VISVESVARAYA TECHNOLOGICAL UNIVERSITY

**"JnanaSangama", Belgaum -590014, Karnataka.**

**LAB REPORT on**

# Object Oriented Java Programming

# (23CS3PCOOJ)

***Submitted by***

**Bhakti basavaraj Varadai (1BM24IC015)**

***in partial fulfilment 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 **Bhakti basavaraj Varadai (1BM24IC015),** 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<br>Associate Professor<br>Department of CSE, BMSCE | Dr. Kavitha Sooda<br>Professor & HOD<br>Department of CSE, BMSCE |

# INDEX

Github Link:
https://github.com/Bhaktivaradai/JAVA

**Program 1**

Implement Quadratic Equation

labprog1.java > labprog1 > main(String[])

```java
import java.util.Scanner;
class labprog1{
    Run | Debug
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        double a, b, c, r1, r2, d;
        System.out.println(x: "enter the coefficient");
        a = sc.nextDouble();
        b = sc.nextDouble();
        c = sc.nextDouble();

        if(a == 0)
            System.out.println(x: "not a quadratic");
        else {
            d = b*b - 4*a*c;

            if(d > 0) {
                System.out.println(x: "roots are real and distinct");
                r1 = (-b + Math.sqrt(d)) / (2*a);
                r2 = (-b - Math.sqrt(d)) / (2*a);
                System.out.println(x: "roots are:");
                System.out.println(r1);
                System.out.println(r2);
            }
            else if(d < 0) {
                System.out.println(x: "roots are imaginary");
                r1 = -b / (2*a);
                r2 = Math.sqrt(Math.abs(d)) / (2*a);
                System.out.println(x: "roots are:");
```

```
28                  System.out.println(x: "roots are:");
29
30
31                  System.out.println(r1 + " + i" + r2);
32                  System.out.println(r1 + " - i" + r2);
33
34              }
35              else {
36                  System.out.println(x: "roots are real and equal");
37                  r1 = -b / (2*a);
38                  System.out.println(x: "roots are:");
39                  System.out.println(r1);
40              }
41          }
42      }
43  }
44
```

OUTPUT :

```
PS C:\Users\Lenovo\Desktop\java lab> cd "c:\Users\Lenovo\Desktop\java lab\" ;
enter the coefficient
1 2 3
roots are imaginary
roots are:
-1.0 + i1.4142135623730951
-1.0 - i1.4142135623730951
PS C:\Users\Lenovo\Desktop\java lab> cd "c:\Users\Lenovo\Desktop\java lab\" ;
enter the coefficient
2 3 1
roots are real and distinct
roots are:
-0.5
-1.0
PS C:\Users\Lenovo\Desktop\java lab>
```

Program 2 : STUDENT SGPA

```java
import java.util.Scanner;
class Student {
    String name;
    String usn;
    int []marks;
    int []credits;
    double calculateSGPA(int n) {
        int totalcredits=0;
        int totalpoints=0;
        for(int i=0;i<n;i++) {
        int gradepoint=marks[i]/10;
        totalpoints+=gradepoint*credits[i];
        totalcredits+=credits[i];
        }
        return (double) totalpoints/totalcredits;
    }
    void display(int n) {
        System.out.println("nsme is"+name);
        System.out.println("usn is"+usn);
        System.out.println("sgpa is"+calculateSGPA(n));
    }
}
class lab2 {
    Run | Debug
    public static void main(String [] args) {
        Scanner sc=new Scanner(System.in);
        int n;
        Student s=new Student();

        System.out.println(x: "enter name");
        s.name=sc.nextLine();
        System.out.println(x: "enter usn");
        s.usn=sc.nextLine();
        System.out.println(x: "enter no of sub");
        n=sc.nextInt();
        System.out.println(x: "enter marks and credits");
        s.marks=new int[n];
        s.credits=new int[n];
        for(int i=0;i<n;i++){
            s.marks[i]=sc.nextInt();
            s.credits[i]=sc.nextInt();
        }
        s.display(n);

    }
}
```

**OUTPUT :**

```
PS C:\Users\Lenovo\Desktop\java lab> cd "c:\Users\Lenovo\Desktop\java lab\lab prep\" ;
enter name
bhakti
enter usn
1bm24ic015
enter no of sub
2
enter marks and credits
78 3
90 4
nsme isbhakti
usn is1bm24ic015
sgpa is8.142857142857142
PS C:\Users\Lenovo\Desktop\java lab\lab prep>
```

**Program 3: BOOK**

```
import java.util.Scanner;
class Book{
    String name;
    String author;
    int numPages;
    int price;
    Book(String name,String author,int numPages,int price){
        this.name=name;
        this.author=author;
        this.numPages=numPages;
        this.price=price;

    }
    public String toString(){
        String name ,author,numPages,price;
        name="Book name : " +this.name+"\n";
        author="Book author : "+this.author+"\n";
        numPages="Number of pages : "+this.numPages+"\n";
        price="Price : "+this.price+"\n";
        return name+author+numPages+ price;
    }
}
public class Lab3{
    public static void main(String[] args){
        Scanner s=new Scanner(System.in);
        String name;
        String author;
        int numPages;
        int price;
```

```java
public class Lab3{
    public static void main(String[] args){
        Scanner s=new Scanner(System.in);
        String name;
        String author;
        int numPages;
        int price;

        System.out.print(s: "Enter the number of books : ");
        int n=s.nextInt();
        Book [] b;
        b=new Book[n];
        for(int i=0;i<n;i++){
            System.out.print("Enter the name of the book "+(i+1)+":");
            name=s.next();
            System.out.print("Enter the author of the book "+(i+1)+":");
            author=s.next();
            System.out.print("Enter the number of pages in the book "+(i+1)+":");
            numPages=s.nextInt();
            System.out.print("Enter the price of the book "+(i+1)+":");
            price=s.nextInt();
            b[i]=new Book(name,author,numPages,price);
        }
        for(int i=0;i<n;i++){
            System.out.println(b[i].toString());
        }


    }
}
```

OUTPUT

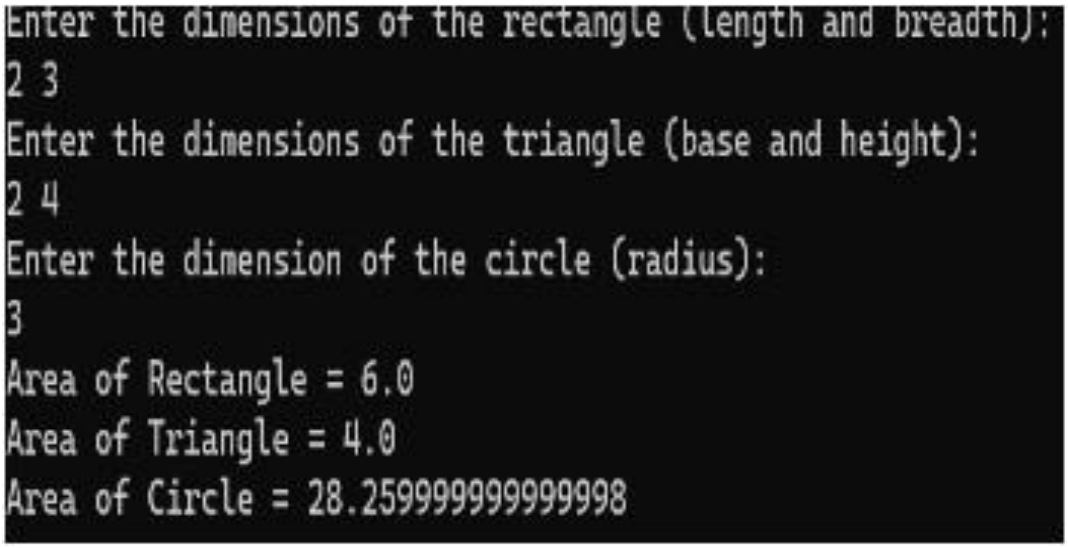
```
Enter the number of books : 2
Enter the name of the book 1:HarryPotter
Enter the author of the book 1:JKRowling
Enter the number of pages in the book 1:1000
Enter the price of the book 1:300
Enter the name of the book 2:Panchatantra
Enter the author of the book 2:VishnuSharma
Enter the number of pages in the book 2:1050
Enter the price of the book 2:400
Book name: HarryPotter
Author name: JKRowling
Price: 1000
Number of pages: 300

Book name: Panchatantra
Author name: VishnuSharma
Price: 1050
Number of pages: 400

Press any key to continue . . .
```

## Program 4: ABSTRACT CLASS

```
import java.util.Scanner;
class InputScanner{
    Scanner s=new Scanner(System.in);
}
abstract class Shape extends InputScanner{
    int dimension1;
    int dimension2;
    abstract void printArea();
}
class Rectangle extends Shape{
    void input(){
        System.out.print(s: "Enter the dimensions of rectecgle (length and breadth) : ");
        dimension1=s.nextInt();
        dimension2=s.nextInt();
    }
    void printArea(){
        double area=dimension1*dimension2;
        System.out.println("Area of rectange is :"+area);
    }
}
class Triangle extends Shape{
    void input(){
        System.out.println(x: "Enter the dimension(height and breadth): ");
        dimension1=s.nextInt();
        dimension2=s.nextInt();
    }
    void printArea(){
        double area=0.5*dimension1*dimension2;
        System.out.println("Area of triangle is : "+area);
    }
}
```

```
class Circle extends Shape{
    void input(){
        System.out.println(x: "Enter the dimenson(radius) :");
        dimension1=s.nextInt();
    }
    void printArea(){
        double area=Math.PI*dimension1*dimension1;
        System.out.println("Area is :"+area);
    }
}
public class Lab4{
    public static void main(String[] args){
        Rectangle r=new Rectangle();
        Triangle t=new Triangle();
        Circle c=new Circle();
        r.input();
        t.input();
        c.input();
        r.printArea();
        t.printArea();
        c.printArea();
    }
}
```

**OUTPUT**


Enter the dimensions of the rectangle (length and breadth):
2 3
Enter the dimensions of the triangle (base and height):
2 4
Enter the dimension of the circle (radius):
3
Area of Rectangle = 6.0
Area of Triangle = 4.0
Area of Circle = 28.259999999999998

## Program 5: BANK

```
import java.util.Scanner;
import java.lang.Math;
abstract class Account{
    protected String customerName;
    protected String accountNumber;
    protected String accountType;
    protected double balance;
    Account(String customerName,String accountNumber,String accountType,double initialBalance){
        this.customerName=customerName;
        this.accountNumber=accountNumber;
        this.accountType=accountType;
        this.balance=initialBalance;
    }
    public void deposit(double amount){
        if(amount>0){
            balance+=amount;
            System.out.println("Deposit:$"+amount);
            displayBalance();
        }
    }
    public void displayBalance(){
        System.out.println("current balance for"+customerName+"("+accountType+") : $"+balance);

    }
    public abstract void computeAndDepositInterest();
    public abstract void permitWithdrawal(double amount);
    public abstract void checkForMinimumBalance();
```

```
class SavAcct extends Account{
    private static final double RATE=0.05;
    public SavAcct(String customerName,String accountNumber,double initialBalance){
        super(customerName,accountNumber,accountType: "Savings",initialBalance);

    }
    @Override
    public void computeAndDepositInterest(){
        double interest=balance*RATE;
        balance+=interest;
        System.out.println("Interest (5%) of $"+interest+"deposited");
        displayBalance();
    }
    @Override
    public void permitWithdrawal(double amount){
        if(amount>0 && balance>=amount){
            balance-=amount;
            System.out.println("Withdrew :$"+amount);
            displayBalance();
        }else{
            System.out.println(x: "Insufficient funds or invalid withdrawal amount");
        }
    }
    @Override
    public void checkForMinimumBalance(){

    }
}
```

```
class CurrAcct extends Account{
    private static final double Min_balance=500.0;
    private static final double PENALTY=50.0;

    public CurrAcct(String customerName,String accountNumber,double initialBalance){
        super(customerName,accountNumber,accountType: "Current ",initialBalance);

    }
    @Override
    public void computeAndDepositInterest(){
        System.out.println(x: "Current ban does not earn interest");

    }
    @Override
    public void permitWithdrawal(double amount){
        if(amount>0 && balance>=amount){
            balance-=amount;
            System.out.println("Withdrew :$"+amount);
            checkForMinimumBalance();
        }else{
            System.out.println(x: "Insufficient funds or invalid withdrawal amount");
        }
    }
```

```
@Override
public void permitWithdrawal(double amount){
    if(amount>0 && balance>=amount){
        balance-=amount;
        System.out.println("Withdrew :$"+amount);
        checkForMinimumBalance();
    }else{
        System.out.println(x: "Insufficient funds or invalid withdrawal amount");
    }
}
@Override
public void checkForMinimumBalance(){
    if(balance<Min_balance && balance>0){
        balance-=PENALTY;
        System.out.println("bALANCE FELL BELOW OF $"+Min_balance+"."+PENALTY+"imposed");
        displayBalance();

    }else if(balance<=0){
        System.out.println(x: "Account is overdrawn or empty.no further actions possible");
    }

}
```

```java
public class BankDemo{
    public static void main(String[] args) {
        System.out.println(x: "---TESTING SAVNGS ACCOUNT");
        SavAcct sa = new SavAcct(customerName: "Alice",accountNumber: "SA1234",initialBalance: 1000.0);
        sa.displayBalance();
        sa.deposit(amount: 200.0);
        sa.permitWithdrawal(amount: 50.0);
        sa.computeAndDepositInterest();
        sa.checkForMinimumBalance();
        System.out.println(x: "---TESTING CURRENT ACCOUNT");
        CurrAcct ca = new CurrAcct(customerName: "BOB",accountNumber: "CA1234",initialBalance: 600.0);
        ca.displayBalance();
        ca.deposit(amount: 50.0);
        ca.computeAndDepositInterest();
        ca.permitWithdrawal(amount: 100.0);
        ca.checkForMinimumBalance();



    }
}
```

**OUTPUT**

```
---TESTING SAVNGS ACCOUNT
current balance forAlice(Savings) : $1000.0
Deposit:$200.0
current balance forAlice(Savings) : $1200.0
Withdrew :$50.0
current balance forAlice(Savings) : $1150.0
Interest (5%) of $57.5deposited
current balance forAlice(Savings) : $1207.5
---TESTING CURRENT ACCOUNT
current balance forBOB(Current ) : $600.0
Deposit:$50.0
current balance forBOB(Current ) : $650.0
Current ban does not earn interest
Withdrew :$100.0
Press any key to continue . . .
```

**Program 6: PACKAGE**

PACKAGE CIE

```java
package CIE;
import java.util.Scanner;
public class Internals extends Student{
    protected int cieMarks[]=new int[5];
    public void inputCIEmarks(){
        Scanner s=new Scanner(System.in);
        for(int i=0;i<5;i++){
            System.out.print("Enter CIE marks for subject "+(i+1)+": ");
            cieMarks[i]=s.nextInt();
        }
    }
}
```

```java
package CIE;
import java.util.Scanner;
public class Student{
    protected String usn=new String();
    protected String name =new String();
    protected int sem;
    public void inputStudentDetails(){
        Scanner s=new Scanner(System.in);
        System.out.println("Enter student name, USN and sem: ");
        name=s.next();
        usn=s.next();
        sem=s.nextInt();
    }
    public void displayStudentDetails(){
        System.out.println("Name: "+name);
        System.out.println("USN: "+usn);
        System.out.println("Sem: "+sem);
    }
}
```

PACKAGE SEE

```java
package SEE;
import CIE.Internals;
import java.util.Scanner;
public class Externals extends Internals{
    protected int seeMarks[]=new int[5];
    protected int finalMarks[]=new int[5];
    public void inputSEEmarks(){
        Scanner s=new Scanner(System.in);
        for(int i=0;i<5;i++){
            System.out.print("Enter SEE marks for subject "+(i+1)+": ");
            seeMarks[i]=s.nextInt();
        }
    }
    public void calculateFinalMarks(){
        for(int i=0;i<5;i++){
            finalMarks[i]=seeMarks[i]+cieMarks[i];
        }
    }
    public void displayFinalMarks(){
        System.out.println("---Final Marks---");
        for(int i=0;i<5;i++){
            System.out.println("Subject "+(i+1)+": "+finalMarks[i]);
        }
```

```java
import SEE.Externals;
import java.util.Scanner;
public class Lab6 {
    public static void main(String[] args) {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter no. of students: ");
        int n=s.nextInt();
        Externals[] students=new Externals[n];
        for(int i=0;i<n;i++){
            System.out.println("Enter details of student "+(i+1)+": ");
            students[i]=new Externals();
            students[i].inputStudentDetails();
            students[i].inputCIEmarks();
            students[i].inputSEEmarks();
            students[i].calculateFinalMarks();
        }
        for(int i=0;i<n;i++){
            System.out.println("\nStudent "+(i+1));
```

```
            students[i].displayFinalMarks();
        }
    }
}
```

**OUTPUT**

```
Enter no. of students:
1
Enter details of student 1:
Enter student name, USN and sem:
Alice
1BM24CS100
3
Enter CIE marks for subject 1: 40
Enter CIE marks for subject 2: 40
Enter CIE marks for subject 3: 50
Enter CIE marks for subject 4: 35
Enter CIE marks for subject 5: 40
Enter SEE marks for subject 1: 35
Enter SEE marks for subject 2: 45
Enter SEE marks for subject 3: 48
Enter SEE marks for subject 4: 40
Enter SEE marks for subject 5: 35

Student 1
---Final Marks---
Subject 1: 75
Subject 2: 85
Subject 3: 98
Subject 4: 75
Subject 5: 75
```

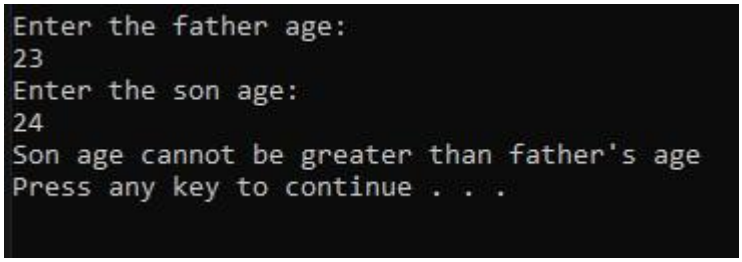
**PROGRAM 7: EXCEPTION**

```
import java.util.Scanner;
class WrongAge extends Exception{
    WrongAge(){
        super(message: "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.println(x: "Enter the father age:");
        fatherAge =s.nextInt();
        if(fatherAge<0){
            throw new WrongAge(msg: "Age cannot be negative");
        }

    }
    public void displayFatherAge(){
        System.out.println("Father age is :"+ fatherAge);
    }
}
```

```
class Son extends Father{
    int sonAge;
    Son() throws WrongAge{
        System.out.println(x: "Enter the son age:");

        sonAge=s.nextInt();
        if(sonAge>fatherAge){
            throw new WrongAge(msg: "Son age cannot be greater than father's age");
        }
        else if(sonAge<0){
            throw new WrongAge(msg: "Age cannot be negative,throw new WrongAge");

        }

    }
    public void displaySonAge(){
        System.out.println("Son age is :"+ fatherAge);
    }
}
```

```
class ExceptionClass{
    public static void main(String[] args) {
        try{
            Father f=new Father();
            Son s=new Son();
            f.displayFatherAge();
            s.displaySonAge();
        }

        catch(Exception e){
            System.out.println(e.getMessage());
        }

    }

}
```

**OUTPUT**

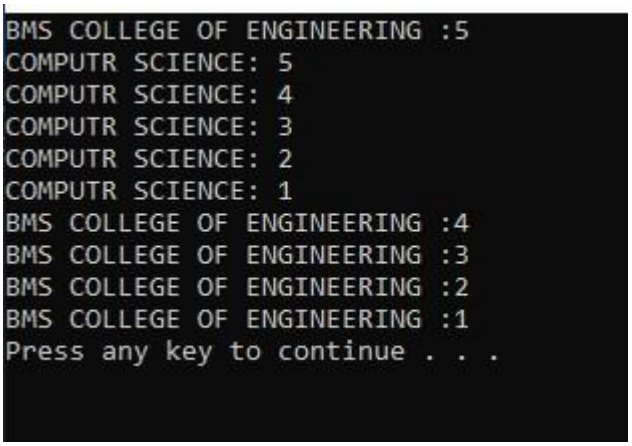

Enter the father age:
23
Enter the son age:
24
Son age cannot be greater than father's age
Press any key to continue . . .

## PROGRAM 8: MULTITHREADING

```
class BMS extends Thread
{
    public void run()
    {
            try {
                for(int i=5;i>=1;i--){
                    System.out.println("BMS COLLEGE OF ENGINEERING :" + i);
                    Thread.sleep(10000);
                }
            } catch (InterruptedException e) {

                System.out.println("BMS Thread is quitting");
            }
    }
}
class CS extends Thread{
    public void run()
    {
            try {
                for(int i=5;i>=1;i--){
                    System.out.println("COMPUTR SCIENCE: " + i);
                    Thread.sleep(2000);

                }
            } catch (InterruptedException e) {
                System.out.println("CS Thread is quitting");
            }
    }
}
```

```
public class BMS_CS{
    public static void main(String args[])
    {
        BMS b=new BMS();
        CS cs=new CS();
        b.start();
        cs.start();
    }

}
```

**OUTPUT**


BMS COLLEGE OF ENGINEERING :5
COMPUTR SCIENCE: 5
COMPUTR SCIENCE: 4
COMPUTR SCIENCE: 3
COMPUTR SCIENCE: 2
COMPUTR SCIENCE: 1
BMS COLLEGE OF ENGINEERING :4
BMS COLLEGE OF ENGINEERING :3
BMS COLLEGE OF ENGINEERING :2
BMS COLLEGE OF ENGINEERING :1
Press any key to continue . . .