VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

CHETHAN REDDY G V (1BM25CS404235)

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 **Chethan Reddy G V (1BM25CS404235),** 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

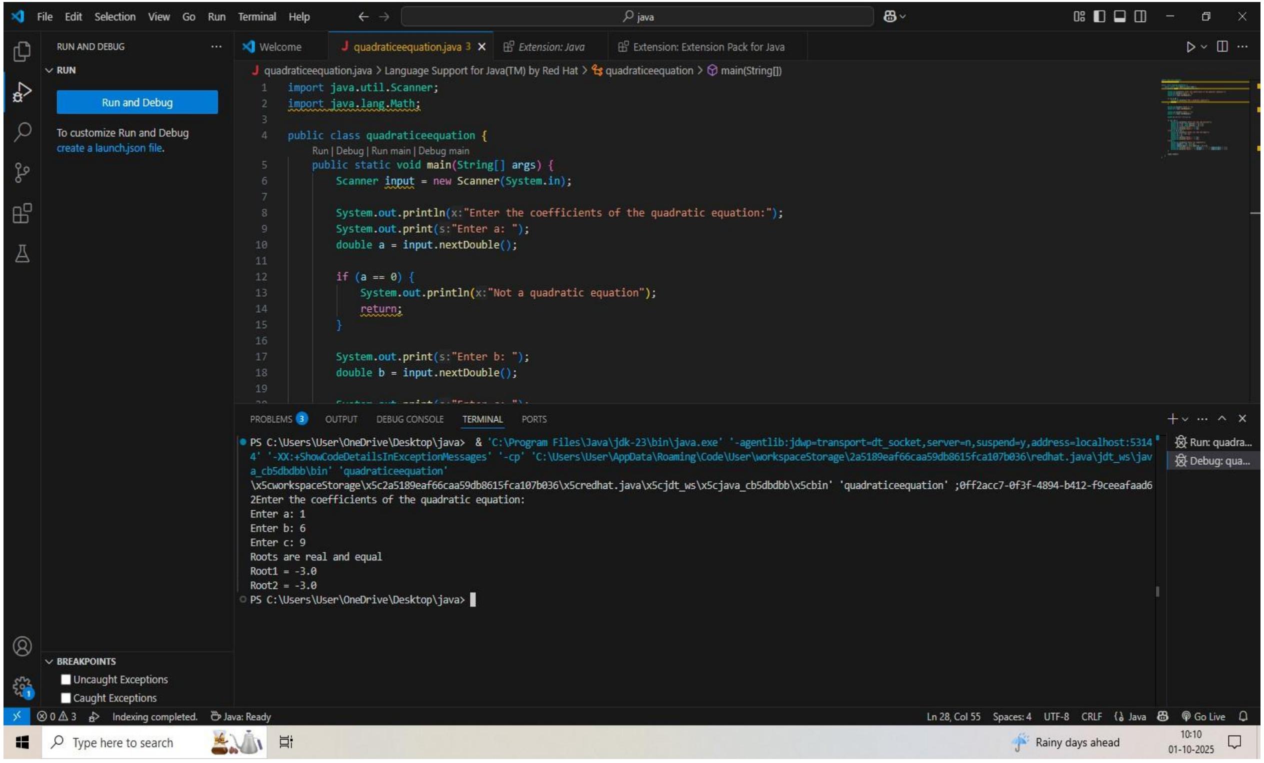
Index

Github Link:
https://github.com/chethan558/Java-repo.git

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.

Output:



File Edit Selection View Go Run Terminal Help
java
RUN AND DEBUG
RUN
Run and Debug
To customize Run and Debug create a launch.json file.
Welcome
quadraticeequation.java 3
Extension: Java
Extension: Extension Pack for Java
quadraticeequation.java > Language Support for Java(TM) by Red Hat > quadraticeequation > main(String[])
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Run: quadra...
Debug: qua...
BREAKPOINTS
Uncaught Exceptions
Caught Exceptions
0 3
Indexing completed.
Java: Ready
Ln 28, Col 55 Spaces: 4 UTF-8 CRLF Java Go Live
Type here to search
Rainy days ahead
10:10
01-10-2025

```java
import java.util.Scanner;
import java.lang.Math;

public class quadraticeequation {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        System.out.println(x:"Enter the coefficients of the quadratic equation:");
        System.out.print(s:"Enter a: ");
        double a = input.nextDouble();

        if (a == 0) {
            System.out.println(x:"Not a quadratic equation");
            return;
        }

        System.out.print(s:"Enter b: ");
        double b = input.nextDouble();

```

```
PS C:\Users\User\OneDrive\Desktop\java>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:5314
4' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\User\AppData\Roaming\Code\User\workspaceStorage\2a5189eaf66caa59db8615fca107b036\redhat.java\jdt_ws\jav
a_cb5dbdbb\bin' 'quadraticeequation'
\x5cworkspaceStorage\x5c2a5189eaf66caa59db8615fca107b036\x5credhat.java\x5cjdt_ws\x5cjava_cb5dbdbb\x5cbin' 'quadraticeequation' ;0ff2acc7-0f3f-4894-b412-f9ceeafaad6
2Enter the coefficients of the quadratic equation:
Enter a: 1
Enter b: 6
Enter c: 9
Roots are real and equal
Root1 = -3.0
Root2 = -3.0
PS C:\Users\User\OneDrive\Desktop\java> 
```

CODE:

```
import java.util.Scanner;
public class QuadraticEquation
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter coefficient a: ");
        double a=sc.nextDouble();
        System.out.println("Enter coefficient b: ");
        double b=sc.nextDouble();
        System.out.println("Enter coefficient c: ");
        double c=sc.nextDouble();
        if (a==0)
        {
           System.out.println("Not a quadratic equation");
           do {
            System.out.println("Enter a non zero value for coefficient a: ");
            a=sc.nextDouble();
```

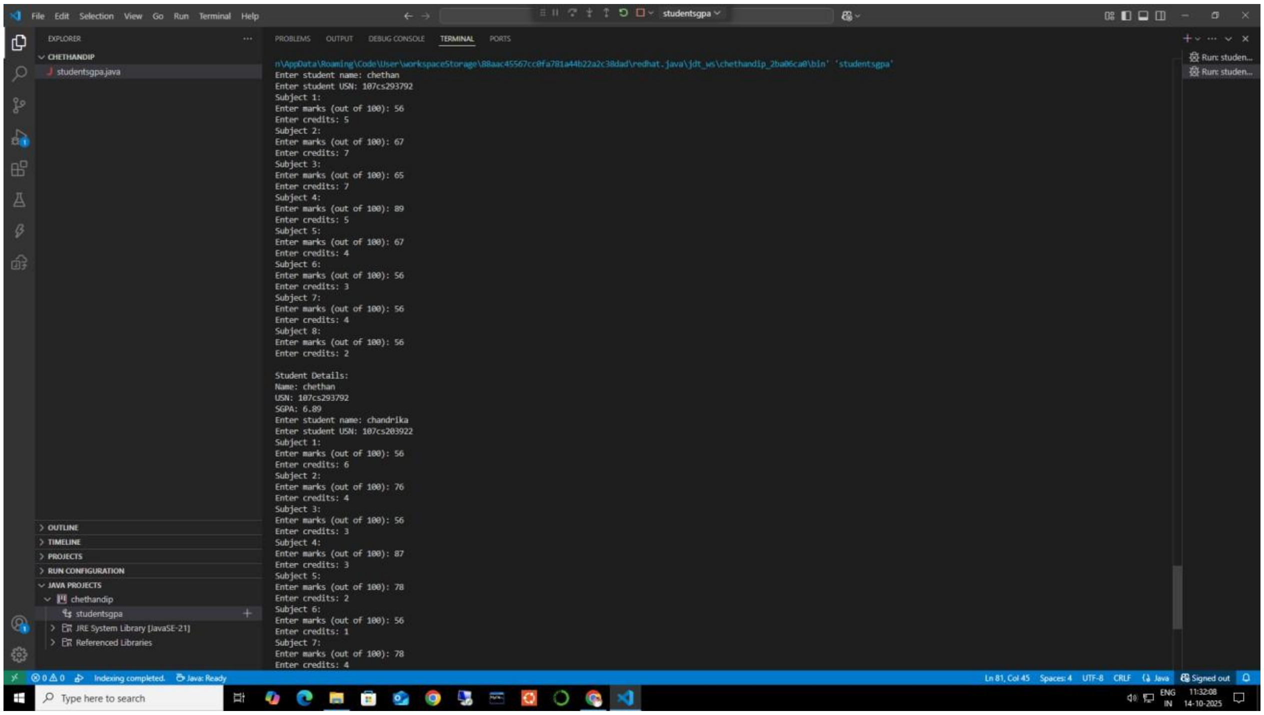
```
   }
   while(a==0);
}

double d=b*b-4*a*c;
if(d>0)
{
    double r1=((-b) + (Math.sqrt(d)))/(2*a);
    double r2=((-b) - (Math.sqrt(d)))/(2*a);
    System.out.println("Roots are real and distinct");
    System.out.println("Root 1: "+r1);
    System.out.println("Root 2: "+r2);
}
else if (d==0)
{
    double r=-b/(2*a);
    System.out.println("Roots are real and equal");
    System.out.println("Root 1 and root 2: "+r);
}
else
{
    System.out.println("Roots are imaginary.No real solution");
```

```
        }
        sc.close();
    }
}
```

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.

Output:



```
n\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06ca0\bin' 'studentsgpa'
Enter student name: chethan
Enter student USN: 107cs293792
Subject 1:
Enter marks (out of 100): 56
Enter credits: 5
Subject 2:
Enter marks (out of 100): 67
Enter credits: 7
Subject 3:
Enter marks (out of 100): 65
Enter credits: 7
Subject 4:
Enter marks (out of 100): 89
Enter credits: 5
Subject 5:
Enter marks (out of 100): 67
Enter credits: 4
Subject 6:
Enter marks (out of 100): 56
Enter credits: 3
Subject 7:
Enter marks (out of 100): 56
Enter credits: 4
Subject 8:
Enter marks (out of 100): 56
Enter credits: 2

Student Details:
Name: chethan
USN: 107cs293792
SGPA: 6.89
Enter student name: chandrika
Enter student USN: 107cs203922
Subject 1:
Enter marks (out of 100): 56
Enter credits: 6
Subject 2:
Enter marks (out of 100): 76
Enter credits: 4
Subject 3:
Enter marks (out of 100): 56
Enter credits: 3
Subject 4:
Enter marks (out of 100): 87
Enter credits: 3
Subject 5:
Enter marks (out of 100): 78
Enter credits: 2
Subject 6:
Enter marks (out of 100): 56
Enter credits: 1
Subject 7:
Enter marks (out of 100): 78
Enter credits: 4
```

CODE:

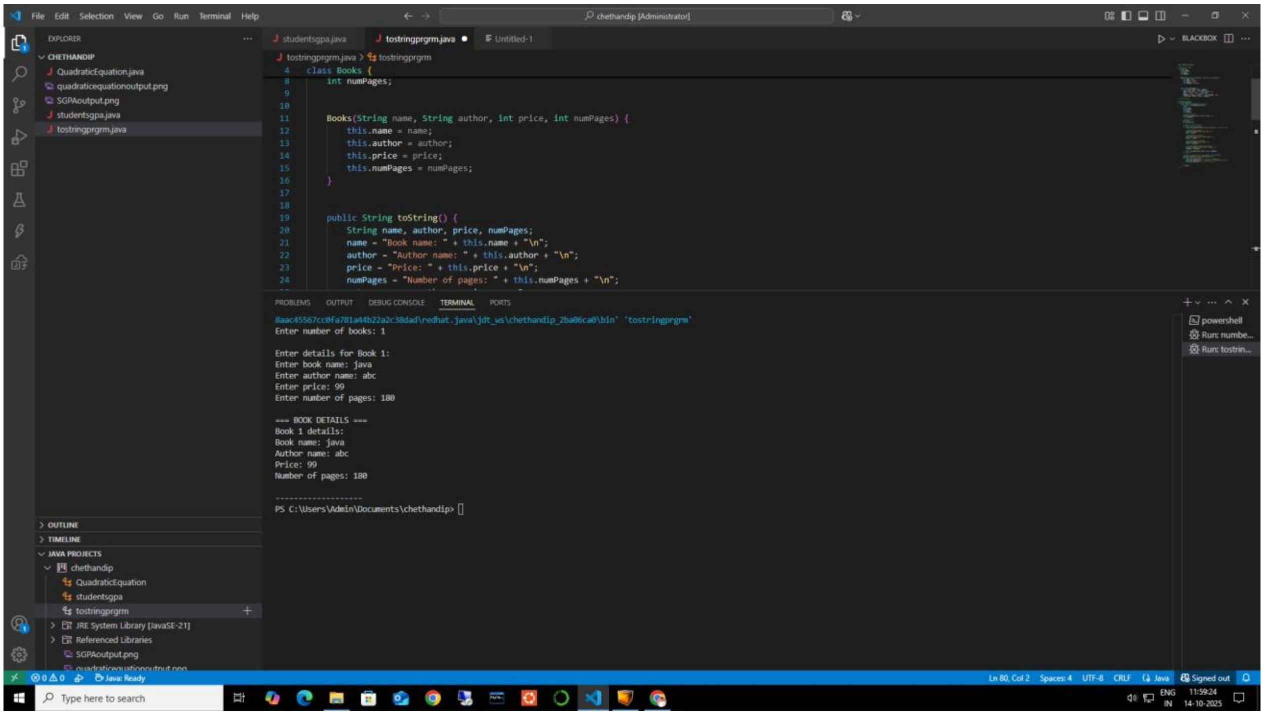
```
import java.util.Scanner;

class Subject {

    int subjectMarks;

    double credits;

    int grade;

}

class Student {

    String name, usn;

    double SGPA;

    Scanner s;

    Subject[] subject;


    Student() {

        s = new Scanner(System.in);

        subject = new Subject[8];

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

            subject[i] = new Subject();

        }

    }

    void getStudentDetails() {

        System.out.print("Enter student name: ");
        name = s.nextLine();
        System.out.print("Enter student USN: ");
        usn = s.nextLine();
    }
    void getMarks() {
        for(int i = 0; i < 8; i++) {
            System.out.println("Subject " + (i+1) + ":");
            System.out.print("Enter marks (out of 100): ");
            subject[i].subjectMarks = s.nextInt();

            if(subject[i].subjectMarks > 100 || subject[i].subjectMarks < 0) {
                System.out.println("Invalid marks! Please enter marks between 0 and
100.");
                i--;
                continue;
            }
            System.out.print("Enter credits: ");
            subject[i].credits = s.nextDouble();
            subject[i].grade = (subject[i].subjectMarks / 10) + 1;
            if(subject[i].grade == 11) {
                subject[i].grade = 10;
            }
            }
            if(subject[i].subjectMarks < 40) {
                subject[i].grade = 0;
            }
        }
    }

    void computeSGPA() {
        int effectiveScore = 0;
        int totalCredits = 0;
        for(int i = 0; i < 8; i++) {
            effectiveScore += (subject[i].grade * subject[i].credits);
            totalCredits += subject[i].credits;
        }
        SGPA = (double) effectiveScore / (double) totalCredits;
    }
    void display() {
        System.out.println("\nStudent Details:");
        System.out.println("Name: " + name);
        System.out.println("USN: " + usn);
        System.out.printf("SGPA: %.2f\n", SGPA);
    }
}
```

```
public class studentsgpa {
    public static void main(String[] args) {
        Student student1 = new Student();
        student1.getStudentDetails();
        student1.getMarks();
        student1.computeSGPA();
        student1.display();
        Student student2=new Student();
        student2.getStudentDetails();
        student2.getMarks();
        student2.computeSGPA();
        student2.display();
    }
}
```

Program 3

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

Output:



```java
class Books {
    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";
```

```
8aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06ca0\bin' 'tostringprgrm'
Enter number of books: 1

Enter details for Book 1:
Enter book name: java
Enter author name: abc
Enter price: 99
Enter number of pages: 180

=== BOOK DETAILS ===
Book 1 details:
Book name: java
Author name: abc
Price: 99
Number of pages: 180

-------------------
PS C:\Users\Admin\Documents\chethandip> 
```

CODE:

```java
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;
    }
}

class tostringprgrm {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        int n;
```

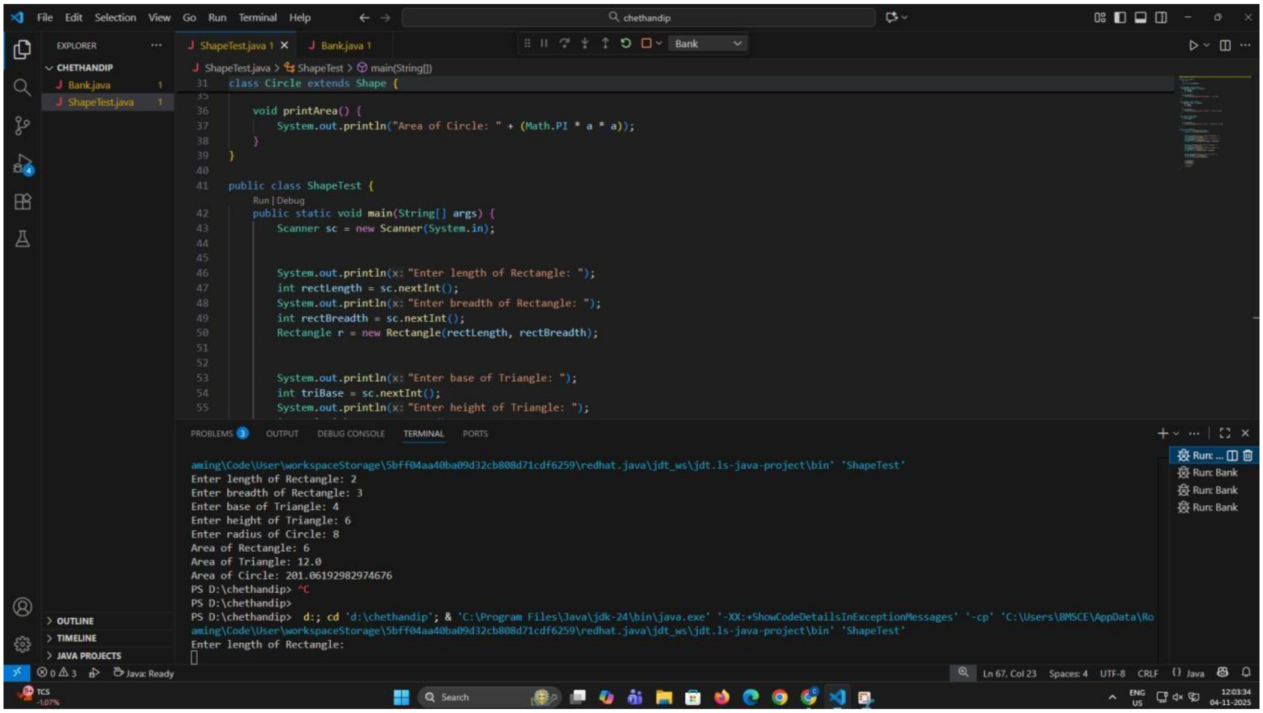
```
String name;
String author;
int price;
int numPages;
System.out.print("Enter number of books: ");
n = s.nextInt();
Books b[];
b = new Books[n];
for (int i = 0; i < n; i++) {
    System.out.println("\nEnter details for Book " + (i + 1) + ":");
    System.out.print("Enter book name: ");
    name = s.next();
    System.out.print("Enter author name: ");
    author = s.next();
    System.out.print("Enter price: ");
    price = s.nextInt();
    System.out.print("Enter number of pages: ");
    numPages = s.nextInt();

    b[i] = new Books(name, author, price, numPages);
}
System.out.println("\n=== BOOK DETAILS ===");
for (int i = 0; i < n; i++) {
    System.out.println("Book " + (i + 1) + " details:");
    System.out.println(b[i]);
```

```
            System.out.println(" ------------------ ");
        }
        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. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

Output:



```
class Circle extends Shape {

    void printArea() {
        System.out.println("Area of Circle: " + (Math.PI * a * a));
    }
}

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


        System.out.println(x: "Enter length of Rectangle: ");
        int rectLength = sc.nextInt();
        System.out.println(x: "Enter breadth of Rectangle: ");
        int rectBreadth = sc.nextInt();
        Rectangle r = new Rectangle(rectLength, rectBreadth);


        System.out.println(x: "Enter base of Triangle: ");
        int triBase = sc.nextInt();
        System.out.println(x: "Enter height of Triangle: ");
```

```
aming\Code\User\workspaceStorage\5bff04aa40ba09d32cb808d71cdf6259\redhat.java\jdt_ws\jdt.ls-java-project\bin' 'ShapeTest'
Enter length of Rectangle: 2
Enter breadth of Rectangle: 3
Enter base of Triangle: 4
Enter height of Triangle: 6
Enter radius of Circle: 8
Area of Rectangle: 6
Area of Triangle: 12.0
Area of Circle: 201.06192982974676
PS D:\chethandip> ^C
PS D:\chethandip>
PS D:\chethandip>  d:; cd 'd:\chethandip'; & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\BMSCE\AppData\Ro
aming\Code\User\workspaceStorage\5bff04aa40ba09d32cb808d71cdf6259\redhat.java\jdt_ws\jdt.ls-java-project\bin' 'ShapeTest'
Enter length of Rectangle:
```

CODE:

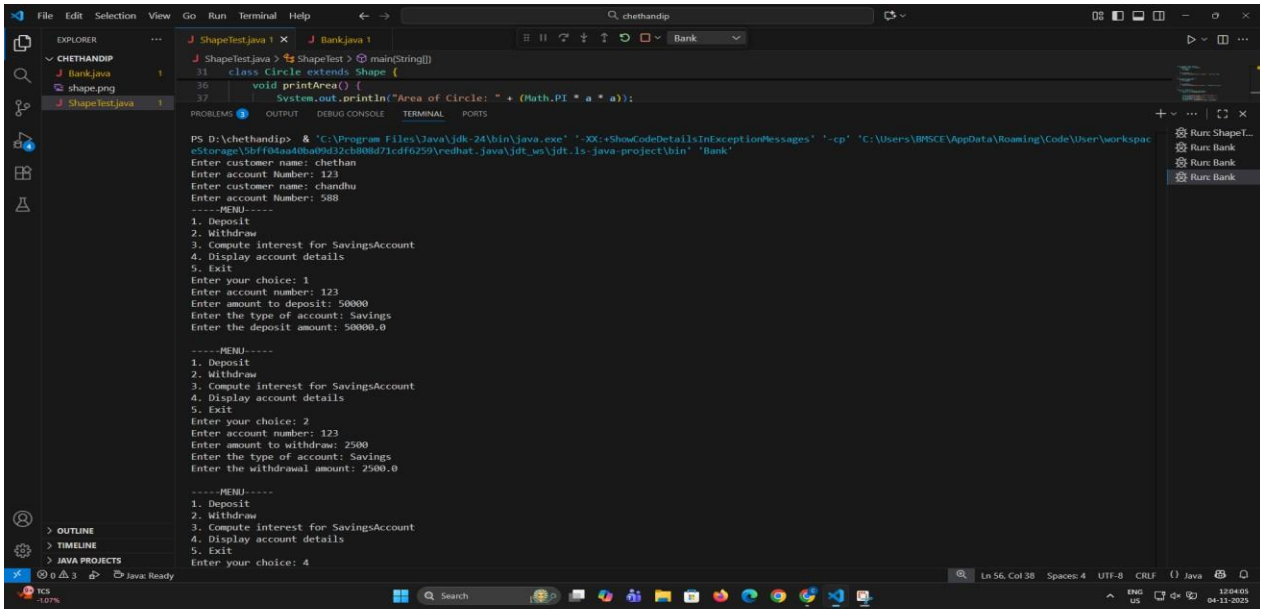
```
import java.util.Scanner;
```

```
abstract class Shape {
    int a, b;

    abstract void printArea();
}

class Rectangle extends Shape {
    Rectangle(int length, int breadth) {
        a = length;
        b = breadth;
    }

    void printArea() {
        System.out.println("Area of Rectangle: " + (a * b));
    }
}

class Triangle extends Shape {
    Triangle(int base, int height) {
        a = base;
        b = height;
    }
    void printArea() {
        System.out.println("Area of Triangle: " + (0.5 * a * b));
    }
}

class Circle extends Shape {
    Circle(int radius) {
        a = radius;
    }

    void printArea() {
        System.out.println("Area of Circle: " + (Math.PI * a * a));
    }
}

public class ShapeTest {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter length of Rectangle: ");
        int rectLength = sc.nextInt();
        System.out.println("Enter breadth of Rectangle: ");
        int rectBreadth = sc.nextInt();
        Rectangle r = new Rectangle(rectLength, rectBreadth);
        System.out.println("Enter base of Triangle: ");
        int triBase = sc.nextInt();
        System.out.println("Enter height of Triangle: ");
        int triHeight = sc.nextInt();
        Triangle t = new Triangle(triBase, triHeight);
        System.out.println("Enter radius of Circle: ");
        int circleRadius = sc.nextInt();
        Circle c = new Circle(circleRadius);
        r.printArea();
        t.printArea();
        c.printArea();

        sc.close();
    }
}
```

Program 5

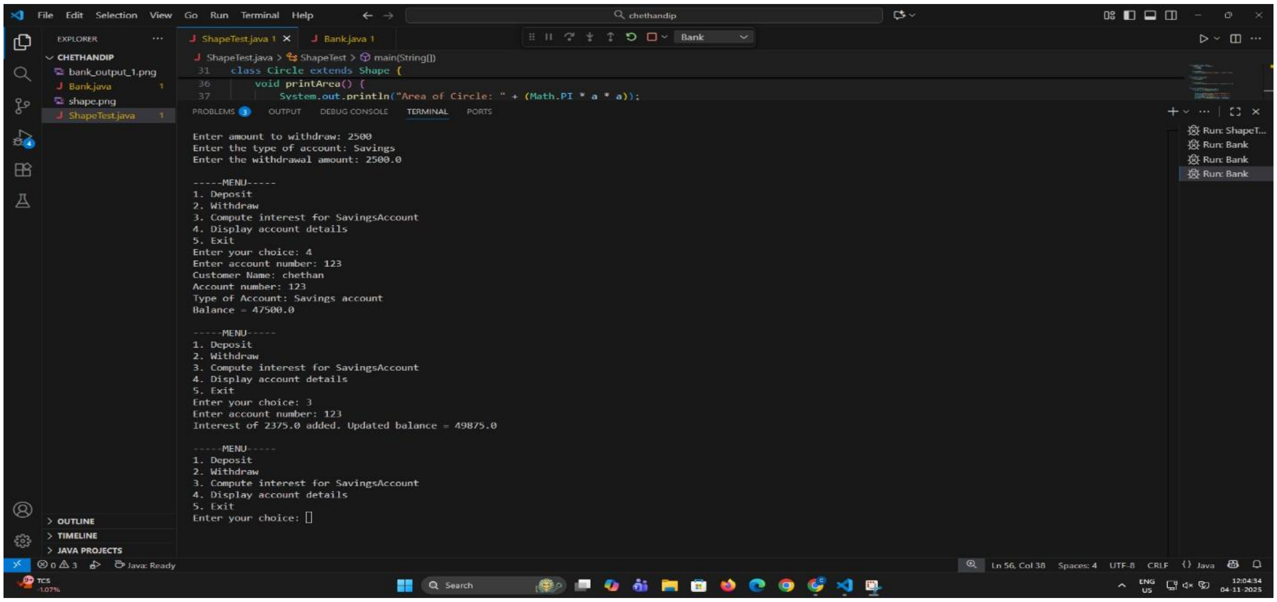
Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book 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.

Output:



```java
class Circle extends Shape {
    void printArea() {
        System.out.println("Area of Circle: " + (Math.PI * a * a));
```

```
PS D:\chethandip>  & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\BMSCE\AppData\Roaming\Code\User\workspac
eStorage\5bff04aa40ba09d32cb808d71cdf6259\redhat.java\jdt_ws\jdt.ls-java-project\bin' 'Bank'
Enter customer name: chethan
Enter account Number: 123
Enter customer name: chandhu
Enter account Number: 588
-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 1
Enter account number: 123
Enter amount to deposit: 50000
Enter the type of account: Savings
Enter the deposit amount: 50000.0

-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 2
Enter account number: 123
Enter amount to withdraw: 2500
Enter the type of account: Savings
Enter the withdrawal amount: 2500.0

-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 4
```



```
class Circle extends Shape {
    void printArea() {
        System.out.println("Area of Circle: " + (Math.PI * a * a));
```

```
Enter amount to withdraw: 2500
Enter the type of account: Savings
Enter the withdrawal amount: 2500.0

-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 4
Enter account number: 123
Customer Name: chethan
Account number: 123
Type of Account: Savings account
Balance = 47500.0

-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 3
Enter account number: 123
Interest of 2375.0 added. Updated balance = 49875.0

-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 
```

CODE:

```
import java.util.Scanner;

class Account {

    String customerName;

    int accountNumber;

    String accountType;

    double balance;

Account(String name, int number, String type) {

        customerName = name;

        accountNumber = number;

        accountType = type;

        balance = 0.0;

    }
```

```
void deposit(double amount) {
    balance += amount;
    System.out.println("Enter the type of account: " + accountType);
    System.out.println("Enter the deposit amount: " + amount);
    System.out.println();
}

void withdraw(double amount) {
    if (balance >= amount) {
        balance -= amount;
        System.out.println("Enter the type of account: " + accountType);
        System.out.println("Enter the withdrawal amount: " + amount);
        System.out.println();
    } else {
        System.out.println("Insufficient balance!");
    }
}

void displayAccount() {
    System.out.println("Customer Name: " + customerName);
    System.out.println("Account number: " + accountNumber);
    System.out.println("Type of Account: " + accountType + " account");
```

```
        System.out.println("Balance = " + balance);
        System.out.println();
    }
}


class Sav_acct extends Account {
    Sav_acct(String name, int number, String type) {
        super(name, number, type);
    }

    void computeInterest() {
        double rate = 0.05; // 5%
        double interest = balance * rate;
        balance += interest;
        System.out.println("Interest of " + interest + " added. Updated balance = " +
balance);
        System.out.println();
    }
}
```

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

        System.out.print("Enter customer name: ");
        String name1 = sc.nextLine();
        System.out.print("Enter account Number: ");
        int acc1 = sc.nextInt();
        sc.nextLine();

        System.out.print("Enter customer name: ");
        String name2 = sc.nextLine();
        System.out.print("Enter account Number: ");
        int acc2 = sc.nextInt();
        sc.nextLine();

        Sav_acct s1 = new Sav_acct(name1, acc1, "Savings");
        Sav_acct s2 = new Sav_acct(name2, acc2, "Savings");

        int choice;
        do {
            System.out.println("-----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 = sc.nextInt();

            switch (choice) {
                case 1:
                    System.out.print("Enter account number: ");
                    int ac = sc.nextInt();
                    System.out.print("Enter amount to deposit: ");
                    double damt = sc.nextDouble();
                    if (ac == s1.accountNumber)
                        s1.deposit(damt);
                    else if (ac == s2.accountNumber)
                        s2.deposit(damt);
                    else
                        System.out.println("Invalid account number!");
                    break;
```

```
case 2:
    System.out.print("Enter account number: ");
    ac = sc.nextInt();
    System.out.print("Enter amount to withdraw: ");
    double wamt = sc.nextDouble();
    if (ac == s1.accountNumber)
        s1.withdraw(wamt);
    else if (ac == s2.accountNumber)
        s2.withdraw(wamt);
    else
        System.out.println("Invalid account number!");
    break;

case 3:
    System.out.print("Enter account number: ");
    ac = sc.nextInt();
    if (ac == s1.accountNumber)
        s1.computeInterest();
    else if (ac == s2.accountNumber)
        s2.computeInterest();
    else
        System.out.println("Invalid account number!");
```

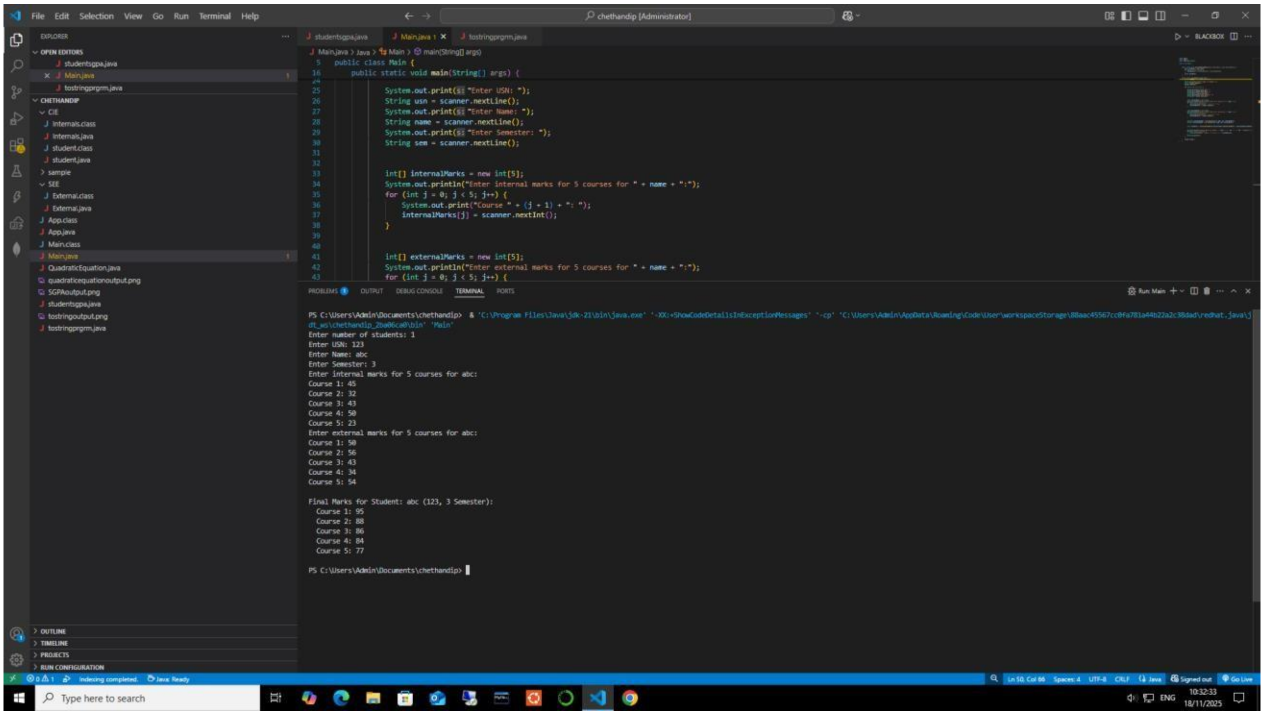
```
            break;

        case 4:
            System.out.print("Enter account number: ");
            ac = sc.nextInt();
            if (ac == s1.accountNumber)
                s1.displayAccount();
            else if (ac == s2.accountNumber)
                s2.displayAccount();
            else
                System.out.println("Invalid account number!");
            break;

        case 5:
            System.out.println("Exiting...");
            break;

        default:
            System.out.println("Invalid  choice!");
    }

} while (choice != 5);
```

```
        sc.close();
    }
}
```

Program 6

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

Output:



```
public class Main {
    public static void main(String[] args) {
            System.out.print(s: "Enter USN: ");
            String usn = scanner.nextLine();
            System.out.print(s: "Enter Name: ");
            String name = scanner.nextLine();
            System.out.print(s: "Enter Semester: ");
            String sem = scanner.nextLine();


            int[] internalMarks = new int[5];
            System.out.println("Enter internal marks for 5 courses for " + name + ":");
            for (int j = 0; j < 5; j++) {
                System.out.print("Course " + (j + 1) + ": ");
                internalMarks[j] = scanner.nextInt();
            }


            int[] externalMarks = new int[5];
            System.out.println("Enter external marks for 5 courses for " + name + ":");
            for (int j = 0; j < 5; j++) {
```

```
PS C:\Users\Admin\Documents\chethandip>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\j
dt_ws\chethandip_2ba06ca0\bin' 'Main'
Enter number of students: 1
Enter USN: 123
Enter Name: abc
Enter Semester: 3
Enter internal marks for 5 courses for abc:
Course 1: 45
Course 2: 32
Course 3: 43
Course 4: 50
Course 5: 23
Enter external marks for 5 courses for abc:
Course 1: 50
Course 2: 56
Course 3: 43
Course 4: 34
Course 5: 54

Final Marks for Student: abc (123, 3 Semester):
  Course 1: 95
  Course 2: 88
  Course 3: 86
  Course 4: 84
  Course 5: 77

PS C:\Users\Admin\Documents\chethandip> 
```

CODE:

- **Project**
    - CIE
        - internals.java
        - student.java
    - SEE
        - externals.java

- Main.java

CIE Folder

Student.java

```
package CIE;


public class student {
    protected String usn;
    protected String name;
    protected String sem;


    public student(String usn, String name, String sem) {
        this.usn = usn;
        this.name = name;
        this.sem = sem;
    }


    public String getUsn() {
        return usn;
    }


    public String getName() {
        return name;
    }

    public String getSem() {
        return sem;
    }
}
```

Internals.java

```
package CIE;

public class Internals extends student {
    private int[] internalMarks = new int[5];  // Marks for 5 courses

    public Internals(String usn, String name, String sem, int[] internalMarks) {
        super(usn, name, sem);
        this.internalMarks = internalMarks;
    }

    public int[] getInternalMarks() {
        return internalMarks;
    }
}
```

SEE Folder

External.java

```
package SEE;


import CIE.student;


public class External extends student {
    private int[] externalMarks = new int[5];


    public External(String usn, String name, String sem, int[] externalMarks) {
        super(usn, name, sem);
        this.externalMarks = externalMarks;
    }


    public int[] getExternalMarks() {
        return externalMarks;
    }
}
```

Main.java

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

public class Main {



    public static int[] calculateFinalMarks(int[] internalMarks, int[] externalMarks) {
        int[] finalMarks = new int[5];
        for (int i = 0; i < 5; i++) {
            finalMarks[i] = internalMarks[i] + externalMarks[i];
        }
        return finalMarks;
    }

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

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

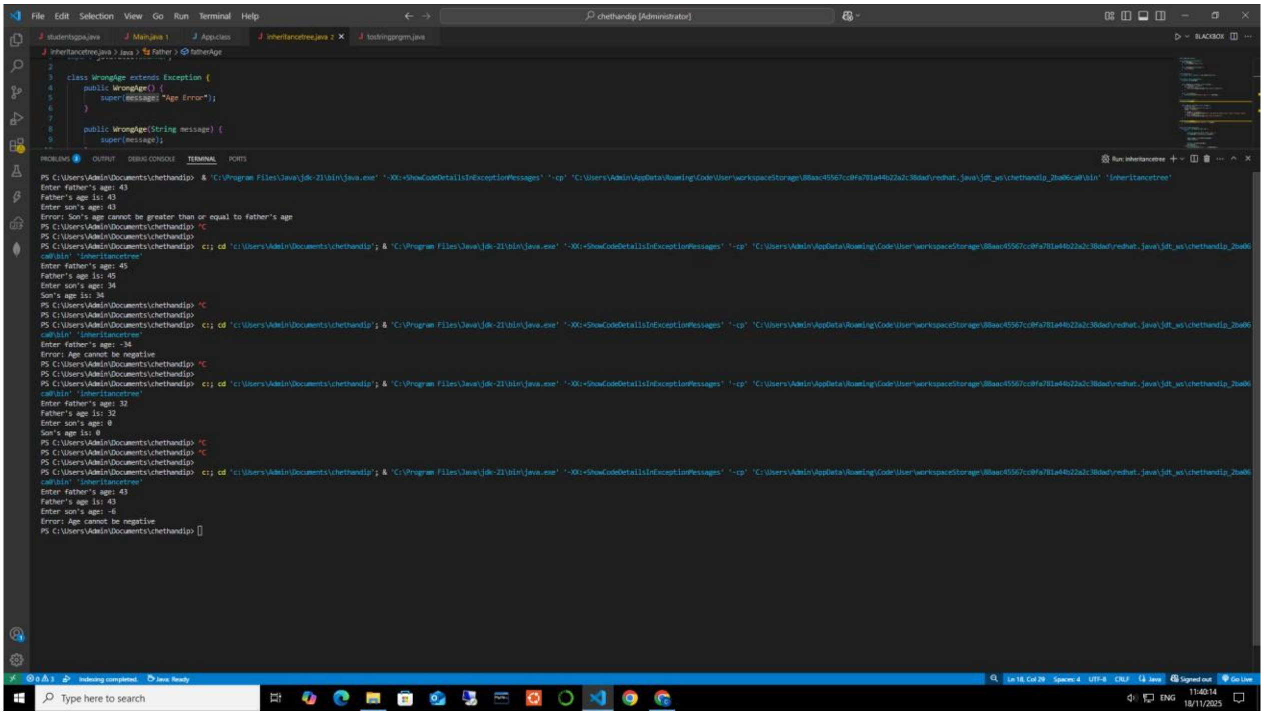
```
for (int i = 0; i < n; i++) {

    System.out.print("Enter USN: ");
    String usn = scanner.nextLine();
    System.out.print("Enter Name: ");
    String name = scanner.nextLine();
    System.out.print("Enter Semester: ");
    String sem = scanner.nextLine();


    int[] internalMarks = new int[5];
    System.out.println("Enter internal marks for 5 courses for " + name + ":");
    for (int j = 0; j < 5; j++) {
        System.out.print("Course " + (j + 1) + ": ");
        internalMarks[j] = scanner.nextInt();
    }


    int[] externalMarks = new int[5];
    System.out.println("Enter external marks for 5 courses for " + name + ":");
    for (int j = 0; j < 5; j++) {
        System.out.print("Course " + (j + 1) + ": ");
```

```
                externalMarks[j] = scanner.nextInt();
            }



            Internals internalStudent = new Internals(usn, name, sem, internalMarks);
            External externalStudent = new External(usn, name, sem, externalMarks);



            int[] finalMarks = calculateFinalMarks(internalStudent.getInternalMarks(), externalStudent.getExternalMarks());



            System.out.println("\nFinal Marks for Student: " + name + " (" + usn + ", " + sem + " Semester):");
            for (int j = 0; j < 5; j++) {
                System.out.println("  Course " + (j + 1) + ": " + finalMarks[j]);
            }
            System.out.println();
        }

        scanner.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<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.

Output:



```
class WrongAge extends Exception {
    public WrongAge() {
        super(message: "Age Error");
    }

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

```
PS C:\Users\Admin\Documents\chethandip>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06ca0\bin' 'inheritancetree'
Enter father's age: 43
Father's age is: 43
Enter son's age: 43
Error: Son's age cannot be greater than or equal to father's age
PS C:\Users\Admin\Documents\chethandip> ^C
PS C:\Users\Admin\Documents\chethandip>
PS C:\Users\Admin\Documents\chethandip>  c:; cd 'c:\Users\Admin\Documents\chethandip'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06
ca0\bin' 'inheritancetree'
Enter father's age: 45
Father's age is: 45
Enter son's age: 34
Son's age is: 34
PS C:\Users\Admin\Documents\chethandip> ^C
PS C:\Users\Admin\Documents\chethandip>
PS C:\Users\Admin\Documents\chethandip>  c:; cd 'c:\Users\Admin\Documents\chethandip'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06
ca0\bin' 'inheritancetree'
Enter father's age: -34
Error: Age cannot be negative
PS C:\Users\Admin\Documents\chethandip> ^C
PS C:\Users\Admin\Documents\chethandip>
PS C:\Users\Admin\Documents\chethandip>  c:; cd 'c:\Users\Admin\Documents\chethandip'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06
ca0\bin' 'inheritancetree'
Enter father's age: 32
Father's age is: 32
Enter son's age: 0
Son's age is: 0
PS C:\Users\Admin\Documents\chethandip> ^C
PS C:\Users\Admin\Documents\chethandip> ^C
PS C:\Users\Admin\Documents\chethandip>
PS C:\Users\Admin\Documents\chethandip>  c:; cd 'c:\Users\Admin\Documents\chethandip'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06
ca0\bin' 'inheritancetree'
Enter father's age: 43
Father's age is: 43
Enter son's age: -6
Error: Age cannot be negative
PS C:\Users\Admin\Documents\chethandip> 
```

CODE:

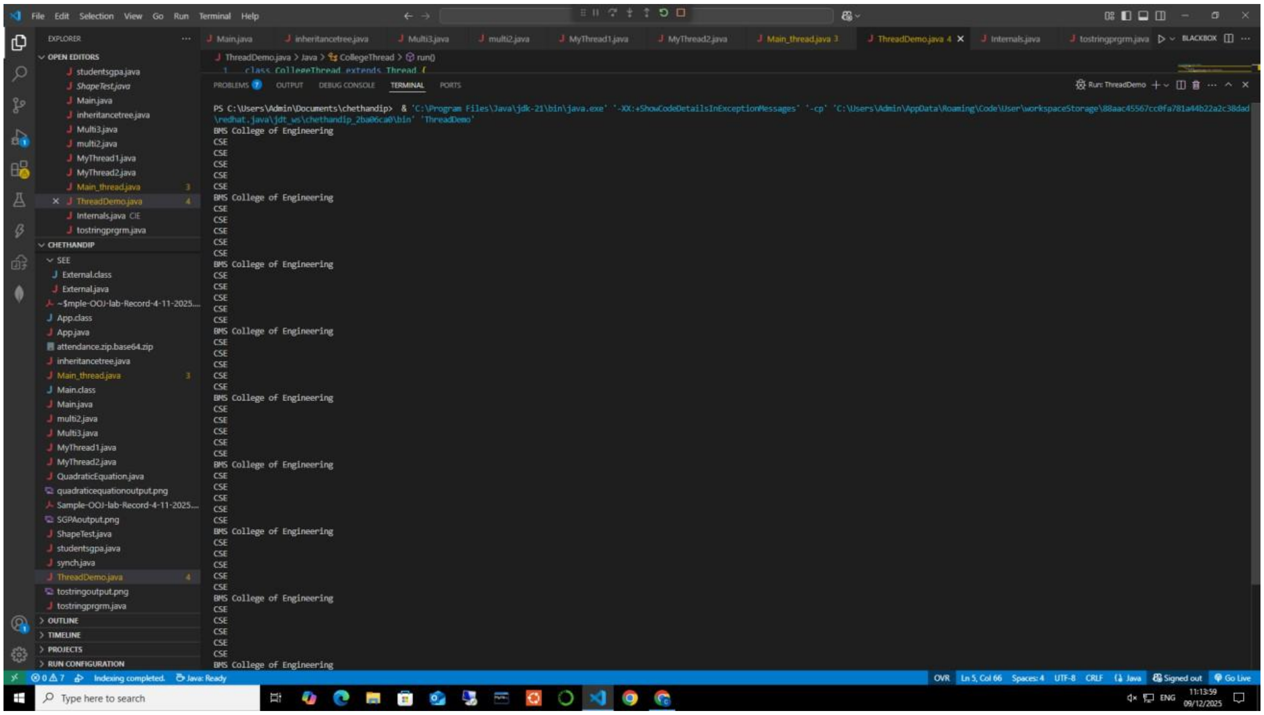
```
import java.util.Scanner;
```

```
class WrongAge extends Exception {
    public WrongAge() {
        super("Age Error");
    }

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

class InputScanner {
    protected static Scanner s = new Scanner(System.in);
}

class Father extends InputScanner {
    protected int fatherAge;


    public Father(int fatherAge) throws WrongAge {
        this.fatherAge = fatherAge;
        if (fatherAge < 0) {
            throw new WrongAge("Age cannot be negative");
        }
    }

    public void display() {
        System.out.println("Father's age is: " + fatherAge);
    }
}

class Son extends Father {
    private int sonAge;



    public Son(int fatherAge) throws WrongAge {
        super(fatherAge);
        System.out.print("Enter son's age: ");
        sonAge = s.nextInt();

        if (sonAge >= fatherAge) {
            throw new WrongAge("Son's age cannot be greater than or equal to father's age");
        } else if (sonAge < 0) {
            throw new WrongAge("Age cannot be negative");
        }
    }

    public void display() {
        System.out.println("Son's age is: " + sonAge);
    }
}

public class inheritancetree {
    public static void main(String[] args) {
        try {

            System.out.print("Enter father's age: ");
            int fatherAge = InputScanner.s.nextInt();


            Father father = new Father(fatherAge);
            father.display();


            Son son = new Son(fatherAge);
            son.display();
        } catch (WrongAge e) {
            System.out.println("Error: " + 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.

Output:



```
PS C:\Users\Admin\Documents\chethandip>  & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Admin\AppData\Roaming\Code\User\workspaceStorage\88aac45567cc0fa781a44b22a2c38dad\redhat.java\jdt_ws\chethandip_2ba06ca0\bin' 'ThreadDemo'
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
```

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

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