price;
```

```
}
```

```
public void setPrice(int price) {
```

```
    this.price = price;
```

```
}
```

```
public int getNumPages() {
```

```
    return numPages;
```

```
}
```

```
public void setNumPages(int numPages) {
```

```
    this.numPages = numPages;
```

```
}
```

```
@Override
```

```
public String toString() {
```

```
    return "Book [Name: " + name + ", Author: " + author + ", Price: " + price + ",  
    Pages: " + numPages + "]";
```

```
 }  
 }
```

```
class Main {  
  
    public static void main(String[] args) {  
  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("Enter the number of books:");  
        int n = scanner.nextInt();  
        scanner.nextLine();  
  
        Books[] books = new Books[n];  
  
        for (int i = 0; i < n; i++) {  
            System.out.println("Enter book's name:");  
            String name = scanner.nextLine();  
  
            System.out.println("Enter author's name: ");  
            String author = scanner.nextLine();  
  
            System.out.println("Enter the price: ");  
            int price = scanner.nextInt();  
  
            System.out.println("Enter the pages :");  
            int numPages = scanner.nextInt();  
            scanner.nextLine();
```

```

books[i] = new Books(name, author, price, numPages);

}

System.out.println("\nDetails of the books:");

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

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

}

System.out.println("\nNAME: Deepthi M");

System.out.println("USN: 1BM23CS088");

scanner.close();

}

}

```

## OUTPUT :

```

F:\JAVA LAB\Books>javac Main.java

F:\JAVA LAB\Books>java Main
Enter the number of books:
2
Enter book's name:
ABC
Enter author's name:
AAA
Enter the price:
345
Enter the pages :
200
Enter book's name:
DEF
Enter author's name:
BBB
Enter the price:
765
Enter the pages :
300

Details of the books:
Book [Name: ABC, Author: AAA, Price: 345, Pages: 200]
Book [Name: DEF, Author: BBB, Price: 765, Pages: 300]

NAME: Deepthi M
USN: 1BM23CS088

```

# OBSERVATION :

```

import java.util.Scanner; // Containing package statement
public class Book { // Containing class declaration
    private String name; // Containing attribute
    private String author; // Containing attribute
    private double price; // Containing attribute
    private int pages; // Containing attribute

    public Book (String name, String author, double price, int pages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.pages = pages;
    }

    public String getName() {
        return name;
    }

    public void setName (String name) {
        this.name = name;
    }

    public double getPrice() {
        return price;
    }

    public void setPrice (double price) {
        this.price = price;
    }

    public int getPages() {
        return pages;
    }

    public void setPages (int pages) {
        this.pages = pages;
    }
}

```

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

```

public String getAuthor() {
    return author;
}

public void setAuthor (String author) {
    this.author = author;
}

public double getPrice () {
    return price;
}

public void setPrice (double price) {
    this.price = price;
}

public int getPages () {
    return pages;
}

public void setPages (int pages) {
    this.pages = pages;
}

```

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

```

public String toString () {
    return "Book [Name : " + name + ", Author : " + author + ","
           + " Price : " + price + ", Pages : " + pages + "]";
}

public class Book {
}

public static void main (String args []) {
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter the no. of books : ");
    int n = sc.nextInt();
    sc.nextLine();

    Book [] books = new Book [n];
    for (int i = 0; i < n; i++) {
        System.out.println ("Enter the book details : ");
        sc.nextLine();
        System.out.println ("Enter book's name : ");
        String name = sc.nextLine();
        System.out.println ("Enter author's name : ");
        String author = sc.nextLine();
        System.out.println ("Enter price : ");

```

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

```

        double price = sc.nextDouble();
        System.out.println ("Enter pages : ");
        int pages = sc.nextInt();
        sc.nextLine();
        books[i] = new Book (name, author, price, pages);
    }

    System.out.println ("Book details : ");
    for (int i = 0; i < n; i++) {
        System.out.println (books[i]);
    }
    sc.close();
}

Enter the no. of books: 2
Enter name:
ABC
Enter author:
PAA
Enter price:
345
Enter no. pages
200

```

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

Entes marini: marina officia domini bimbi e soci a squalo.

def *get\_ip*(*example*):  
 *ip* = *example*.split(' ')  
 *ip* = *ip*[0]

Enter author's name: JDS with lowercase & extra letters

BBB is primarily about a state's financial health

ents her pris: • *causativeal* 3'11.2002 until

765

Enter thru pages

Laminae 2200-1100m

300

~~Boat details:~~

(130) how good

Book Name: ABC, authors: PAA, Price: 345, Pages: 200

Book [Name: DEF, author, BBB, price, 765, pages: 300]

~~0/8 seen~~

~~86  
15/10/24~~

10° dip in to 201312 axis "J" attitude two minutes

2012-13: "The first year of the new decade will be a year of great challenges and opportunities for our country."

Chilean Free Expresses Guiney

7 (1-10<sup>3</sup>) biov

*Spelunking, bouldering, DSD 2012-08-25, Pseudoceratopis*

## PROGRAM 4 :

CODE:

```
abstract class Shape {
```

```
    int x1;
```

```
    int x2;
```

```
    Shape(int x1, int x2) {
```

```
        this.x1 = x1;
```

```
        this.x2 = x2;
```

```
}
```

```
    abstract void area();
```

```
}
```

```
class Rectangle extends Shape {
```

```
    Rectangle(int length, int breadth) {
```

```
        super(length, breadth);
```

```
}
```

```
    void area() {
```

```
        int result = x1 * x2;
```

```
        System.out.println("Area of Rectangle: " + result);
```

```
}
```

```
}
```

```
class Triangle extends Shape {  
    Triangle(int base, int height) {  
        super(base, height);  
    }  
  
    void area() {  
        int result = (x1 * x2) / 2;  
        System.out.println("Area of Triangle: " + result);  
    }  
}  
  
class Circle extends Shape {  
    Circle(int radius,int zero) {  
        super(radius,zero);  
    }  
  
    void area() {  
        double result = 3.14 * x1 * x1;  
        System.out.println("Area of Circle: " + result);  
    }  
}  
  
public class Main6 {  
    public static void main(String[] args) {  
        Rectangle rect = new Rectangle(5, 4);  
    }  
}
```

```
rect.area();  
  
Triangle tri = new Triangle(5, 4);  
tri.area();  
  
Circle circ = new Circle(3, 0);  
circ.area();  
  
System.out.println("\n Name : Deepthi M ");  
System.out.println("\nUSN : 1BM23CS088 ");  
  
}  
  
}
```

OUTPUT :

```
F:\JAVA LAB\Abstract_shape>javac Main6.java  
  
F:\JAVA LAB\Abstract_shape>java Main6  
Area of Rectangle: 20  
Area of Triangle: 10  
Area of Circle: 28.259999999999998  
  
Name : Deepthi M  
  
USN : 1BM23CS088
```

# OBSERVATION :

```

import java.util.Scanner;
abstract class Shape {
    int m1;
    int m2;
    public Shape (int m1, int m2) {
        this.m1 = m1;
        this.m2 = m2;
    }
    abstract void area();
}
class Rectangle extends Shape {
    Rectangle (int l, int w) {
        super (l, w);
    }
    @Override
    void area() {
        int area = m1 * m2;
        System.out.println("Area : " + area);
    }
}

```

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

```

public Area();
triangle t1 = new triangle(3, 9);
t1.area();
circle c1 = new circle(6);
c1.area();
}

```

*seen*

~~22/10/2021~~

```

class triangle extends Shape {
    triangle (int b, int h) {
        super (b, h);
    }
    void area() {
        float area = b * h * 0.5;
        System.out.println("Area : " + area);
    }
}
class circle extends Shape {
    circle (int r) {
        super();
    }
    void area() {
        float area = 3.14 * r * r;
        System.out.println("Area : " + area);
    }
}
public class Main {
    public static void main(String args[]) {
        rectangle r1 = new rectangle(5, 10);
    }
}

```

## PROGRAM 5 :

### CODE :

```
import java.util.Scanner;

class Account {
    String customerName;
    int accountNumber;
    String accountType;
    double balance;

    Account(String name, int accNumber, String accType) {
        customerName = name;
        accountNumber = accNumber;
        accountType = accType;
        balance = 0;
    }

    public void deposit(double amount) {
        balance += amount;
        System.out.println("Deposited: " + amount + ". Updated balance: " + balance);
    }

    public void displayBalance() {
        System.out.println("Account Balance: " + balance);
    }
}
```

```
public void withdraw(double amount) {  
    System.out.println("This operation is specific to account type.");  
}  
}  
  
class SavAccount extends Account {  
    double interestRate = 0.04;  
  
    SavAccount(String name, int accNumber) {  
        super(name, accNumber, "Savings");  
    }  
  
    public void computeInterest() {  
        double interest = balance * interestRate;  
        balance += interest;  
        System.out.println("Interest added: " + interest + ". Updated balance: " + balance);  
    }  
  
    @Override  
    public void withdraw(double amount) {  
        if (balance >= amount) {  
            balance -= amount;  
            System.out.println("Withdrawn: " + amount + ". Updated balance: " + balance);  
        } else {  
            System.out.println("Insufficient balance.");  
        }  
    }  
}
```

```
    }
}

}

class CurAccount extends Account {
    double minBalance = 500.0;
    double serviceCharge = 50.0;

    CurAccount(String name, int accNumber) {
        super(name, accNumber, "Current");
    }

    public void checkMinBalance() {
        if (balance < minBalance) {
            balance -= serviceCharge;
            System.out.println("Balance below minimum. Service charge imposed: " +
serviceCharge + ". Updated balance: " + balance);
        }
    }

    @Override
    public void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println("Withdrawn: " + amount + ". Updated balance: " + balance);
            checkMinBalance();
        } else {
    }
```

```
        System.out.println("Insufficient balance.");
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter customer name:");
        String name = sc.next();

        System.out.println("Enter account number:");
        int accountnumber = sc.nextInt();

        SavAccount savingsAccount = new SavAccount(name, accountnumber);

        System.out.println("Enter customer name:");
        String name1 = sc.next();

        System.out.println("Enter account number:");
        int accountnumber1 = sc.nextInt();

        CurAccount currentAccount = new CurAccount(name1, accountnumber1);

        while (true) {
            System.out.println("\n-----MENU-----");
            System.out.println("1. Deposit\n2. Withdraw\n3. Compute Interest for Savings Account\n4. Display Account Details\n5. Exit");

            System.out.print("Enter your choice: ");
            int choice = sc.nextInt();

```

```
System.out.print("Enter the type of account (saving/current): ");
String accType = sc.nextLine();

if (accType.equals("saving")) {
    switch (choice) {
        case 1:
            System.out.print("Enter the deposit amount: ");
            double depositAmount = sc.nextDouble();
            savingsAccount.deposit(depositAmount);
            break;
        case 2:
            System.out.print("Enter the withdrawal amount: ");
            double withdrawalAmount = sc.nextDouble();
            savingsAccount.withdraw(withdrawalAmount);
            break;
        case 3:
            savingsAccount.computeInterest();
            break;
        case 4:
            System.out.println("Customer name: " + savingsAccount.customerName);
            System.out.println("Account number: " +
savingsAccount.accountNumber);
            System.out.println("Type of Account: " + savingsAccount.accountType);
            savingsAccount.displayBalance();
            break;
        case 5:
```

```
        System.exit(0);
    break;
default:
    System.out.println("Invalid choice.");
}

} else if (accType.equals("current")) {
    switch (choice) {
        case 1:
            System.out.print("Enter the deposit amount: ");
            double depositAmount = sc.nextDouble();
            currentAccount.deposit(depositAmount);
            break;
        case 2:
            System.out.print("Enter the withdrawal amount: ");
            double withdrawalAmount = sc.nextDouble();
            currentAccount.withdraw(withdrawalAmount);
            break;
        case 3:
            System.out.println("Current accounts do not earn interest.");
            break;
        case 4:
            System.out.println("Customer name: " + currentAccount.customerName);
            System.out.println("Account number: " +
currentAccount.accountNumber);
            System.out.println("Type of Account: " + currentAccount.accountType);
            currentAccount.displayBalance();
            break;
    }
}
```

```
case 5:
```

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

```
    break;
```

```
default:
```

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

```
}
```

```
}
```

```
else {
```

```
    System.out.println("Invalid account type.");
```

```
}
```

```
System.out.println("\n Name : Deepthi M ");
```

```
System.out.println("\nUSN : 1BM23CS088 ");
```

```
}
```

```
}
```

```
}
```

OUTPUT :

```
F:\JAVA LAB\Bank>javac Bank.java
```

```
F:\JAVA LAB\Bank>
F:\JAVA LAB\Bank>java Bank
Enter customer name:
ABC
Enter account number:
123
Enter customer name:
DDD
Enter account number:
333

-----MENU-----
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 1
Enter the type of account (saving/current): saving
Enter the deposit amount: 3000
Deposited: 3000.0. Updated balance: 3000.0
```

Name : Deepthi M

USN : 1BM23CS088

```
-----MENU-----
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: Exit
Exception in thread "main" java.util.InputMismatchException
```

## OBSERVATION :

29-10-21

```
import java.util.Scanner;
class Account {
    String customerName;
    int accountNumber;
    String accountType;
    double balance;
    Account (String name, int accno, String actype) {
        customerName = name;
        accountNumbers = accno;
        accountType = actype;
        balance = 0;
    }
    public void deposit (double amount) {
        balance += amount;
        System.out.println (" Deposited : " + balance + " updated balance : " + balance);
    }
    public void displayBalance () {
        System.out.println (" Account balance : " + balance);
    }
}
```

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

```
public void withdraw (double amount) {
    System.out.println (" operation specific to account type ");
}
class Savacc extends Account {
    double interestRate = 0.04;
    Savacc (String name, int accno, double amount) {
        super (name, accno, "savings");
    }
    public void computeInterest () {
        double interest = balance * interestRate;
        balance += interest;
        System.out.println ("introduced addid and update balance : " +
                           " (" + interest + ", " + balance) " by adding " + interest);
    }
    public void withdraw (double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println (" withdrawn amount and updated balance : " +
                               " (" + amount + ", " + balance) " for amount " +
                               " (" + balance - amount + ", " + balance) " by deducting " +
                               " (" + amount + ", " + balance) );
        } else {
            System.out.println ("insufficient balance");
        }
    }
}
```

```

Page 1/2
public class Current {
    double minbalance = 500.0;
    double servicecharge = 5.0;
    String name, accno;
    double current;

    public void withdraw(double amount) {
        if (amount < minbalance) {
            System.out.println("Balance below minimum balance, service charge imposed: " + servicecharge);
            current -= servicecharge;
        }
        current -= amount;
    }

    public void deposit(double amount) {
        current += amount;
    }

    public void display() {
        System.out.println("Name: " + name);
        System.out.println("Account No: " + accno);
        System.out.println("Current Balance: " + current);
    }
}

public class Bank {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter customer's name: ");
        String name = sc.nextLine();
        System.out.print("Enter account no: ");
        int accno = sc.nextInt();
        System.out.print("Enter initial balance: ");
        double balance = sc.nextDouble();

        Current curracc = new Current(name, accno);
        curracc.deposit(balance);

        curracc.withdraw(100);
        curracc.display();
    }
}

```

```

else {
    balance -= amount;
    System.out.println("New balance: " + balance);
}

public class Bank {
    public static void main(String args[]) {
        Scanner sc = new Scanner("ABC", "1", 1000);
        sc.deposit(1000);
        sc.computeInterest();
        System.out.println("Enter customer's name: ");
        String name = sc.nextLine();
        System.out.println("Enter account no: ");
        int accno = sc.nextInt();
        System.out.println("Enter initial balance: ");
        double balance = sc.nextDouble();

        SavingsAccount savacc = new SavingsAccount(name, accno);
        savacc.deposit(balance);

        savacc.withdraw(100);
        savacc.display();
    }
}

```

```

public class Bank {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter customer's name: ");
        String name = sc.nextLine();
        System.out.print("Enter account no: ");
        int accno = sc.nextInt();
        System.out.print("Enter initial balance: ");
        double balance = sc.nextDouble();

        Current curracc = new Current(name, accno);
        curracc.deposit(balance);

        curracc.withdraw(100);
        curracc.display();
    }
}

```

```

classmate
Date _____
Page _____
System.out.println("savings / current");
String acctype = sc.nextLine();
if (acctype.equals("savings")) {
    switch (choice) {
        case 1:
            System.out.println("Enter deposit amount: ");
            double deposit = sc.nextDouble();
            savingsaccount.deposit(deposit);
            break;
        case 2:
            System.out.println("Enter withdrawal amt: ");
            double withdrawal = sc.nextDouble();
            savingsaccount.withdraw(withdrawal);
            break;
        case 3:
            savingsaccount.computeInterest();
            break;
        case 4:
            System.out.println("Customer's name: " + savingsaccount.customer.name +
                " Account type: " + savingsaccount.accounttype +
                " Account no: " + savingsaccount.accno);
            savingsaccount.displayBalance();
            break;
    }
}

```

```

default : {
    System.out.println("Error");
}
{
    ((Customer) currentaccount).display();
}
else if (acc.type.equals("current")) {
    ((Current) currentaccount).display();
}
switch(choice) {
    case 1 : {
        ((Customer) currentaccount).deposit();
        System.out.println("Deposit amount: ");
        double depamt = scanner.nextDouble();
        currentaccount.withdraw(depamt);
        break;
    }
    case 2 : {
        ((Customer) currentaccount).withdraw();
        System.out.println("Withdrawal amount: ");
        double withdrawl = scanner.nextDouble();
        currentaccount.deposit(withdrawl);
        break;
    }
    case 3 : {
        ((Customer) currentaccount).showInterest();
        System.out.println("No interest on current acc");
    }
}

case 4 : {
    System.out.println("Customer name: " + currentaccount.name);
    System.out.println("Account number: " + currentaccount.accno);
    System.out.println("Account type: " + currentaccount.acctype);
    currentaccount.display();
    break;
}

case 5 : {
    ((Customer) currentaccount).display();
}

```

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

```

exit();
break;
default : {
    System.out.println("Error");
}
{
    System.out.println("Invalid account");
}
}

1. Enter customer's name:  

ABC  

Enter customer no  

123  

Enter acc  

1. Deposit  

2. Withdraw  

3. Interest  

4. Display  

5. Exit  

Enter choice 1  

Enter type of account: savings

```

## PROGRAM 6 :

### CODE :

```

package CIE;
public class Internal extends Student1
{

```

```

int[] internalmarks=new int[5];

```

```

public Internal(String name, String usn, int sem, int[] internalmarks)
{

```

```

super(usn, name, sem);

```

```
this.internalmarks=internalmarks;  
}  
  
public void displayinternalmarks()  
{  
for(int i=0;i<5;i++)  
{  
System.out.println("Subject"+(i+1)+":"+internalmarks[i]);  
}  
}  
}  
}
```

```
package CIE;  
  
public class Student1  
{  
String usn;  
String name;  
int sem;  
  
public Student1(String usn,String name,int sem)  
{  
this.usn=usn;  
this.name=name;  
this.sem=sem;  
}  
  
public void display()  
{  
System.out.println("USN:"+usn+"NAME:"+name+"SEM:"+sem);  
}
```

```
}

}

package SEE;

import CIE.Student1;

public class Externals extends Student1

{

int[] externalmarks=new int[5];

public Externals(String name,String usn,int sem,int[] externalmarks)

{



super(usn,name,sem);

this.externalmarks=externalmarks;

}

public void displayexternalmarks()

{

for(int i=0;i<5;i++)

{



System.out.println("Subject"+(i+1)+":"+externalmarks[i]);

}

}

import CIE.Internal;

import SEE.Externals;

import java.util.Scanner;

public class PackageMain{
```

```
public static void main(String args[])
{
Scanner s=new Scanner(System.in);
System.out.println("Enter usn:");
String usn=s.nextLine();
System.out.println("Enter name:");
String name=s.nextLine();
System.out.println("Enter semester:");
int sem=s.nextInt();
int imarks[]=new int[5];
int emarks[]=new int[5];

for(int i=0;i<5;i++)
{
System.out.println("Enter internal marks for Subject"+(i+1));
imarks[i]=s.nextInt();
}

for(int i=0;i<5;i++)
{
System.out.println("Enter external marks for Subject"+(i+1));
emarks[i]=s.nextInt();
}

Internal in=new Internal(usn,name,sem,imarks);
Externals e=new Externals(usn,name,sem,emarks);
in.display();
e.display();
```

```

in.displayinternalmarks();
e.displayexternalmarks();
int totalmarks=0;
for(int i=0;i<5;i++)
{
    totalmarks+=imarks[i]+emarks[i]/2;
}
System.out.println("Total Marks:"+totalmarks);
System.out.println("Name : Deepthi M \n USN : 1BM23CS088");
}
}

```

## OUTPUT :

```

F:\JAVA LAB\package>java PackageMain
Enter usn:
2
Enter name:
AAA
Enter semester:
3
Enter internal marks for Subject1
40
Enter internal marks for Subject2
15
Enter internal marks for Subject3
28
Enter internal marks for Subject4
32
Enter internal marks for Subject5
36
Enter external marks for Subject1
37
Enter external marks for Subject2
39
Enter external marks for Subject3
90
Enter external marks for Subject4
100
Enter external marks for Subject5
91
USN:AAANAME:2SEM:3
USN:AAANAME:2SEM:3
Subject1:40
Subject2:15
Subject3:28
Subject4:32
Subject5:36
Subject1:87
Subject2:89
Subject3:90
Subject4:100
Subject5:91
Total Marks:378
Name : Deepthi M
USN : 1BM23CS088

```

## OBSERVATION :

```

(2) 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 + "\nName : "
        + name + "\nSem : " + sem);
}

package CIE;
public class Internals {
    public int[] internalMarks = new int[6];
}

public class Internals (int[] marks) {
    Arrays.copyOf(this.marks, this.marks.length);
}

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

```

```

public class Main {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter no of students: ");
        int n = sc.nextInt ();
        for (int i=0; i<n; i++) {
            System.out.println ("Enter usn: ");
            String usn = sc.nextLine ();
            System.out.println ("Enter name: ");
            String name = sc.nextLine ();
            System.out.println ("Enter semester: ");
            int sem = sc.nextInt ();
            final marks & net;
            for (j=0; j<5; j++) {
                System.out.println ("Enter external marks: ");
                int intmarks = sc.nextInt ();
                externalmarks[i][j] = sc.nextInt ();
            }
            3
            External external = new External (usn, name, sem,
            finalmarks);
            System.out.println ("Final marks: ");
            for (int j=0; j<5; j++) {
                System.out.print (course[i][j] + " ");
            }
            System.out.println ();
        }
    }
}

```

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

```

sci.close();
} // question 1

class External {
    int marks;
}

class Student {
    int marks;
}

class External extends Student {
    private int marks;
    public External() {
        marks = 0;
    }
    public void setMarks(int m) {
        marks = m;
    }
    public int getMarks() {
        return marks;
    }
}

class Main {
    public static void main(String[] args) {
        External e = new External();
        Student s = new Student();
        e.setMarks(10);
        System.out.println("External Marks: " + e.getMarks());
        System.out.println("Student Marks: " + s.getMarks());
    }
}

```

## PROGRAM 7 :

### CODE :

```
interface Polygon {
```

```
    double getArea();
```

```
    double getPerimeter();
```

```
}
```

```
class Rectangle implements Polygon {
```

```
    private double length;
```

```
    private double width;
```

```
    public Rectangle(double length, double width) {
```

```
        this.length = length;
```

```
        this.width = width;
```

```
}
```

```
@Override
```

```
public double getArea() {
```

```
    return length * width;
```

```
}
```

```
@Override
```

```
public double getPerimeter() {
```

```
    return 2 * (length + width);
```

```
}
```

```
}

class Circle implements Polygon {
    private double rad;

    public Circle(double rad) {
        this.rad = rad;
    }

    @Override
    public double getArea() {
        return Math.PI * rad * rad;
    }

    @Override
    public double getPerimeter() {
        return 2 * Math.PI * rad;
    }
}

public class Main {
    public static void main(String[] args) {
        Polygon r = new Rectangle(3.5, 5);
        Polygon c = new Circle(23);

        System.out.println("Rectangle Area: " + r.getArea());
    }
}
```

```

System.out.println("Rectangle Perimeter: " + r.getPerimeter());

System.out.println("Circle Area: " + c.getArea());
System.out.println("Circle Perimeter: " + c.getPerimeter());
System.out.println("\n Name : Deepthi M ");
System.out.println("\nUSN : 1BM23CS088 ");

}

}

```

## OUTPUT :

```

F:\JAVA LAB\Interface>javac Main.java

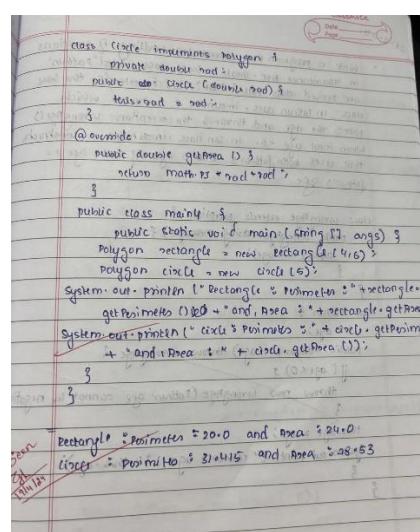
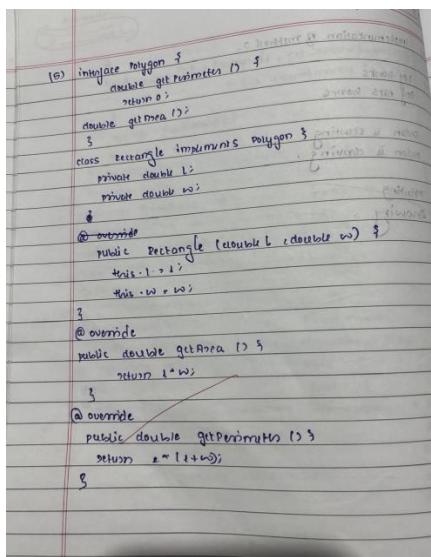
F:\JAVA LAB\Interface>java Main
Rectangle Area: 17.5
Rectangle Perimeter: 17.0
Circle Area: 1661.9025137490005
Circle Perimeter: 144.51326206513048

Name : Deepthi M

USN : 1BM23CS088

```

## OBSERVATION :



## PROGRAM 8 :

CODE : class WrongAge extends Exception

```
{  
    public WrongAge(String message)  
    {  
        super(message);  
    }  
}
```

class Father

```
{  
    int age;  
    public Father (int age) throws WrongAge  
    {  
        if(age<0)  
        {  
            throw new WrongAge("Father's age cannot be negative");  
        }  
        this.age=age;  
        System.out.println("Father's age : " +this.age);  
    }  
}
```

class Son extends Father

```
{  
    int s_age;
```

```
public Son(int f_age , int s_age) throws WrongAge
{
    super(f_age);
    if(s_age>=f_age)
    {
        throw new WrongAge("Son's age cannot be more than Father's age");
    }
    this.s_age=s_age;
    System.out.println("Son's age : " +this.s_age);
}

public class Main
{
    public static void main(String[] args)
    {
        try
        {
            Son s1= new Son(30,36);
        }
        catch(WrongAge e )
        {
            System.out.println("Exception : " +e.getMessage());
        }
        System.out.println("\n Name : Deepthi M ");
        System.out.println("\nUSN : 1BM23CS088 ");
    }
}
```

}

}

## OUTPUT :

F:\JAVA LAB\Exception>javac Main.java

F:\JAVA LAB\Exception>java Main

Father's age : 30

Exception : Son's age cannot be more than Father's age

Name : Deepthi M

USN : 1BM23CS088

## OBSERVATION :

26-11-21

While a program that demonstrates handling of exception 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 input age < 0. In Son class implement a constructor that uses both father's & son's age if son's age > father's age.

```

class WrongAge extends Exception {
    public WrongAge (String message) {
        super (message);
    }
}

class Father {
    int age;
    Father (int age) {
        if (age < 0)
            throw new WrongAge ("Father's age cannot be negative");
        this.age = age;
    }
    System.out.println ("Father's age : " + this.age);
}

```

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

```

class Son extends Father {
    public Son (int sonAge) throws WrongAge {
        super (father.age);
        if (sonAge > father.age)
            throw new WrongAge ("Son's age cannot be greater than father's age");
        this.sonAge = sonAge;
        System.out.println ("Son's age : " + this.sonAge);
    }
}

public class Main {
    public static void main (String [] args) {
        try {
            Father father = new Father (30);
            Son son = new Son (50, 30);
        } catch (WrongAge e) {
            System.out.println ("Exception: " + e.getMessage ());
        }
    }
}

```

seen  
written

```

try {
    Son son2 = new Son (40, 20);
} catch (WrongAge e) {
    System.out.println ("Exception: " + e.getMessage ());
}

```

Father's age : 30  
Exception : Son's age cannot be greater than or equal to father's age  
Exception : Father's age cannot be negative!  
Father's age : 40  
Son's age : 22.

## PROGRAM 9 :

### CODE :

```
public class ThreadProgram {  
    public static void main(String[] args) {  
        Thread t1 = new Thread(() -> {  
            while (true) {  
                System.out.println("BMSCE");  
                try {  
                    Thread.sleep(10000);  
                }  
                catch (InterruptedException e) {  
                    System.out.println("Thread t1 interrupted: " + e.getMessage());  
                }  
            }  
        });  
    }  
}
```

```
Thread t2 = new Thread(() -> {  
    while (true) {  
        System.out.println("CSE");  
        try {  
            Thread.sleep(2000);  
        }  
    }  
});
```

```

        catch (InterruptedException e)
    {
        System.out.println("Thread t2 interrupted: " + e.getMessage());
    }
}

t1.start();
t2.start();

System.out.println("\n Name : Deepthi M ");
System.out.println("\nUSN : 1BM23CS088 ");

}

}

```

OUTPUT :

```

F:\JAVA LAB\Thread>javac ThreadProgram.java
F:\JAVA LAB\Thread>java ThreadProgram
Name : Deepthi M
USN : 1BM23CS088
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE

```

OBSERVATION :

12-24.

(23) write a program which creates two threads, one thread displaying "BMSCE" once every 10 sec, and another thread once every 2 sec.

```

public class ThreadExample {
    public static void main (String args) {
        Thread t1 = new Thread () {
            while (true) {
                System.out.println ("BMSCE");
                my {
                    Thread.sleep (10000);
                }
                catch (InterruptedException e) {
                    {
                }
            }
        };
        Thread t2 = new Thread () {
            while (true) {
                System.out.println ("CSE");
                my {
                    Thread.sleep (2000);
                }
                catch (InterruptedException e) {
                    {
                }
            }
        };
    }
}

```

## PROGRAM 10 :

CODE :

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
public class IntegerDivisionUI {  
    public static void main(String[] args) {  
  
        JFrame frame = new JFrame("Integer Division Calculator");  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setSize(400, 200);  
        frame.setLayout(new GridLayout(4, 2, 10, 10));  
  
        JLabel labelNum1 = new JLabel("Num1:");  
        JTextField textNum1 = new JTextField();  
  
        JLabel labelNum2 = new JLabel("Num2:");  
        JTextField textNum2 = new JTextField();  
  
        JLabel labelResult = new JLabel("Result:");  
        JTextField textResult = new JTextField();  
        textResult.setEditable(false);
```

```
JButton buttonDivide = new JButton("Divide");

frame.add(labelNum1);
frame.add(textNum1);
frame.add(labelNum2);
frame.add(textNum2);
frame.add(labelResult);
frame.add(textResult);
frame.add(buttonDivide);

buttonDivide.addActionListener(new ActionListener() {

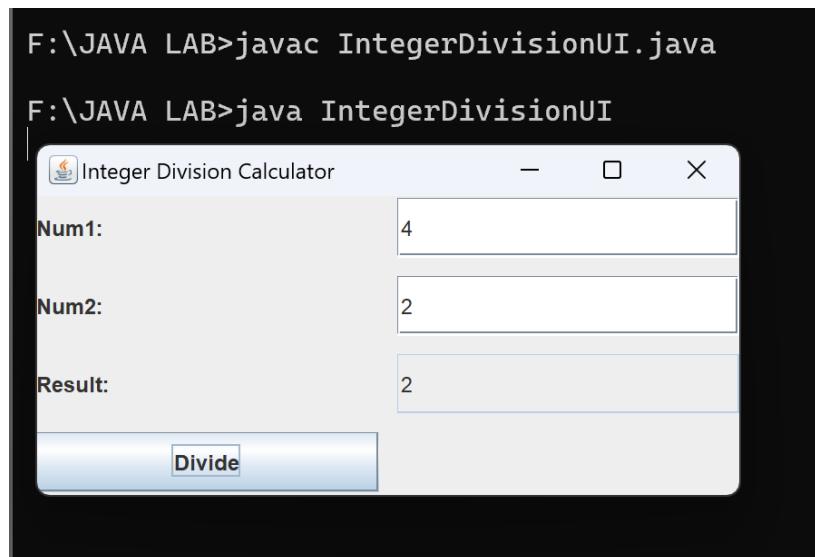
    @Override
    public void actionPerformed(ActionEvent e) {
        try {

            int num1 = Integer.parseInt(textNum1.getText());
            int num2 = Integer.parseInt(textNum2.getText());

            int result = num1 / num2;
            textResult.setText(String.valueOf(result));
        } catch (NumberFormatException ex) {
            JOptionPane.showMessageDialog(frame, "Please enter valid integers!",
                "Input Error", JOptionPane.ERROR_MESSAGE);
        } catch (ArithmaticException ex) {
```

```
JOptionPane.showMessageDialog(frame, "Division by zero is not allowed!",  
"Arithmetic Error", JOptionPane.ERROR_MESSAGE);  
}  
}  
});  
  
frame.setVisible(true);  
}  
}
```

OUTPUT :



OBSERVATION :

```

(2) import java.awt.*;
import java.awt.event.*;
import javax.swing.event.ActionEvent;
public class DivisionCalculator extends JFrame {
    private JTextField num1Field, num2Field, resultField;
    private JButton divideButton;
    public DivisionCalculator() {
        setTitle("Division Calculator");
        setSize(400, 200);
        setLayout(new FlowLayout());
        num1Field = new JTextField("Num1 : ");
        num2Field = new JTextField("Num2 : ");
        resultField = new JTextField("Result : ");
        resultField.setEditable(false);
        divideButton = new JButton("Divide");
        divideButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                int num1 = Integer.parseInt(num1Field.getText());
                int num2 = Integer.parseInt(num2Field.getText());
                int result = num1 / num2;
                resultField.setText(String.valueOf(result));
            }
        });
        add(divideButton);
        add(num1Field);
        add(num2Field);
        add(resultField);
        add(divideButton);
        add(resultField);
    }
}

```

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

```

divideButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
            String num1Text = num1Field.getText();
            String num2Text = num2Field.getText();
            int num1 = Integer.parseInt(num1Text);
            int num2 = Integer.parseInt(num2Text);
            int result = num1 / num2;
            resultField.setText(String.valueOf(result));
        } catch (NumberFormatException ex) {
            JOptionPane.showMessageDialog(null, "Input Error", "Division Error", JOptionPane.ERROR_MESSAGE);
        }
    }
});

```

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

```

catch (ArithmaticException ex) {
    JOptionPane.showMessageDialog(null, "cannot divide by zero", "Math Err", JOptionPane.ERROR_MESSAGE);
}
}
}

public static void main(String[] args) {
    SwingUtilities.invokeLater(new Runnable() {
        @Override
        public void run() {
            public void main() {
                divisionCalculator frame = new divisionCalculator();
                frame.setVisible(true);
            }
        }
    });
}

```

Scen