

**VISVESVARAYA TECHNOLOGICAL
UNIVERSITY**
“JnanaSangama”, Belgaum -590014, Karnataka.



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

Submitted by

SUDHIR UPADHAYA (1BF24CS304)

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 **SUDHIR UPADHAYA(1BF24CS304)**, 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

Sl. No.	Date	Experiment Title	Page No.
1	23/9/25	Quadratic Equations	4-5
2	13/10/25	SGPA Calculator	6-10
3	14/10/25	Bookstore Program	11-13
4	4/11/25	Shapes Program	14-16
5	4/11/25	Bank Program	17-23
6	18/11/25	Packages	24-26
7	26/11/25	Errors	27-28
8	9/12/25	Multi Threading	29-30
9	9/12/25	Open Ended Question 1	30-32
10	9/12/25	Open Ended Question 2	32-35

Github link: <https://github.com/sudhirupadhyacs24-glitch/SUDHIR-UPADHAYA-USN1BF24CS304-..>

Program 1

Implement Quadratic Equation

Code:

```
import java.util.Scanner;

class quadratic
{
    public static void main(String[] args)
    {
        Scanner s = new Scanner(System.in);
        float a,b,c,d;
        double r1,r2;
        System.out.println("Enter coefficient of a:");
        a= s.nextFloat();
        System.out.println("Enter coefficient of b: ");
        b= s.nextFloat();
        System.out.println("Enter coefficient of c:");
        c= s.nextFloat();
        d= b*b - 4*a*c;
        if (a==0)
        {
            System.out.println("Not a quadratic equation");
        }
        else {

            if (d==0) {
                r1=(-b)/(2*a);
                r2=r1;
                System.out.println("Roots are real and equal:"+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 distinct R1:"+r1+ "and R2:"+r2);
            }
            else {
                System.out.println("Roots are imaginary");
                r1 = (-b)/(2*a);
                r2 = Math.sqrt(-d)/(2*a);
                System.out.println("Root 1:"+r1+"+"+r2+"i");
                System.out.println("Root 2:"+r1+"-"+r2+"i");
            }
        }
        System.out.println("USN:1BF24CS262 Name:Sahil Biswas");
        s.close();
    }
}
```

```
    }  
}
```

Output:

```
Name = Sudhir Upadhyaya  
USN = 1BF24CS304  
enter the value of a:  
enter the value of b:  
3  
enter the value of c:  
2  
Not a quadratic equation  
PS C:\Users\Admin\Desktop\1BF24CS304> cd "c:\Users\Admin\Desktop\1BF24CS304\" ; if ($?) { javac quadratic.java } ; if ($?) { java quadratic }  
Name = Sudhir Upadhyaya  
USN = 1BF24CS304  
enter the value of a:  
1  
enter the value of b:  
enter the value of c:  
3  
Roots are real and distinct.  
the values of r1:-1.0  
the values of r2:-3.0  
PS C:\Users\Admin\Desktop\1BF24CS304> cd "c:\Users\Admin\Desktop\1BF24CS304\" ; if ($?) { javac quadratic.java } ; if ($?) { java quadratic }  
Name = Sudhir Upadhyaya  
USN = 1BF24CS304  
enter the value of a:  
4  
enter the value of b:  
enter the value of c:  
2  
Roots are imaginary.  
the values of r1:0.0  
the values of r1:0.5
```

PROGRAM 2:

Code:

```
import java.util.Scanner;

class Subject {
    int subjectMarks;
    int credits;
    int grade;

    public Subject() {
        this.subjectMarks = 0;
        this.credits = 0;
        this.grade = 0;
    }
}

class Student {
    String name;
    String usn;
    double SGPA;
    Scanner s;
    Subject[] subject;

    public Student() {
        s = new Scanner(System.in);
        subject = new Subject[8];
        for (int i = 0; i < 8; i++) {
            subject[i] = new Subject();
        }
    }

    public void getStudentDetails() {
        System.out.print("Enter student name: ");
        name = s.nextLine();
        System.out.print("Enter student USN: ");
        usn = s.nextLine();
    }

    public void getMarks() {
        for (int i = 0; i < 8; i++) {
            System.out.print("Enter marks for subject " + (i + 1) + ": ");
        }
    }
}
```

```
subject[i].subjectMarks = s.nextInt();
System.out.print("Enter credits for subject " + (i + 1) + ": ");
subject[i].credits = s.nextInt();

subject[i].grade = (subject[i].subjectMarks / 10) + 1;
```

```

if(subject[i].grade == 11) {
    subject[i].grade = 10;
}
if(subject[i].grade <= 4) {
    subject[i].grade = 0;
}
}

public 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 / totalCredits;
}

public void display() {
    System.out.println("\nStudent Details:");
    System.out.println("Name: " + name);
    System.out.println("USN: " + usn);
    System.out.println("SGPA: " + SGPA);
}

}

public class SGPAcalculator {
    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();

        Student[] students = new Student[n];

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

            students[i] = new Student();
            students[i].getStudentDetails();
            students[i].getMarks();
            students[i].computeSGPA();
        }

        System.out.println("\n--- Student Results ---");
    }
}

```

```

        for (int i = 0; i < n; i++) {
            students[i].display();
        }

        scanner.close();
    }
}

```

Output:

```

Sudhir Upadhaya
1BF24CS304
Enter the number of students:
2
Enter USN:1bf24cs304
Enter the name of student:
surajrastogi
Enter details of credits and marks in order for 4 subjects:
Enter credits for subject1:
3
Enter marks for 1:
95
Enter credits for subject2:
4
Enter marks for 2:
96
Enter credits for subject3:
4
Enter marks for 3:
95
Enter credits for subject4:
3
Enter marks for 4:
88
Enter credits for subject5:
3
Enter marks for 5:
87
Enter credits for subject6:
4
Enter marks for 6:
89
Enter credits for subject7:
4
Enter marks for 7:
90
STUDENT DETAILS
USN:1bf24cs304
NAME:surajrastogi
Subject 1- CREDITS:3,MARKS: 95
Subject 2- CREDITS:4,MARKS: 96
Subject 3- CREDITS:4,MARKS: 95
Subject 4- CREDITS:3,MARKS: 88
Subject 5- CREDITS:3,MARKS: 87
Subject 6- CREDITS:4,MARKS: 89
Subject 7- CREDITS:4,MARKS: 90

SGPA: 9.6

```

```
Enter USN:1bf24cs303
Enter the name of student:
karanyadav
Enter details of credits and marks in order for 4 subjects:
Enter credits for subject1:
3
Enter marks for 1:
88
Enter credits for subject2:
4
Enter marks for 2:
88
Enter credits for subject3:
4
Enter marks for 3:
90
Enter credits for subject4:
4
Enter marks for 4:
94
Enter credits for subject5:
4
Enter marks for 5:
95
Enter credits for subject6:
4
Enter marks for 6:
96
Enter credits for subject7:
4
Enter marks for 7:
80
STUDENT DETAILS
USN:1bf24cs303
NAME:karanyadav
Subject 1- CREDITS:3,MARKS: 88
Subject 2- CREDITS:4,MARKS: 88
Subject 3- CREDITS:4,MARKS: 90
Subject 4- CREDITS:4,MARKS: 94
Subject 5- CREDITS:4,MARKS: 95
Subject 6- CREDITS:4,MARKS: 96
Subject 7- CREDITS:4,MARKS: 80
SGPA: 9.592592592593
```

Program 3: Bookstore Program

Code:

```
import java.util.Scanner;

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

    Books(String name, String author, int price, int num_pages)
    {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }
    public String toString()
    {
        String name = "Book name: " + this.name + "\n";
        String author = "Author name: " + this.author + "\n";
        String price = "Price: " + this.price + "\n";
        String num_pages = "Number of pages: " + this.num_pages + "\n";
        return name + author + price + num_pages;
    }
}

public class BookStore
{
    public static void main(String args[])
    {
        Scanner in = new Scanner(System.in);
        String name, author;
        int price, num_pages;

        System.out.print("Enter number of books: ");
        int n = in.nextInt();

        Books[] b = new Books[n];

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

            System.out.print("Name: ");
            name = in.next();
```

```
System.out.print("Author: ");
author = in.next();

System.out.print("Price: ");
price = in.nextInt();

System.out.print("Number of pages: ");
num_pages
= in.nextInt();

b[i] = new Books(name, author, price, num_pages
);
}

System.out.println("\n\nBook details:\n");

for (int i = 0; i < n; i++)
{
    System.out.println(b[i].toString());
}

in.close();
}
```

Output:

```
Enter number of books: 2
```

```
Enter details for book 1:
```

```
Name: Twilight
```

```
Author: Meyer
```

```
Price: 999
```

```
Number of pages: 500
```

```
Enter details for book 2:
```

```
Name: IT
```

```
Author: King
```

```
Price: 1499
```

```
Number of pages: 800
```

```
Book details:
```

```
Book name: Twilight
```

```
Author name: Meyer
```

```
Price: 999
```

```
Number of pages: 500
```

```
Book name: IT
```

```
Author name: King
```

```
Price: 1499
```

```
Number of pages: 800
```

Program 4: Shapes Program

Code:

```
import java.util.Scanner;

class InputScanner {
    Scanner sc = new Scanner(System.in);

    int getInt(String msg) {
        System.out.print(msg);
        return sc.nextInt();
    }

    double getDouble(String msg) {
        System.out.print(msg);
        return sc.nextDouble();
    }
}

abstract class Shape extends InputScanner {
    int a, b;
    abstract void printArea();
}

class Rectangle extends Shape {
    void input() {
        a = getInt("Enter length: ");
        b = getInt("Enter breadth: ");
    }

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

class Triangle extends Shape {
    void input() {
        a = getInt("Enter base: ");
        b = getInt("Enter height: ");
    }

    void printArea() {
        System.out.println("Area of Triangle = " + (0.5 * a * b));
    }
}

class Circle extends Shape {
    void input() {
```

```

        a = getInt("Enter radius: ");
    }

    void printArea() {
        System.out.println("Area of Circle = " + (3.1416 * a * a));
    }
}

public class MainShape {
    public static void main(String[] args) {
        System.out.println("Enter the dimensions of rectangle:");
        Rectangle r = new Rectangle();
        r.input();
        r.printArea();

        System.out.println("\nEnter the dimensions of triangle:");
        Triangle t = new Triangle();
        t.input();
        t.printArea();

        System.out.println("\nEnter the dimension of circle:");
        Circle c = new Circle();
        c.input();
        c.printArea();
    }
}

```

Output:

Enter the dimensions of rectangle:
Enter length: 10
Enter breadth: 5
Area of Rectangle = 50

Enter the dimensions of triangle:
Enter base: 10
Enter height: 8
Area of Triangle = 40.0

Enter the dimension of circle:
Enter radius: 7
Area of Circle = 153.9384

Program 5: Bank Program

Code:

```
import java.util.Scanner;

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

    Account(String name, int accNo, String type) {
        customerName = name;
        accountNumber = accNo;
        accountType = type;
        balance = 0.0;
    }

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

    void display() {
        System.out.println("Customer name: " + customerName);
        System.out.println("Account number: " + accountNumber);
        System.out.println("Type of Account: " + accountType);
        System.out.println("Account Balance: " + balance);
    }
}

class SavAcct extends Account {
    SavAcct(String name, int accNo, String type) {
        super(name, accNo, type);
    }

    void computeInterest() {
        double rate = 0.04;
        double interest = balance * rate;
        balance += interest;
        System.out.println("Interest added: " + interest + ". Updated balance: " + balance);
    }

    void withdraw(double amount) {
        if (amount > balance) {
            System.out.println("Insufficient balance!");
        } else {
    
```

```

        balance -= amount;
        System.out.println("Withdrawn: " + amount + ". Updated balance: " + balance);
    }
}
}

class CurAcct extends Account {
    final double MIN_BAL = 500.0;
    final double SERVICE_CHARGE = 100.0;

    CurAcct(String name, int accNo, String type) {
        super(name, accNo, type);
    }

    void checkBalance() {
        if (balance < MIN_BAL) {
            balance -= SERVICE_CHARGE;
            System.out.println("Balance below minimum. Service charge imposed: " +
SERVICE_CHARGE);
            System.out.println("Updated balance: " + balance);
        }
    }

    void withdraw(double amount) {
        if (amount > balance) {
            System.out.println("Insufficient balance!");
        } else {
            balance -= amount;
            checkBalance();
            System.out.println("Withdrawn: " + amount + ". Updated balance: " + balance);
        }
    }
}

public class BankSystem {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Account[] accounts = new Account[2];

        for (int i = 0; i < 2; i++) {
            System.out.print("Enter customer name: ");
            String name = sc.nextLine();
            System.out.print("Enter account number: ");
            int accNo = Integer.parseInt(sc.nextLine());
            System.out.print("Enter type of account (saving/current): ");
            String type = sc.nextLine().toLowerCase();
        }
    }
}

```

```

if (type.equals("saving"))
    accounts[i] = new SavAcct(name, accNo, type);
else
    accounts[i] = new CurAcct(name, accNo, type);
}

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 Savings Account");
    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 the type of account (saving/current): ");
            String type = sc.next().toLowerCase();
            System.out.print("Enter the deposit amount: ");
            double depAmt = sc.nextDouble();
            for (Account acc : accounts) {
                if (acc.accountType.equals(type)) {
                    acc.deposit(depAmt);

                    break;
                }
            }
            break;

        case 2:
            System.out.print("Enter the type of account (saving/current): ");
            type = sc.next().toLowerCase();
            System.out.print("Enter the withdrawal amount: ");
            double wAmt = sc.nextDouble();
            for (Account acc : accounts) {
                if (acc.accountType.equals(type)) {
                    if (type.equals("saving"))
                        ((SavAcct) acc).withdraw(wAmt);
                    else
                        ((CurAcct) acc).withdraw(wAmt);

                    break;
                }
            }
    }
}

```

```

        break;

case 3:
    System.out.print("Enter the type of account (saving/current): ");
    type = sc.nextLine().toLowerCase();
    for (Account acc : accounts) {
        if (acc.accountType.equals("saving") && type.equals("saving")) {
            ((SavAcct) acc).computeInterest();

            break;
        }
    }
    break;

case 4:
    System.out.print("Enter the type of account (saving/current): ");
    type = sc.nextLine().toLowerCase();
    for (Account acc : accounts) {
        if (acc.accountType.equals(type)) {
            acc.display();
            break;
        }
    }
    break;

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

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

} while (choice != 5);

sc.close();
}

```

Output:

```
Enter customer name for savings account: sudhir
Enter account Number: 1253453311
Enter customer name for current account: suryansh
Enter account Number: 123456789
```

```
-----MENU-----
```

- 1. Deposit
- 2. Withdraw
- 3. Compute interest for SavingsAccount
- 4. Display account details
- 5. Exit

```
Enter your choice: 1
```

```
Enter the type of account: saving
```

```
Enter the deposit amount: 2000
```

```
-----MENU-----
```

- 1. Deposit
- 2. Withdraw
- 3. Compute interest for SavingsAccount
- 4. Display account details
- 5. Exit

```
Enter your choice: 2
```

```
Enter the type of account: saving
```

```
Enter the withdrawal amount: 500
```

```
-----MENU-----
```

- 1. Deposit
- 2. Withdraw
- 3. Compute interest for SavingsAccount
- 4. Display account details
- 5. Exit

```
Enter your choice: 3
```

```
Enter the rate of interest: 10
```

```
Enter the time period (years): 2
```

```
Interest added = 315.000000000002
```

```
-----MENU-----
```

- 1. Deposit
- 2. Withdraw
- 3. Compute interest for SavingsAccount
- 4. Display account details
- 5. Exit

```
Enter your choice: 4
```

```
Enter the type of account: saving
```

```
Customer name: sudhir
```

```
Account number: 1253453311
```

```
Type of Account: saving
```

```
Balance = 1815.000000000002
```

```
-----MENU-----
```

- 1. Deposit
- 2. Withdraw
- 3. Compute interest for SavingsAccount
- 4. Display account details
- 5. Exit

```
Enter your choice: 5
```

```
Exiting...
```

```
PS C:\Users\sudhi\.vscode\java> █
```

Program 6: Packages

Code:

```
package CIE;
import java.util.Scanner;

public class Student {
    protected String usn;
    protected String name;
    protected int sem;

    public void inputStudentDetails() {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter USN: ");
        usn = s.nextLine();

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

        System.out.print("Enter Semester: ");
        sem = s.nextInt();
    }

    public void displayStudentDetails() {
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
        System.out.println("Semester: " + sem);
    }
}

package CIE;
import java.util.Scanner;

public class Internals extends Student {

    protected int marks[] = new int[5];

    public void inputCIEmarks() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter 5 CIE marks: ");

        for (int i = 0; i < 5; i++) {
            System.out.print("CIE Mark in Subject " + (i + 1) + ": ");
            marks[i] = s.nextInt();
        }
    }
}
```

```

}

package SEE;

import CIE.Internals;
import java.util.Scanner;

public class Externals extends Internals {

    protected int marks[];
    protected int finalMarks[];

    public Externals() {
        marks = new int[5];
        finalMarks = new int[5];
    }

    public void inputSEEmarks() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter 5 SEE marks: ");

        for (int i = 0; i < 5; i++) {
            System.out.print("SEE Mark in Subject " + (i + 1) + ": ");
            marks[i] = s.nextInt();
        }
    }

    public void calculateFinalMarks() {
        for (int i = 0; i < 5; i++) {
            finalMarks[i] = super.marks[i] + marks[i]/2; // CIE + SEE
        }
    }

    public void displayFinalMarks() {
        System.out.println("\n--- Final Marks ---");
        displayStudentDetails();

        for (int i = 0; i < 5; i++) {
            System.out.println("Final Marks in Subject " + (i + 1) + ": " + finalMarks[i]);
        }
    }
}

import SEE.Externals; class Main {
    public static void main(String args[]) {

        Externals obj = new Externals(

```

```
obj.inputStudentDetails();
obj.inputCIEmarks();
obj.inputSEEmarks();
obj.calculateFinalMarks();
obj.displayFinalMarks();
}
}
```

OUTPUT:

```
PS D:\1BF24CS304> java Main.java
Enter number of students
2
Enter the name:
Sudhir
Enter the usn:
1BF24CS304
Enter the sem:
3
Enter the subject 1 CIE marks:
36
Enter the subject 2 CIE marks:
48
Enter the subject 3 CIE marks:
47
Enter the subject 4 CIE marks:
46
Enter the subject 5 CIE marks:
32
Enter the subject 1 SEE marks:
78
Enter the subject 2 SEE marks:
98
Enter the subject 3 SEE marks:
85
Enter the subject 4 SEE marks:
78
Enter the subject 5 SEE marks:
89
Name of the student is: Sudhir
USN is: 1BF24CS304
SEM is: 3
The final marks in all 5 courses are:
Course 1 is: 75
Course 2 is: 97
Course 3 is: 89
Course 4 is: 85
Course 5 is: 76
```

```
Enter the name:  
Sahil  
Enter the usn:  
1BF23CS  
Enter the sem:  
3  
Enter the subject 1 CIE marks:  
47  
Enter the subject 2 CIE marks:  
42  
Enter the subject 3 CIE marks:  
43  
Enter the subject 4 CIE marks:  
26  
Enter the subject 5 CIE marks:  
32  
Enter the subject 1 SEE marks:  
89  
Enter the subject 2 SEE marks:  
86  
Enter the subject 3 SEE marks:  
98  
Enter the subject 4 SEE marks:  
95  
Enter the subject 5 SEE marks:  
76  
Name of the student is: Sahil  
USN is: 1BF23CS  
SEM is: 3  
The final marks in all 5 courses are:  
Course 1 is: 91  
Course 2 is: 85  
Course 3 is: 92  
Course 4 is: 73  
Course 5 is: 70
```

Program 7:

```
import java.util.*;  
  
class WrongAge extends Exception {  
    WrongAge(String message) {  
        super(message);  
    }  
}  
  
class InputScanner {  
    Scanner sc;  
  
    InputScanner() {  
        sc = new Scanner(System.in);  
    }  
  
    int getInt() {  
        return sc.nextInt();  
    }  
}  
  
class Father extends InputScanner {  
    int fatherAge;  
  
    Father() throws WrongAge {  
        System.out.print("Enter Father's Age: ");  
        fatherAge = getInt();  
  
        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 = getInt();

if (sonAge < 0) {
    throw new WrongAge("Age cannot be negative");
}

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

void display() {
    super.display();
    System.out.println("Son's Age: " + sonAge);
}

public class Main {
    public static void main(String[] args) {
        try {
            Son s = new Son();
            s.display();

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

```

Output:

```
Enter father's age: 25
Enter son's age: 25
Father's age is: 25
Exception Caught: Son's age must be less than father's age!
> PS C:\Users\sudhi\.vscode\java\labprogram7> java Main
Enter father's age: 25
Enter son's age: 2
Father's age is: 25
Son's age is: 2
> PS C:\Users\sudhi\.vscode\java\labprogram7> java Main
Enter father's age: 15
Enter son's age: 16
Father's age is: 15
Exception Caught: Son's age must be less than father's age!
> PS C:\Users\sudhi\.vscode\java\labprogram7>
```

PROGRAM 8:

```
class MessageThread extends Thread {  
    String message;  
    int interval;  
    int count;  
  
    MessageThread(String msg, int time, int count) {  
        this.message = msg;  
        this.interval = time;  
        this.count = count;  
    }  
  
    public void run() {  
        try {  
            for (int i = 1; i <= count; i++) {  
                System.out.println(message);  
                Thread.sleep(interval);  
            }  
        } catch (InterruptedException e) {  
            System.out.println("Thread Interrupted");  
        }  
    }  
}  
  
public class Main {  
    public static void main(String[] args) {  
  
        MessageThread t1 = new MessageThread("BMS College of Engineering", 10000, 5);  
  
        MessageThread t2 = new MessageThread("CSE", 2000, 10);  
  
        t1.start();  
        t2.start();  
    }  
}
```

Output:

```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
```

Program 9: Open Ended Question 1

Code:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class DivisionUI extends JFrame implements ActionListener {

    JTextField num1Field, num2Field, resultField;
    JButton divideButton;

    DivisionUI() {
        setTitle("Integer Division");
        setSize(350, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new GridLayout(4, 2, 5, 5));

        add(new JLabel("Num 1:"));
        num1Field = new JTextField();
        add(num1Field);

        add(new JLabel("Num 2:"));
        num2Field = new JTextField();
        add(num2Field);

        divideButton = new JButton("Divide");
        add(divideButton);

        divideButton.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == divideButton) {
            int num1 = Integer.parseInt(num1Field.getText());
            int num2 = Integer.parseInt(num2Field.getText());
            resultField.setText(String.valueOf(num1 / num2));
        }
    }
}
```

```

        add(new JLabel("Result:"));
        resultField = new JTextField();
        resultField.setEditable(false);
        add(resultField);

        setVisible(true);
    }

public void actionPerformed(ActionEvent e) {
    try {
        int num1 = Integer.parseInt(num1Field.getText());
        int num2 = Integer.parseInt(num2Field.getText());

        if (num2 == 0) {
            throw new ArithmeticException("Cannot divide by zero");
        }

        int result = num1 / num2;
        resultField.setText(Integer.toString(result));

    } catch (NumberFormatException ex) {
        JOptionPane.showMessageDialog(this,
            "Please enter valid integers!",
            "Number Format Error",
            JOptionPane.ERROR_MESSAGE);

    } catch (ArithmeticException ex) {
        JOptionPane.showMessageDialog(this,
            ex.getMessage(),
            "ArithmetiC Error",
            JOptionPane.ERROR_MESSAGE);
    }
}

public static void main(String[] args) {
    new DivisionUI();
}
}

```

Output:

Program 10: Open Ended Question 2

Code:

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

```

this.n = n;
valueSet = true;
System.out.println("Put: " + n);

System.out.println("\nIntimate Consumer\n");
notify();
}
}

class Producer implements Runnable {

    Q q;

    Producer(Q q) {
        this.q = q;
        new Thread(this, "Producer").start();
    }

    public void run() {
        int i = 0;
        while (i < 5) {
            q.put(i++);
        }
    }
}

class Consumer implements Runnable {

    Q q;

    Consumer(Q q) {
        this.q = q;
        new Thread(this, "Consumer").start();
    }

    public void run() {
        int i = 0;
        while (i < 5) {
            int r = q.get();
            System.out.println("Consumed: " + r);
            i++;
        }
    }
}

```

