

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

Autonomous Institute, Affiliated to VTU



## **LAB REPORT**

**23CS3PCOOJ**

Submitted in partial fulfillment of the requirements for  
Lab Bachelor of Engineering  
in  
Computer Science and Engineering

Submitted by:

**Abhishek Gouda police patil**  
**(1BM22CS006)**

Department of Computer Science and Engineering,  
B.M.S College of Engineering, Bull Temple Road,  
Basavanagudi, Bangalore, 560 019 2023-2024.

## INDEX

| <b>Sl-No</b> | <b>Title Name</b> | <b>Date</b> | <b>Page no</b> |
|--------------|-------------------|-------------|----------------|
| <b>1</b>     | Lab Program 1     | 22-12-2024  | 1-2            |
| <b>2</b>     | Lab Program 2     | 29-12-2024  | 3-4            |
| <b>3</b>     | Lab Program 3     | 12-01-2024  | 5-7            |
| <b>4</b>     | Lab Program 4     | 12-01-2024  | 8-9            |
| <b>5</b>     | Lab Program 5     | 19-01-2024  | 10-15          |
| <b>6</b>     | Lab Program 6     | 02-02-2024  | 16-18          |
| <b>7</b>     | Lab Program 7     | 16-02-2024  | 19-21          |
| <b>8</b>     | Lab Program 8     | 16-02-2024  | 22-23          |
| <b>9</b>     | Lab Program 9     | 23-02-2024  | 24-27          |

## LAB-1

/\*Develop a Java program that prints all real solutions to the quadratic equation  $ax^2+bx+c = 0$ . Read in a, b, c and use the quadratic formula. If the discriminate  $b^2-4ac$  is negative, display a message stating that there are no real solutions.\*/

```
import java.util.Scanner;

public class Quad {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

        System.out.println("Abhishek Gouda 1BM22CS006");

        System.out.println("Enter co-efficients");

        int a = in.nextInt();

        int b = in.nextInt();

        int c = in.nextInt();

        if (a == 0) {

            System.out.println("Invalid Input");

        } else {

            double r1, r2;

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

            if (d > 0) {

                System.out.println("roots are real and distinct");

                r1 = (-b + Math.sqrt(d)) / (2 * a);

                r2 = (-b - Math.sqrt(d)) / (2 * a);

                System.out.println("r1 = " + r1 + "\n" + "r2 = " + r2);

            } else if (d < 0) {

                System.out.println("roots are imaginary");

                r1 = (-b / (2 * a));

                r2 = (Math.sqrt(Math.abs(d)) / (2 * a));

                System.out.println("r1 = " + r1 + "\n" + "r2 = " + r2 + "(i)");

            }

        }

    }

}
```

```

    } else {
        System.out.println("roots are equal");
        r1 = (-b / (2 * a));
        System.out.println("r = " + r1);
    }
}
}
}
}

```

OUTPUT :

```

C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhig\OneDrive\Desktop\javaa\week 2>javac Quad.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 2>java Quad
Abhishek Gouda  1BM22CS006
Enter co-efficients
1 2 1
roots are equal
r = -1.0

C:\Users\abhig\OneDrive\Desktop\javaa\week 2>java Quad
Abhishek Gouda  1BM22CS006
Enter co-efficients
3 2 1
roots are imaginary
r1 = 0.0
r2 = 0.47140452079103173(i)

C:\Users\abhig\OneDrive\Desktop\javaa\week 2>

```

## LAB-2

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

```
import java.util.Scanner;

class Student {

    String name, usn;

    int[] credits = new int[8];

    int[] marks = new int[8];

    public void details() {

        Scanner reader = new Scanner(System.in);

        System.out.print("Student name : ");

        name = reader.nextLine();

        System.out.print("USN : ");

        usn = reader.nextLine();

        for (int i = 0; i < 8; i++) {

            System.out.println("Course " + (i + 1) + " credit and marks");

            credits[i] = reader.nextInt();

            marks[i] = reader.nextInt();

        }

    }

    public double sgpa() {

        int totalcredits = 0, gradepoint = 0, sum = 0;

        for (int i = 0; i < 8; i++) {

            totalcredits += credits[i];

            gradepoint = marks[i] / 10 + 1;

            if (gradepoint == 11) {

                gradepoint = 10;

            }

        }

        return (double) totalcredits / gradepoint;

    }

}
```

```

        } else if (gradepoint <= 4) {
            gradepoint = 0;
        }
        sum += gradepoint * credits[i];
    }
    return (double) sum / totalcredits;
}
}

public class Sgpa{
    public static void main(String[] args){
        System.out.println("Abhishek Gouda 1BM22CS006");
        Student student = new Student();
        student.details();
        System.out.println(student.name + "\t" + student.usn);
        System.out.println("SGPA : " + student.sgpa());
    }
}

```

Output:

```

C:\Windows\System32\cmd.e
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhig\OneDrive\Desktop\javaa\week 3>javac Sgpa.java
C:\Users\abhig\OneDrive\Desktop\javaa\week 3>java Sgpa
Abhishek Gouda 1BM22CS006
Student name : Abhi
USN : 1BM22cs006
Course 1 credit and marks
3 60
Course 2 credit and marks
2 70
Course 3 credit and marks
1 100
Course 4 credit and marks
4 91
Course 5 credit and marks
4 89
Course 6 credit and marks
2 78
Course 7 credit and marks
3 99
Course 8 credit and marks
4 88
Abhi 1BM22cs006
SGPA : 8.91304347826087

```

### **LAB-3**

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

```
class Book {  
    private String name;  
    private String author;  
    private double price;  
    private int numPages;  
    public Book(String name, String author, double price, int numPages) {  
        this.name = name;  
        this.author = author;  
        this.price = price;  
        this.numPages = numPages;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public void setAuthor(String author) {  
        this.author = author;  
    }  
    public void setPrice(double price) {  
        this.price = price;  
    }  
    public void setNumPages(int numPages) {  
        this.numPages = numPages;  
    }  
}
```

```

    public String getName() {
        return name;
    }
    public String getAuthor() {
        return author;
    }
    public double getPrice() {
        return price;
    }
    public int getNumPages() {
        return numPages;
    }
    @Override
    public String toString() {
        return "Book Details\n Name=" + name + ", Author=" + author + ", Price=" + price + ",
numPages=" + numPages + "\n";
    }
}

public class Program3 {
    public static void main(String[] args) {
        int n = 2;

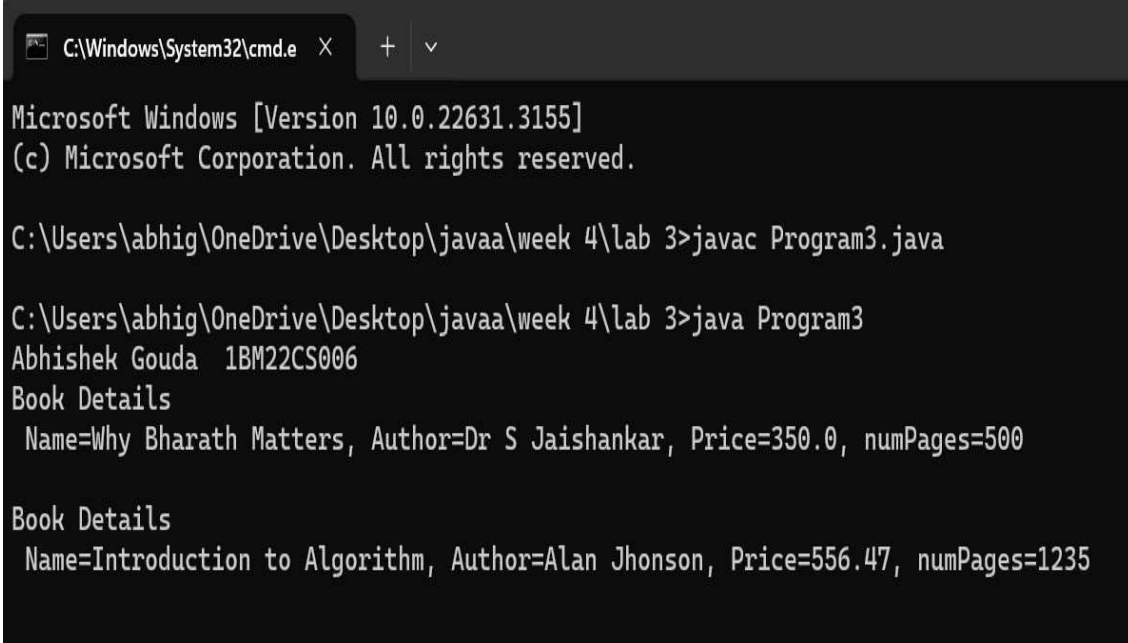
        System.out.println("Abhishek Gouda 1BM22CS006");

        Book[] books = new Book[n];
        books[0] = new Book("Why Bharath Matters", "Dr S Jaishankar", 350, 500);
        books[1] = new Book("Introduction to Algorithm", "Alan Jhonson", 556.47, 1235);
        for (Book b : books) {
            System.out.println(b);
        }
    }
}

```



Output:



```
C:\Windows\System32\cmd.e X + v
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhig\OneDrive\Desktop\javaa\week 4\lab 3>javac Program3.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 4\lab 3>java Program3
Abhishek Gouda 1BM22CS006
Book Details
Name=Why Bharath Matters, Author=Dr S Jaishankar, Price=350.0, numPages=500

Book Details
Name=Introduction to Algorithm, Author=Alan Jhonson, Price=556.47, numPages=1235
```

## **LAB-4**

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

```
abstract class Shape {  
    double x, y;  
    abstract void print_area();  
}  
  
class Rectangle extends Shape {  
    Rectangle(double x, double y) {  
        this.x = x;  
        this.y = y;  
    }  
    public void print_area() {  
        System.out.println("Rectangle area : " + x * y);  
    }  
}  
  
class Triangle extends Shape {  
    Triangle(double x, double y) {  
        this.x = x;  
        this.y = y;  
    }  
    public void print_area() {  
        System.out.println("Triangle area : " + x * y * 0.5);  
    }  
}  
  
class Circle extends Shape {  
    Circle(double x) {
```

```

        this.x = x;
    }

    public void print_area() {
        System.out.println("Circle area : " + 3.14 * x * x);
    }
}

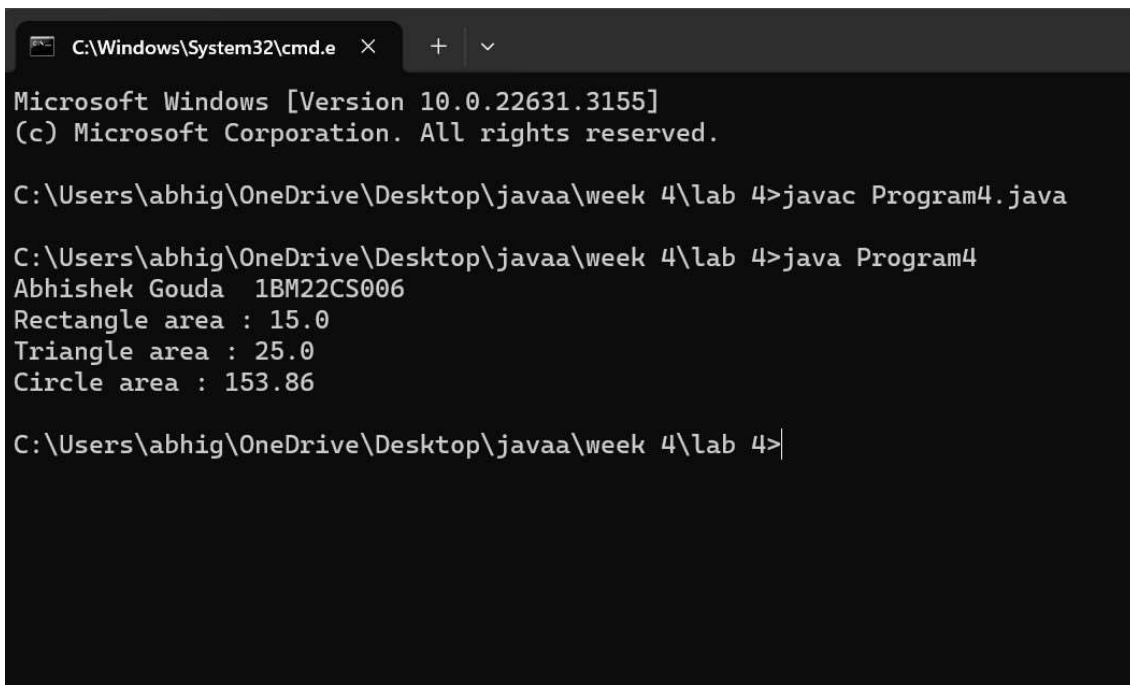
public class Program4 {
    public static void main(String[] args) {
        System.out.println("Abhishek Gouda 1BM22CS006");

        Rectangle r = new Rectangle(5, 3);
        Triangle t = new Triangle(10, 5);
        Circle c = new Circle(7);

        r.print_area();
        t.print_area();
    }
}

```

Output:



```

C:\Windows\System32\cmd.e
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhig\OneDrive\Desktop\javaa\week 4\lab 4>javac Program4.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 4\lab 4>java Program4
Abhishek Gouda 1BM22CS006
Rectangle area : 15.0
Triangle area : 25.0
Circle area : 153.86

C:\Users\abhig\OneDrive\Desktop\javaa\week 4\lab 4>

```

## **LAB-5**

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

Create a class Account that stores customer name, account number and type of account. From this derive the classes 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

Check for the minimum balance, impose penalty if necessary and update the balance. \*/

```
class Account {  
    private String customerName;  
    private int accountNumber;  
    private String accountType;  
    private double balance;  
    public Account(String customerName, int accountNumber, String accountType, double  
balance) {  
        this.customerName = customerName;  
        this.accountNumber = accountNumber;  
        this.accountType = accountType;  
        this.balance = balance;  
    }  
}
```

```

public void setCustomerName(String customerName) {
    this.customerName = customerName;
}

public void setAccountNumber(int accountNumber) {
    this.accountNumber = accountNumber;
}

public void setAccountType(String accountType) {
    this.accountType = accountType;
}

public void setBalance(double balance) {
    this.balance = balance;
}

public String getCustomerName() {
    return customerName;
}

public int getAccountNumber() {
    return accountNumber;
}

public String getAccountType() {
    return accountType;
}

public double getBalance() {
    return balance;
}

public void deposit(double amount) {
    if (amount > 0) {
        balance += amount;
        System.out.println("Deposit of " + amount + " Successful");
        System.out.println("New balance : " + balance);
    }
}

```

```

        return;
    }

    System.out.println("Invalid amount. Deposit Failed");
}

public void displayBalance() {
    System.out.println("The balance of acc no. " + accountNumber + " is " + balance);
}

public void compoundInterest() {
}

public void withdraw(double amount) {
    if (amount > 0 && amount <= balance) {
        balance -= amount;
        System.out.println("Withdrawal of " + amount + " Successful");
        System.out.println("New balance : " + balance);
        return;
    } else if (amount > balance) {
        System.out.println("Insufficient Balance");
    } else if (amount <= 0) {
        System.out.println("Invalid amount");
    }
    System.out.println("Withdraw Failed");
}

public void checkBalance() {
}
}

class sav_acct extends Account {
    private double interestRate;
    private double interest;

```

```

    public sav_acct(String customerName, int accountNumber, String accountType, double
balance, double interestRate) {
        super(customerName, accountNumber, accountType, balance);
        this.interestRate = interestRate;
    }
    public void setInterestRate(double interestRate) {
        this.interestRate = interestRate;
    }
    public void setInterest(double interest) {
        this.interest = interest;
    }
    public double getInterestRate() {
        return interestRate;
    }
    public double getInterest() {
        return interest;
    }
    public void compoundInterest() {
        interest = getBalance() * (1 + (0.5 / 12)) - getBalance();
        deposit(interest);
    }
}

class cur_acct extends Account {
    private double minBalance;
    private double serviceCharge;

    public cur_acct(String customerName, int accountNumber, String accountType, double
balance, double serviceCharge,
        double minBalance) {
        super(customerName, accountNumber, accountType, balance);
        this.minBalance = minBalance;
    }
}

```

```

        this.serviceCharge = serviceCharge;
    }

    public void setMinBalance(double minBalance) {
        this.minBalance = minBalance;
    }

    public void setServiceCharge(double serviceCharge) {
        this.serviceCharge = serviceCharge;
    }

    public double getMinBalance() {
        return minBalance;
    }

    public double getServiceCharge() {
        return serviceCharge;
    }

    public void checkBalance() {
        if (getBalance() < minBalance) {
            setBalance(getBalance() - serviceCharge);
            System.out.println("Low balance warning!");
        }
    }
}

public class Bank {

    public static void main(String[] args) {

        System.out.println("Abhishek Gouda 1BM22CS006");

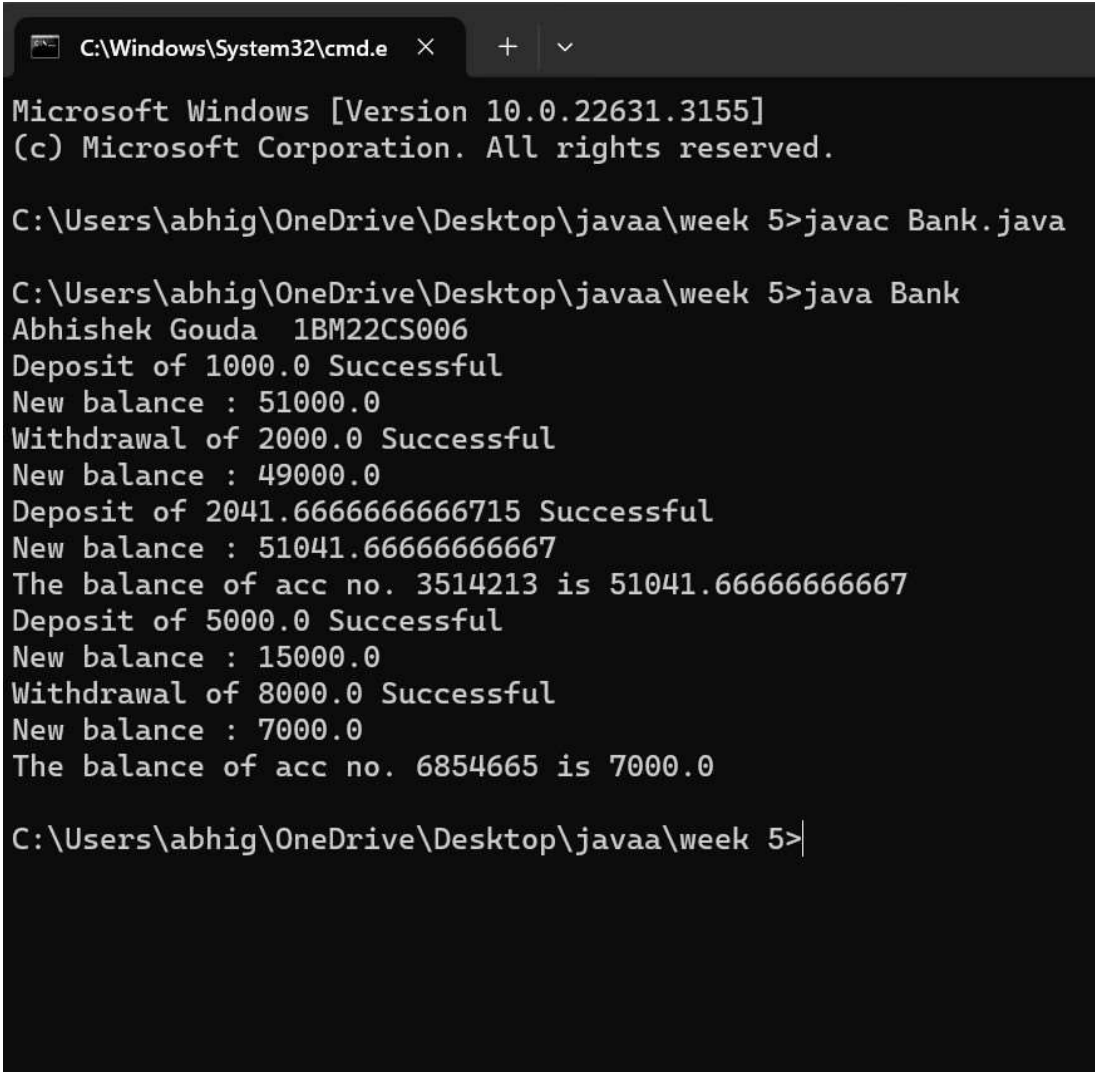
        sav_acct sav = new sav_acct("Amba singh", 3514213, "savings", 50000, 5);
        sav.deposit(1000);
        sav.withdraw(2000);
        sav.compoundInterest();
        sav.displayBalance();
    }
}

```



```
cur_acct cur = new cur_acct("Loli Prakash", 6854665, "current", 10000, 500, 1000);  
cur.deposit(5000);  
cur.withdraw(8000);  
cur.checkBalance();  
cur.displayBalance();  
  
}
```

Output:



```
C:\Windows\System32\cmd.e  X  +  v  
Microsoft Windows [Version 10.0.22631.3155]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\abhig\OneDrive\Desktop\javaa\week 5>javac Bank.java  
  
C:\Users\abhig\OneDrive\Desktop\javaa\week 5>java Bank  
Abhishek Gouda  1BM22CS006  
Deposit of 1000.0 Successful  
New balance : 51000.0  
Withdrawal of 2000.0 Successful  
New balance : 49000.0  
Deposit of 2041.66666666666715 Successful  
New balance : 51041.666666666667  
The balance of acc no. 3514213 is 51041.666666666667  
Deposit of 5000.0 Successful  
New balance : 15000.0  
Withdrawal of 8000.0 Successful  
New balance : 7000.0  
The balance of acc no. 6854665 is 7000.0  
  
C:\Users\abhig\OneDrive\Desktop\javaa\week 5>|
```

## **LAB-6**

/\* Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses. \*/

//CIE PACKAGE

//Student.java

```
package CIE;

import java.util.*;

public class Student
{
    public int sem;
    public String usn;
    public String name;
    public void accept()
    {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter the details :\n");
        usn=scan.nextLine();
        name=scan.nextLine();
        sem=scan.nextInt();
    }
}
```

//Internals.java

```
package CIE;

public class Internal {
    public int im[] = new int[5];
}
```

```

//SEE PACKAGE
//External.java
package SEE;
import CIE.Student;
public class External extends Student {
    public int sm[] = new int[5];
}

//FinalMarks.java
import java.util.*;
import SEE.*;
import CIE.*;
public class Finalmarks
{
    public static void main(String args[])
    {
        int fm[]=new int[5];
        Scanner sc= new Scanner(System.in);
        System.out.println("Enter n: ");
        int n=sc.nextInt();
        SEE.External st[]=new SEE.External[n];
        CIE.Internal s[]=new CIE.Internal[n];
        for(int i=0; i<n; i++)
        {
            st[i]=new SEE.External();
            s[i]=new CIE.Internal();
            System.out.println("Enter details "+(i+1));
            st[i].accept();
            for(int j=0; j<5; j++)
            {

```

```

        System.out.println("Enter im and sm of sub "+(j+1));

        s[i].im[j]=sc.nextInt();

        st[i].sm[j]=sc.nextInt();

        fm[j]=s[i].im[j]+st[i].sm[j];

    }

    System.out.println("Final marks of "+st[i].name);

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

    {

        System.out.println("Course "+(k+1)+" = "+fm[k]);

    }

}

}
}

```

```

C:\Users\arbaa\Desktop\CS0 >javac FinalMarks.java

C:\Users\arbaa\Desktop\CS0 >java FinalMarks
Enter n:
1
Enter details for student 1
Enter USN, Name, and Semester:
008
Abhishek
3
Enter im and sm of subject 1
15
90
Enter im and sm of subject 2
36
92
Enter im and sm of subject 3
40
99
Enter im and sm of subject 4
34
98
Enter im and sm of subject 5
35
90
Final marks of Abhishek
Subject 1 = 105
Subject 2 = 128
Subject 3 = 139
Subject 4 = 132
Subject 5 = 125

```

## LAB-7

/\*Write a program that demonstrates handling of exceptions in inheritance tree.

Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.\*/

```
import java.util.Scanner;

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

class Father {
    public int FatherAge;

    Father(int FatherAge) throws WrongAge {
        if (FatherAge <= 0)
            throw new WrongAge("Father's Age is negative");
        this.FatherAge = FatherAge;
    }
}

class Son extends Father {
    public int SonAge;

    Son(int SonAge, int FatherAge) throws WrongAge {
        super(FatherAge);
        if (SonAge >= FatherAge)
            throw new WrongAge("Son's Age should be less than Father's Age");
        if (SonAge <= 0)
```

```

        throw new WrongAge("Son's Age is negative");
    this.SonAge = SonAge;
}
}

public class FatherSon {
    public static void main(String[] args) {
        System.out.println("Abhishek Gouda 1BM22CS006");
        Scanner s = new Scanner(System.in);
        int f_age, s_age;
        try {
            System.out.print("Father's Age : ");
            f_age = s.nextInt();
            System.out.print("Son's Age : ");
            s_age = s.nextInt();
            // Check for invalid ages and throw specific WrongAge exceptions
            if (f_age < 0) {
                throw new WrongAge("Father's Age is negative");
            } else if (s_age < 0) {
                throw new WrongAge("Son's Age is negative");
            } else if (s_age >= f_age) {
                throw new WrongAge("Son's Age should be less than Father's Age");
            }
            // Create Son object only if all conditions are met
            Son son = new Son(s_age, f_age); // No need for inner try-catch
        } catch (WrongAge wa) {
            System.out.println(wa);
        }
    }
}

```

Output:

```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>javac FatherSon.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>java FatherSon
Abhishek Gouda 1BM22CS006
Father's Age : 50
Son's Age : -10
WrongAge: Son's Age is negative

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>javac FatherSon.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>java FatherSon
Abhishek Gouda 1BM22CS006
Father's Age : -20
Son's Age : 10
WrongAge: Father's Age is negative

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>javac FatherSon.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>java FatherSon
Abhishek Gouda 1BM22CS006
Father's Age : 20
Son's Age : 50
WrongAge: Son's Age should be less than Father's Age

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 7>
```

## LAB-8

/\*Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.\*/

```
class A extends Thread {  
    public void run() {  
        try {  
            for (int i = 0; i < 5; i++) {  
                System.out.println("BMS College of Engineering");  
                Thread.sleep(10000);  
            }  
        } catch (InterruptedException ie) {  
        }  
    }  
}
```

```
class B extends Thread {  
    public void run() {  
        try {  
            for (int i = 0; i < 5; i++) {  
                System.out.println("CSE");  
                Thread.sleep(2000);  
            }  
        } catch (InterruptedException ie) {  
        }  
    }  
}
```

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



```
        System.out.println("Abhishek Gouda 1BM22CS006");

        A a = new A();

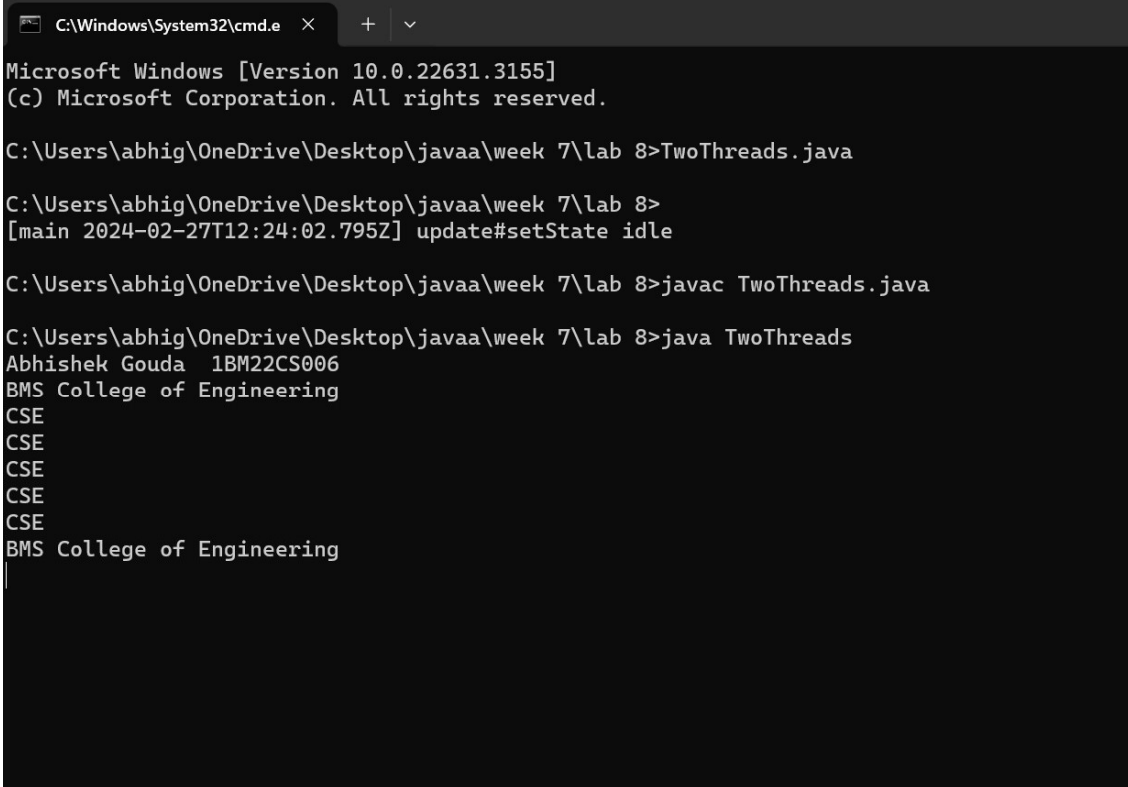
        B b = new B();

        a.start();

        b.start();

    }
```

Output:



```
C:\Windows\System32\cmd.e  X  +  v

Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 8>TwoThreads.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 8>
[main 2024-02-27T12:24:02.795Z] update#setState idle

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 8>javac TwoThreads.java

C:\Users\abhig\OneDrive\Desktop\javaa\week 7\lab 8>java TwoThreads
Abhishek Gouda 1BM22CS006
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
|
```

## **LAB-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 ArithmeticException. Display the exception in a message dialog box.

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;

public class LastLab {

    public LastLab() {

        JFrame jfrm = new JFrame("Divider App");

        jfrm.setSize(300, 300);

        jfrm.setLayout(new FlowLayout());

        jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JLabel jlab = new JLabel("Enter the divider and dividend : ");

        JTextField ajtf = new JTextField(10);

        JTextField bjtf = new JTextField(10);

        JButton button = new JButton("Calculate");

        JLabel err = new JLabel();

        JLabel alab = new JLabel();

        JLabel blab = new JLabel();

        JLabel anslab = new JLabel();

        jfrm.add(err);
```

```
jfrm.add(jlab);  
jfrm.add(ajtff);  
jfrm.add(bjtf);  
jfrm.add(button);  
jfrm.add(alab);  
jfrm.add(blabb);  
jfrm.add(anslab);
```

```
ActionListener l = new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {  
        System.out.println("Action event from a text field");  
    }  
};  
ajtff.addActionListener(l);  
bjtf.addActionListener(l);
```

```
button.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {  
        try {  
            int a = Integer.parseInt(ajtff.getText());  
            int b = Integer.parseInt(bjtf.getText());  
            double ans = a / (double) b;  
  
            alab.setText("\nA = " + a);  
            blabb.setText("\nB = " + b);  
            anslab.setText("\nAns = " + ans);  
        } catch (NumberFormatException e) {  
            alab.setText("");  
            blabb.setText("");  
        }  
    }  
});
```

```

        anslab.setText("");
        err.setText("Enter only Integers\n");
    } catch (ArithmeticException e) {
        alab.setText("");
        blab.setText("");
        anslab.setText("");
        err.setText("B should be NON zero!\n");
    }
}

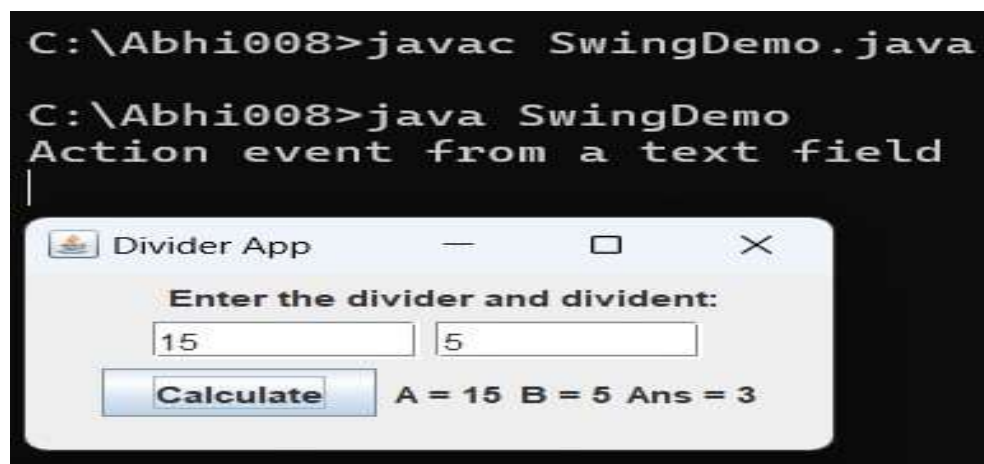
});

jfrm.setVisible(true);
}

public static void main(String[] args) {
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new LastLab();
        }
    });
}
}

```

Output1:

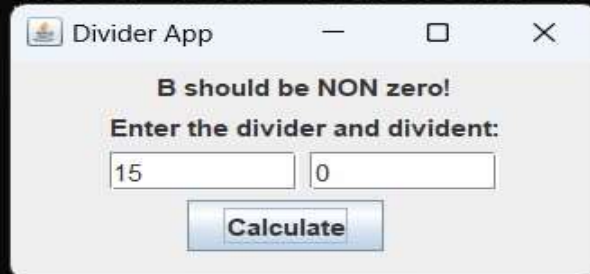


```
C:\Abhi008>java SwingDemo  
Action event from a text field
```



The screenshot shows a Java Swing window titled "Divider App". Inside the window, the text "Enter Only Integers!" is displayed in bold. Below it, the prompt "Enter the divider and dividend:" is shown. There are two text input fields: the first contains "12a" and the second contains "6". Below the input fields is a button labeled "Calculate".

```
C:\Abhi008>java SwingDemo  
Action event from a text field  
Action event from a text field
```



The screenshot shows the same "Divider App" window. The text "B should be NON zero!" is now displayed in bold above the input fields. The prompt "Enter the divider and dividend:" remains. The first text input field now contains "15" and the second contains "0". The "Calculate" button is still present below the fields.