VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

Sambriddhi Prasai (1BF24CS268)

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

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
Aug-2025 to Jan-2026

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 **Sambriddhi Prasai(1BF24CS268),** 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.

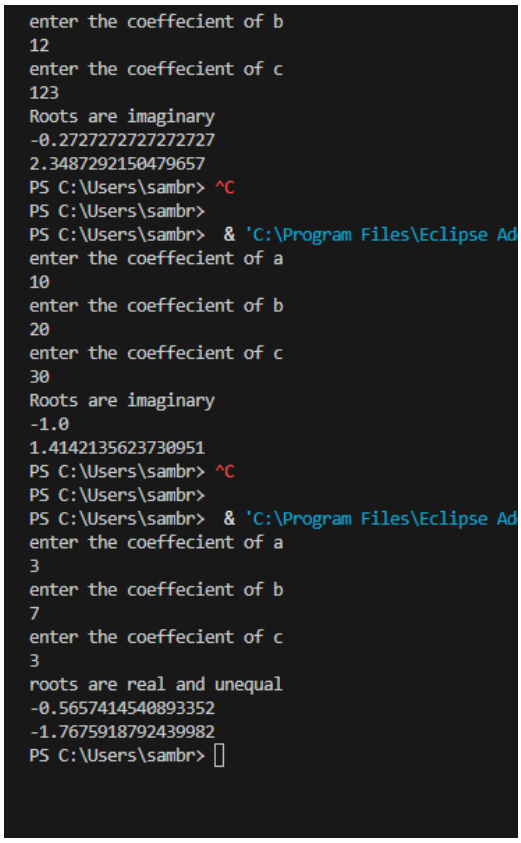
Dr. Seema Patil Associate Professor Department of CSE, BMSCE	Dr. Kavitha Sooda Professor & HOD Department of CSE, BMSCE

Index

GitHub Link: https://github.com/sambriddhi06/JAVA-

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.



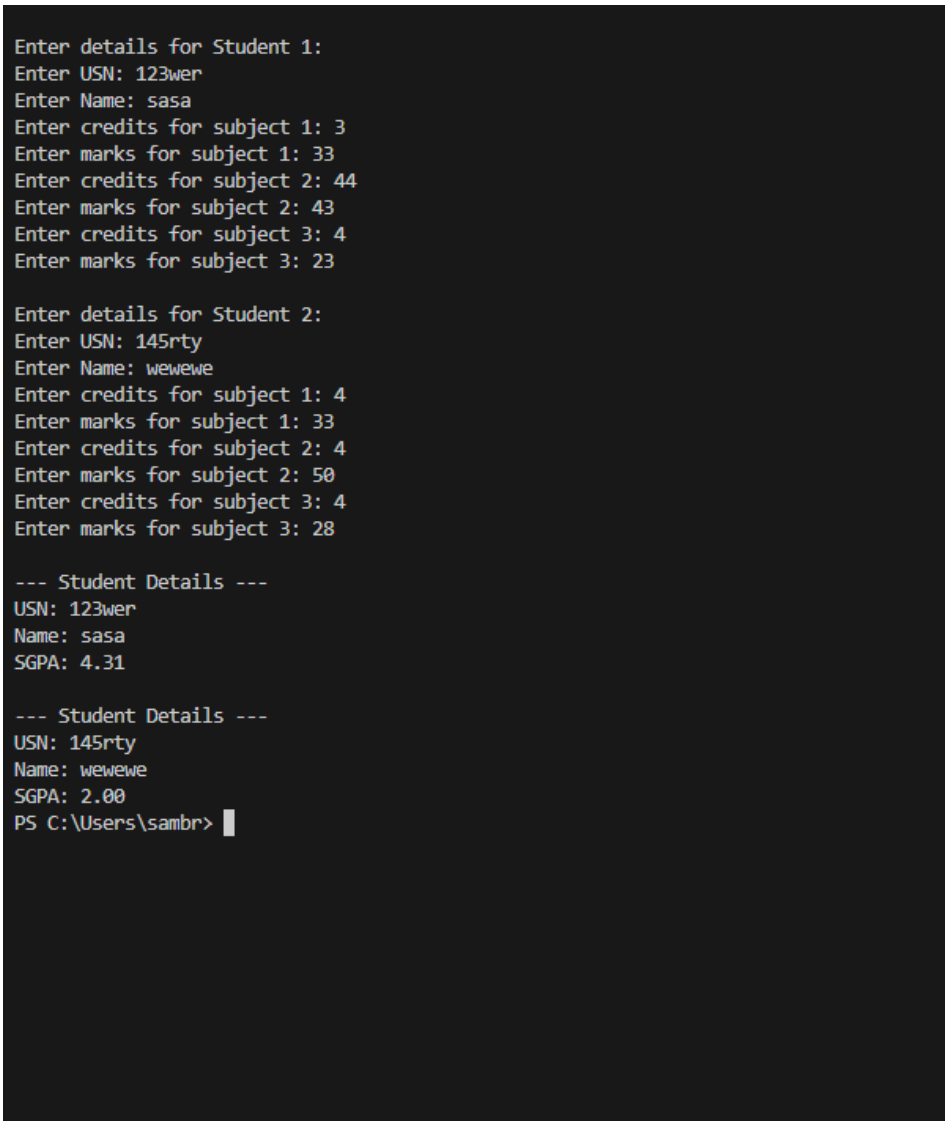
Code:

```
import java.util.Scanner;
import java.lang.Math;
class Quadratic
{
   public static void main (String[] args)
{
   double a,b,c;
   double d,r1,r2;
```

```
    Scanner in =new Scanner(System.in);
    System.out.println("enter the coeffecient of a");
    a=in.nextDouble();
    System.out.println("enter the coeffecient of b");
    b=in.nextDouble();
    System.out.println("enter the coeffecient of c");
    c=in.nextDouble();
    if(a==0)
    {
     System.out.println("not a quadratic equation");
     System.out.println("enter new non zero value of a");
     a=in.nextDouble();
    }

    d= b*b - 4*a*c;
    if(d==0)
    {
       r1 = (-b)/(2*a);
       System.out.println("roots are real and equal");
       System.out.println(r1);
    }
    else if (d>0)
    {
       r1 = ((-b) + (Math.sqrt(d)))/(double)(2*a);
       r2 = ((-b) - (Math.sqrt(d)))/(double)(2*a);
       System.out.println("roots are real and unequal");
       System.out.println(r1);
       System.out.println(r2);
    }
    else if (d<0)
    {
       System.out.println("Roots are imaginary");
       r1 = (-b)/(2*a);
       r2 = Math.sqrt(-d)/(2*a);
       System.out.println(r1);
       System.out.println(r2);

    }
}
}
```

Program 2

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




```
import java.util.Scanner;

class Student {
```

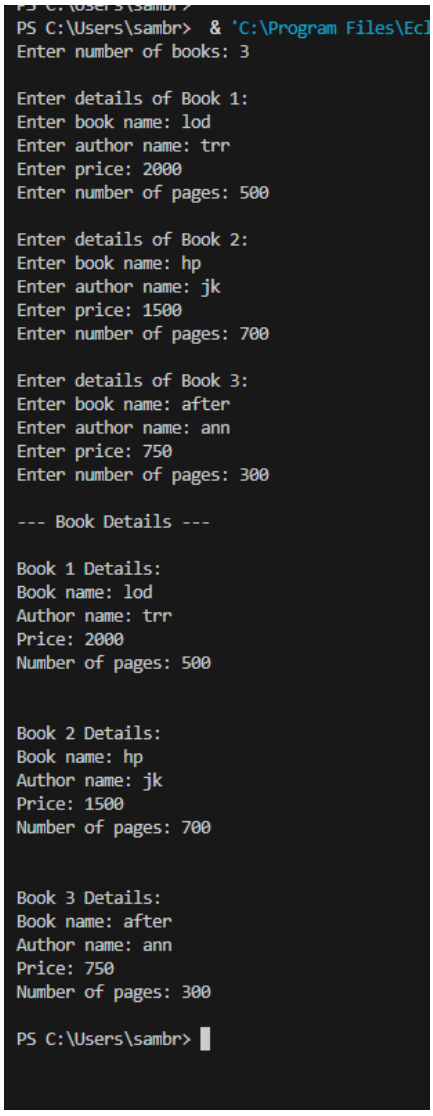
```
String usn;
String name;
int[] credits;
int[] marks;
double sgpa;

Student(int n) {
    credits = new int[n];
    marks = new int[n];
}

void acceptDetails(Scanner sc, int n) {
    System.out.print("Enter USN: ");
    usn = sc.nextLine();
    System.out.print("Enter Name: ");
    name = sc.nextLine();

    for (int i = 0; i < n; i++) {
        System.out.print("Enter credits for subject " + (i + 1) + ": ");
        credits[i] = sc.nextInt();
        System.out.print("Enter marks for subject " + (i + 1) + ": ");
        marks[i] = sc.nextInt();
    }
    sc.nextLine();
}

void calculateSGPA(int n) {
    int totalCredits = 0;
    int totalPoints = 0;

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

        if (marks[i] >= 90)
            gradePoint = 10;
```

```
            else if (marks[i] >= 80)
                gradePoint = 9;
            else if (marks[i] >= 70)
                gradePoint = 8;
            else if (marks[i] >= 60)
                gradePoint = 7;
            else if (marks[i] >= 50)
                gradePoint = 6;
            else if (marks[i] >= 40)
                gradePoint = 5;
            else
                gradePoint = 0;

            totalPoints += gradePoint * credits[i];
            totalCredits += credits[i];
        }

        sgpa = (double) totalPoints / totalCredits;
    }

    void displayDetails() {
        System.out.println("\n--- Student Details ---");
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
        System.out.println("SGPA: " + String.format("%.2f", sgpa));
    }
}

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

        System.out.print("Enter number of subjects: ");
        int n = sc.nextInt();
        sc.nextLine();
```

```
        Student s1 = new Student(n);
        Student s2 = new Student(n);

        System.out.println("\nEnter details for Student 1:");
        s1.acceptDetails(sc, n);
        s1.calculateSGPA(n);

        System.out.println("\nEnter details for Student 2:");
        s2.acceptDetails(sc, n);
        s2.calculateSGPA(n);

        s1.displayDetails();
        s2.displayDetails();

        sc.close();
    }
}
```

Program 3

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



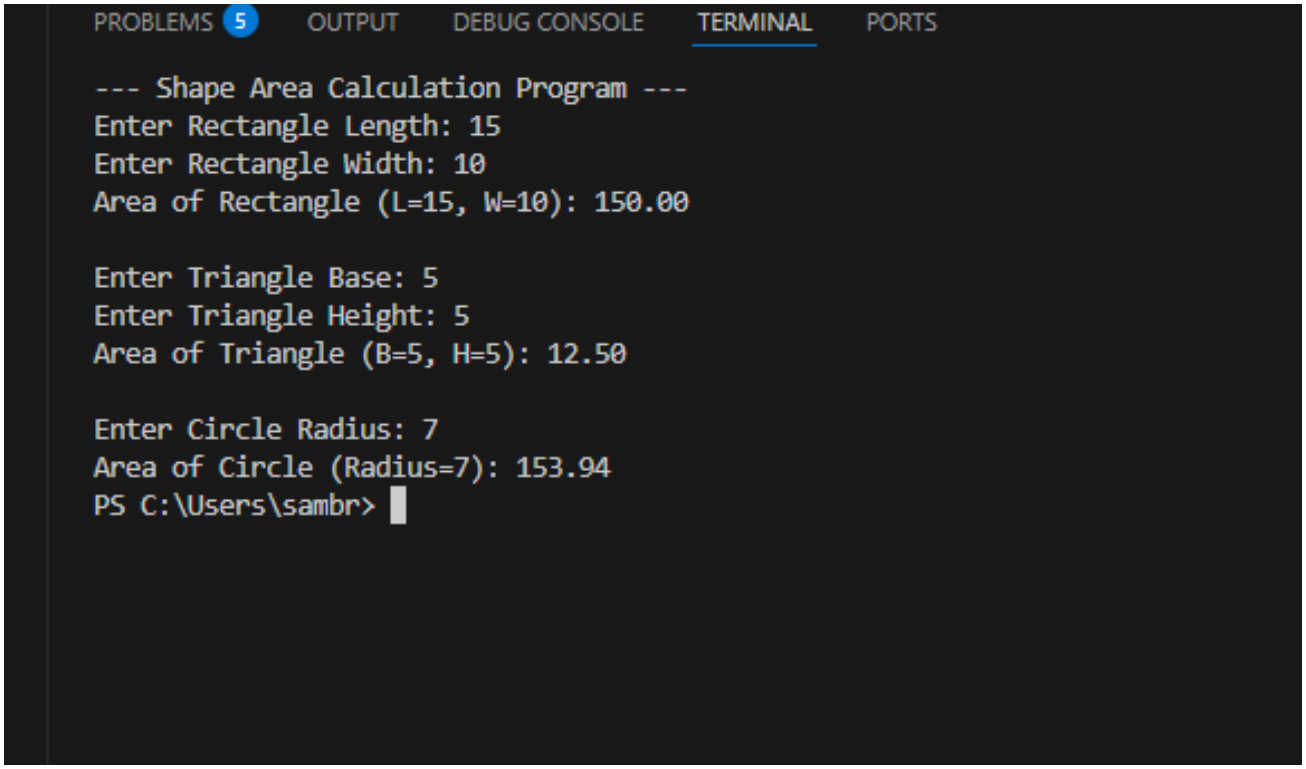
Code:

```
import java.util.Scanner;

class Books {
   String name;
   String author;
   int price;
   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 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;
   }
}

public class BookMain {
   public static void main(String args[]) {
      Scanner s = new Scanner(System.in);
      int n, price, numPages;
      String name, author;
```

```
        System.out.print("Enter number of books: ");
        n = s.nextInt();
        s.nextLine();

        Books[] b = new Books[n];

        for (int i = 0; i < n; i++) {
            System.out.println("\nEnter details of Book " + (i + 1) + ":");

            System.out.print("Enter book name: ");
            name = s.nextLine();

            System.out.print("Enter author name: ");
            author = s.nextLine();

            System.out.print("Enter price: ");
            price = s.nextInt();

            System.out.print("Enter number of pages: ");
            numPages = s.nextInt();
            s.nextLine();
            b[i] = new Books(name, author, price, numPages);
        }

        System.out.println("\n--- Book Details ---");
        for (int i = 0; i < n; i++) {
            System.out.println("\nBook " + (i + 1) + " Details:");
            System.out.println(b[i]);
        }

        s.close();
    }
}
```

Program 4

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape



Code:

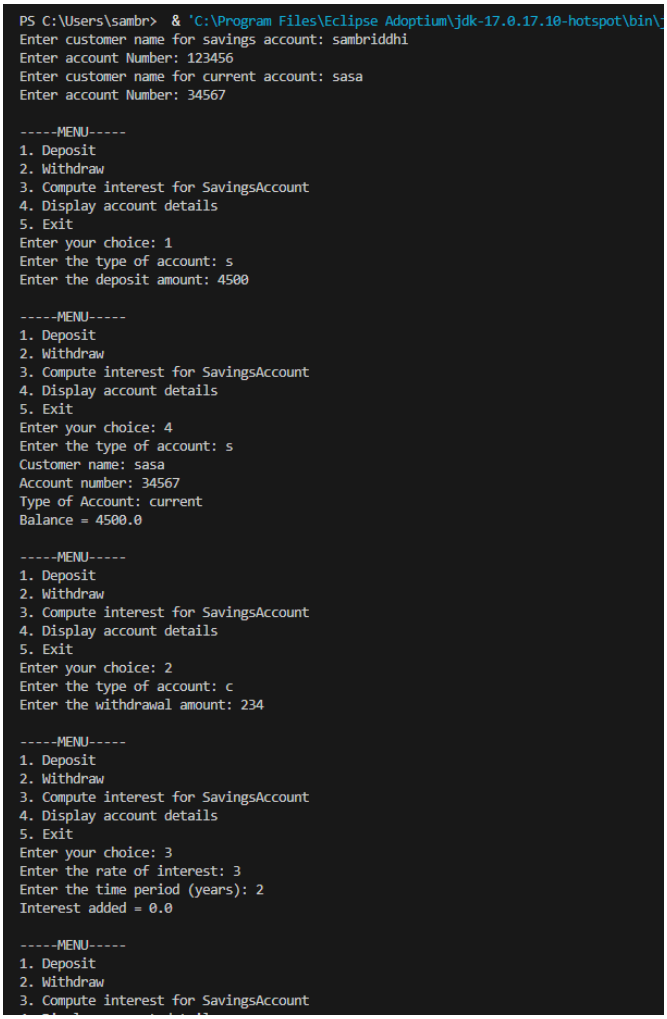
```
import java.util.Scanner;
abstract class Shape {
    int value1;
    int value2;
    public abstract void printArea();
}
class Rectangle extends Shape {
```

```
    public Rectangle(int length, int width) {
        this.value1 = length;
        this.value2 = width;
    }
    public void printArea() {
        // Area = Length * Width
        double area = (double)value1 * value2;
        System.out.printf("Area of Rectangle (L=%d, W=%d): %.2f\n", value1, value2,
area);
    }
}
class Triangle extends Shape {
    public Triangle(int base, int height) {
        this.value1 = base;
        this.value2 = height;
    }
    public void printArea() {
        // Area = 0.5 * Base * Height
        double area = 0.5 * value1 * value2;
        System.out.printf("Area of Triangle (B=%d, H=%d): %.2f\n", value1, value2,
area);
    }
}
class Circle extends Shape {
    private static final double PI = Math.PI;
    public Circle(int radius) {
        this.value1 = radius;
        this.value2 = 0;
    }
    public void printArea() {
        double area = PI * value1 * value1;
        System.out.printf("Area of Circle (Radius=%d): %.2f\n", value1, area);
    }
}
public class shapearea {
```

```
    public static void main(String[] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.println("--- Shape Area Calculation Program ---");
        System.out.print("Enter Rectangle Length: ");
        int rLength = scanner.nextInt();
        System.out.print("Enter Rectangle Width: ");
        int rWidth = scanner.nextInt();
        Rectangle rect = new Rectangle(rLength, rWidth);
        rect.printArea();
        System.out.print("\nEnter Triangle Base: ");
        int tBase = scanner.nextInt();
        System.out.print("Enter Triangle Height: ");
        int tHeight = scanner.nextInt();
        Triangle tri = new Triangle(tBase, tHeight);
        tri.printArea();
        System.out.print("\nEnter Circle Radius: ");
        int cRadius = scanner.nextInt();
        Circle circ = new Circle(cRadius);
        circ.printArea();

        scanner.close();

    }
}
```

Program 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.



Code:

```
import java.util.Scanner;
class Account {
   String customerName;
   int accountNumber;
   String accountType;
   double balance;
   void getAccountDetails()
   {
      Scanner s = new Scanner(System.in);
      System.out.print("Enter customer name: ");
      customerName = s.next();
      System.out.print("Enter account Number: ");
      accountNumber = s.nextInt();
      System.out.print("Enter type of account (saving/current): ");
      accountType = s.next();
      balance = 0;
   }
   void display() {
      System.out.println("Customer name: " + customerName);
      System.out.println("Account number: " + accountNumber);
      System.out.println("Type of Account: " + accountType);
      System.out.println("Balance = " + balance);
   }
}
class Sav_acct extends Account {
   void deposit()
   {
      Scanner s = new Scanner(System.in);
      System.out.print("Enter the deposit amount: ");
      double amount = s.nextDouble();
      balance += amount;
   }
   void withdraw()
```

```
    {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the withdrawal amount: ");
        double amount = s.nextDouble();
        if (amount > balance) {
            System.out.println("Insufficient balance!");
        } else {
            balance -= amount;
        }
    }
    void computeInterest()
    {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the rate of interest: ");
        double rate = s.nextDouble();
        System.out.print("Enter the time period (years): ");
        int time = s.nextInt();
        double interest = balance * Math.pow((1 + rate / 100), time) - balance;
        balance += interest;
        System.out.println("Interest added = " + interest);
    }
}
class Cur_acct extends Account {
    final double minBalance = 500;  // minimum required balance
    final double serviceCharge = 100;
    void deposit()
    {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the deposit amount: ");
        double amount = s.nextDouble();
        balance += amount;
    }
    void withdraw()
    {
        Scanner s = new Scanner(System.in);
```

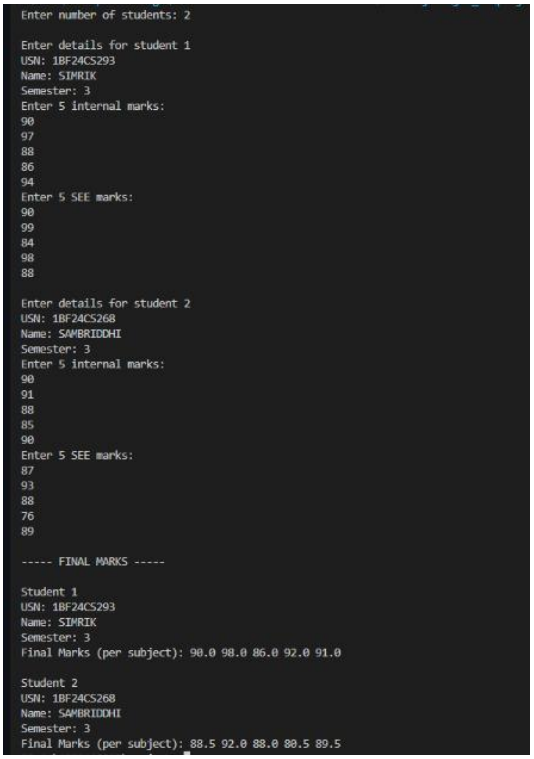
```
        System.out.print("Enter the withdrawal amount: ");
        double amount = s.nextDouble();
        if (amount > balance) {
            System.out.println("Insufficient balance!");
        } else {
            balance -= amount;
            checkMinBalance();
        }
    }
    void checkMinBalance()
    {
        if (balance < minBalance) {
            balance -= serviceCharge;
            System.out.println("Balance below minimum! Service charge of Rs." +
serviceCharge + " imposed.");
        }
    }
}
public class mainbank {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        Sav_acct sav = new Sav_acct();
        Cur_acct cur = new Cur_acct();
        System.out.print("Enter customer name for savings account: ");
        sav.customerName = s.next();
        System.out.print("Enter account Number: ");
        sav.accountNumber = s.nextInt();
        sav.accountType = "saving";
        System.out.print("Enter customer name for current account: ");
        cur.customerName = s.next();
        System.out.print("Enter account Number: ");
        cur.accountNumber = s.nextInt();
        cur.accountType = "current";
        int choice;
        do {
```

```
System.out.println("\n-----MENU-----");
System.out.println("1. Deposit");
System.out.println("2. Withdraw");
System.out.println("3. Compute interest for SavingsAccount");
System.out.println("4. Display account details");
System.out.println("5. Exit");
System.out.print("Enter your choice: ");
choice = s.nextInt();
switch (choice) {
   case 1:
      System.out.print("Enter the type of account: ");
      String type = s.next();
      if (type.equalsIgnoreCase("saving"))
         sav.deposit();
      else
         cur.deposit();
      break;
   case 2:
      System.out.print("Enter the type of account: ");
      type = s.next();
      if (type.equalsIgnoreCase("saving"))
         sav.withdraw();
      else
         cur.withdraw();
      break;
   case 3:
      sav.computeInterest();
      break;
   case 4:
      System.out.print("Enter the type of account: ");
      type = s.next();
      if (type.equalsIgnoreCase("saving"))
         sav.display();
      else
         cur.display();
```

```
                break;
            case 5:
                System.out.println("Exiting...");
                break;
            default:
                System.out.println("Invalid choice!");
        }
    } while (choice != 5);
  }
}
```

Program 6

Create a package CIE which has two classes - Personal 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 Personal. 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.



Code:

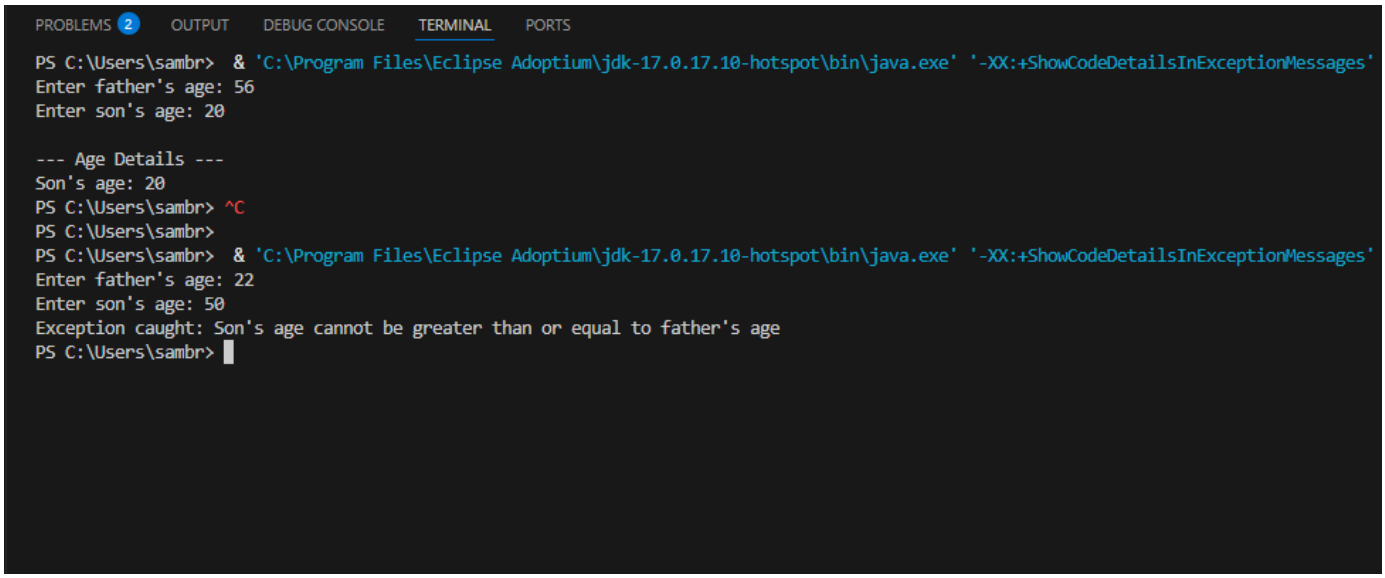
```
import cie.*;
import see.*;
import java.util.*;

public class FinalMarks {
```

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter number of students: ");
    int n = sc.nextInt();

    External[] students = new External[n];
    Internals[] internals = new Internals[n];

    for (int i = 0; i < n; i++) {
        System.out.println("\nEnter details for student " + (i + 1));
        System.out.print("USN: ");
        String usn = sc.next();
        System.out.print("Name: ");
        String name = sc.next();
        System.out.print("Semester: ");
        int sem = sc.nextInt();

        students[i] = new External(usn, name, sem);
        internals[i] = new Internals();

        System.out.println("Enter 5 internal marks:");
        for (int j = 0; j < 5; j++)
            internals[i].internalMarks[j] = sc.nextInt();

        System.out.println("Enter 5 SEE marks:");
        for (int j = 0; j < 5; j++)
            students[i].seeMarks[j] = sc.nextInt();
    }

    System.out.println("\n----- FINAL MARKS -----");
    for (int i = 0; i < n; i++) {
        System.out.println("\nStudent " + (i + 1));
        System.out.println("USN: " + students[i].usn);
        System.out.println("Name: " + students[i].name);
        System.out.println("Semester: " + students[i].sem);
```

```
            System.out.print("Final Marks (per subject): ");
            for (int j = 0; j < 5; j++) {
                double finalMark = (internals[i].internalMarks[j] / 2.0)
                        + (students[i].seeMarks[j] / 2.0);
                System.out.print(finalMark + " ");
            }
            System.out.println();
        }
        sc.close();
    }
}
```

Program 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=father’s age.



Code:

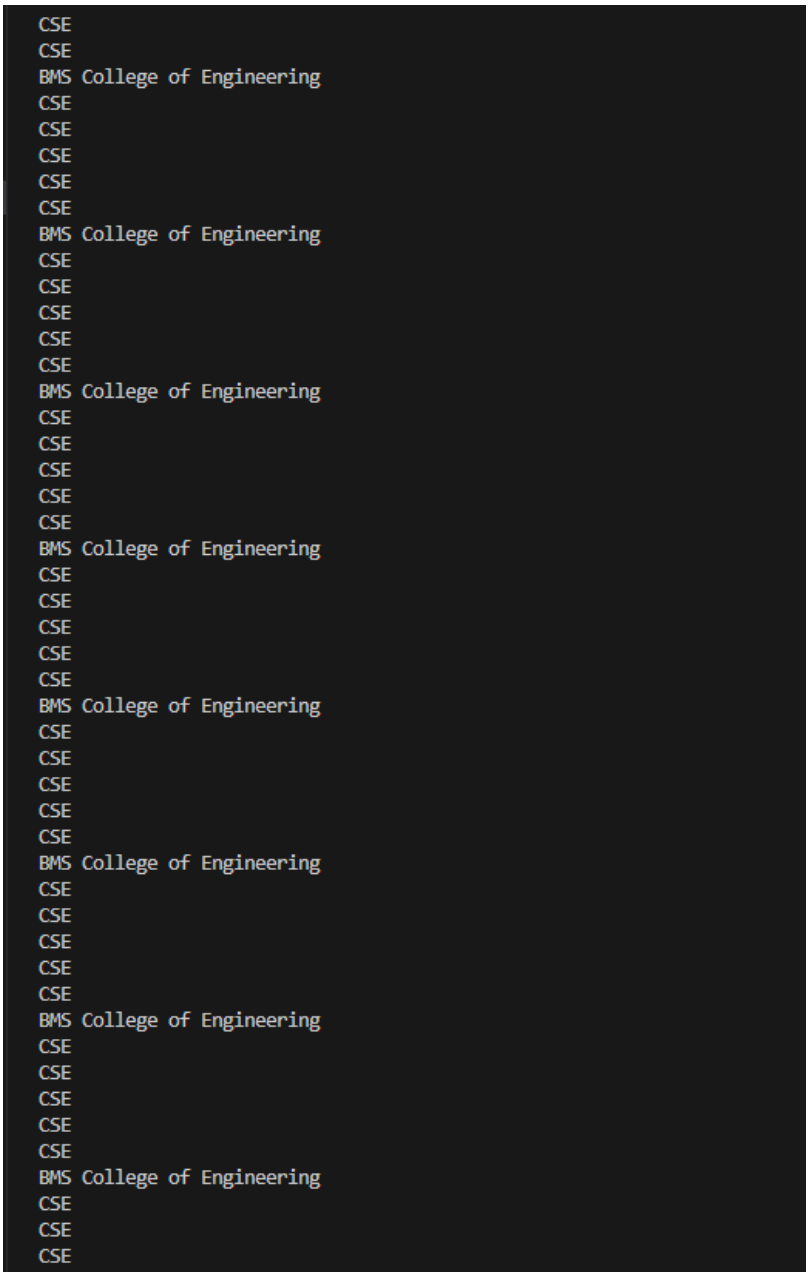
```
import java.util.*;
class WrongAge extends Exception {
   WrongAge() {
      super("Age Error!");
   }
   WrongAge(String msg) {
      super(msg);
   }
}
class InputScanner {
   Scanner s = new Scanner(System.in);
}
```

```
class Father extends InputScanner {
   int fatherAge;
   Father() throws WrongAge {
      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();
      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 && fatherAge < 0)
         throw new WrongAge("Age cannot be negative");
   }
   void display() {
      System.out.println("Son's age: " + sonAge);
   }
}
public class MainAge {
   public static void main(String[] args) {
      try {
         Son obj = new Son();
         System.out.println("\n--- Age Details ---");
```

```
            obj.display();
        } catch (WrongAge e) {
            System.out.println("Exception caught: " + e.getMessage());
        }
    }
}
```

Program 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.



Code:

```
class CollegeThread extends Thread {
    public void run() {
        try {
```

```
            while (true) {
                System.out.println("BMS College of Engineering");
                Thread.sleep(10000);
            }
        } catch (InterruptedException e) {
            System.out.println("Thread Interrupted");
        }
    }
}

class CSEThread extends Thread {
    public void run() {
        try {
            while (true) {
                System.out.println("CSE");
                Thread.sleep(2000);
            }
        } catch (InterruptedException e) {
            System.out.println("Thread Interrupted");
        }
    }
}

public class ThreadDisplay {
    public static void main(String[] args) {
        CollegeThread t1 = new CollegeThread();
        CSEThread t2 = new CSEThread();

        t1.start();
        t2.start();
    }
}
```