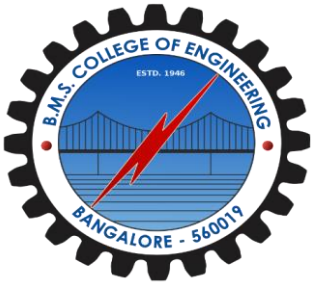
B.M.S. COLLEGE OF ENGINEERING

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum) Bull Temple Road, Basavanagudi, Bengaluru – 560019



LAB REPORT

On

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted By:

TUSHAR TYAGI

(1BM22CS311)

In partial fulfilment of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

2023-24

B.M.S. COLLEGE OF ENGINEERING

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)
Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of

Computer Science & Engineering (CSE)

CERTIFICATE

This is to certify that the Lab work entitled "Object Oriented Programming in Java (23CS3PCOOJ)" conducted by **Tushar Tyagi (1BM22CS311)** who is bonafide student at **B.M.S.College of Engineering**. It is in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** during the academic year 2023-24. The Lab report has been approved as it satisfies the academic requirements in respect of Object Oriented Programming in Java (22CS3PCOOJ) work prescribed for the said degree.

Tushar Tyagi
1BM22CS311

Shravya AR
Assistant Professor
Department of CSE
BMSCE, Bengaluru-19

Table of contents

Sl. No.	Program Title	Page No.
1	Solution to Quadratic equation	
2	Student details and Percentage calculation	
3	Book Details	
4	Calculating Area of different Shapes	
5	Bank Details	
6	CIE and SEE marks details(PACKAGES)	
7	Exception Handling	
8	Threads	

PROGRAM 1:

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

```
import java.util.Scanner;
class Quad{
int a, b, c;
double root1, root2, d;
Scanner s = new Scanner(System.in);
void input()
{
System.out.println("Quadratic equation is in the form : ax^2 + bx + c");
System.out.print("Enter a:");
a = s.nextInt();
System.out.print("Enter b:");
b = s.nextInt();
System.out.print("Enter c:");
c = s.nextInt();
}
void discriminant() {
d= (b*b)-(4*a*c);
}
void calculateRoots() {
if(d>0)
{
System.out.println("Roots are real and unequal");
root1 = ( - b + Math.sqrt(d))/(2*a);
root2 = (-b - Math.sqrt(d))/(2*a);
System.out.println("First root is:"+root1);
System.out.println("Second root is:"+root2);
}
else if(d == 0)
{
System.out.println("Roots are real and equal");
root1 = (-b+Math.sqrt(d))/(2*a);
System.out.println("Root:"+root1);
}
else
{
System.out.println("No real solutions. Roots are imaginary");
}
}
}
class Main {
public static void main(String[] args) {
Quad q= new Quad();
```

```
q.input();
q.discriminant();
q.calculateRoots();
}
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop>java Main
Quadratic equation is in the form : ax^2 + bx + c
Enter a:1
Enter b:2
Enter c:1
Roots are real and equal
Root:-1.0

C:\Users\Tushar\OneDrive\Desktop>
```

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 the percentage of a student.

```
import java.util.Scanner;
class Student{
   String usn;
   String name;
   int marks[]= new int[6];

   void Details()
   {
Scanner s=new Scanner(System.in);
System.out.println("Enter USN");
usn=s.next();
System.out.println("Enter Name");
name=s.next();
System.out.println("Enter marks for 6 subjects:");
for(int i = 0; i < 6; i++)
{
System.out.print("Subject " + (i + 1) + ": ");
marks[i]=s.nextInt();
}
   }
   double percentage()
   {
```

```
int total=0;
for(int i=0;i<6;i++)
{
total+=marks[i];
}
double p=total/6;
return p;
   }
   void display()
   {
System.out.println("\nStudent Details:");
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
        System.out.println("Marks:");
        for(int i = 0; i < 6; i++)
        {
System.out.println("Subject " + (i + 1) + ": " + marks[i]);
        }
        System.out.println("Percentage: "+ percentage() + "%");
   }
}
class Lab1student
{
public static void main(String args[]){
Scanner s = new Scanner(System.in);
System.out.print("Enter the number of students: ");
int n = s.nextInt();
Student[] students = new Student[n];
for (int i = 0; i < n; i++)
{
students[i] = new Student();
System.out.println("\nEnter details for Student " + (i + 1) + ":");
students[i].Details();
}
for (Student student : students)
{
student.display();
}
}
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop>javac student.java

C:\Users\Tushar\OneDrive\Desktop>java student
Enter number of students:
2

Enter details for Student1:
Enter USN:
1bm22cs111
Enter Name:
aaa
Enter marks of 6 Subjects
Subject1:
89
Subject2:
80
Subject3:
90
Subject4:
78
Subject5:
80
Subject6:
78

Enter details for Student2:
Enter USN:
1bm22cd33
Enter Name:
sss
Enter marks of 6 Subjects
Subject1:
1
Subject2:
70
Subject3:
90
Subject4:
80
Subject5:
90
Subject6:
70

Student Details:
USN:1bm22cs111
Nameaaa
Percentage:82.5%

Student Details:
USN:1bm22cd33
Namesss
Percentage:66.83333333333333%

C:\Users\Tushar\OneDrive\Desktop>
```

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.

```
import java.util.Scanner;
class Book{
String name;
String author;
int price;
int numpages;
Book(){}
Book(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 Books{
public static void main(String args[])
{Scanner s=new Scanner(System.in);
int n;
String name;
String author;
int price;
int numpages;
System.out.println("\nEnter number of books:");
n=s.nextInt();
Book b[];
b=new Book[n];
for(int i=0;i<n;i++)
{System.out.println("\nBook"+(i+1)+":");
System.out.println("Enter name of the book:");
name=s.next();
System.out.println("Enter author name:");
author=s.next();
System.out.println("Enter price:");
```

```
price=s.nextInt();
System.out.println("Enter number of pages:");
numpages=s.nextInt();
b[i]=new Book(name,author,price,numpages);;
}
for(int i=0;i<n;i++)
{System.out.println("\nBook"+(i+1)+"\n"+b[i]);}
}
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop>javac Books.java

C:\Users\Tushar\OneDrive\Desktop>java Books

Enter number of books:
2

Book1:
Enter name of the book:
abc
Enter author name:
qwe
Enter price:
123
Enter number of pages:
100

Book2:
Enter name of the book:
dfg
Enter author name:
xyz
Enter price:
456
Enter number of pages:
200

Book1
Book name:abc
Author name:qwe
Price:123
Number of pages:100


Book2
Book name:dfg
Author name:xyz
Price:456
Number of pages:200
```

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.

```
import java.util.Scanner;
abstract class shape {
int dim1;
int dim2;
shape(int dim1, int dim2) {
this.dim1 = dim1;
this.dim2 = dim2;
}
abstract void printArea();
}
class rectangle extends shape {
rectangle(int length, int breadth) {
super(length, breadth);
}
void printArea() {
double area = dim1 * dim2;
System.out.println("Area of rectangle = " + area);
}
}
class triangle extends shape {
triangle(int height, int base) {
super(height, base);
}
void printArea() {
double area = 0.5 * dim1 * dim2;
```

```
System.out.println("Area of triangle = " + area);
}
}
class circle extends shape {
circle(int radius) {
super(radius, 0);
}
void printArea() {
double area = Math.PI * dim1 * dim1;
System.out.println("Area of circle = " + area);
}
}
public class Abstract {
public static void main(String[] args) {
Scanner s = new Scanner(System.in);
System.out.println("Enter the length and breadth of the rectangle");
int l = s.nextInt();
int b = s.nextInt();
System.out.println("Enter base and height of the triangle");
int ba = s.nextInt();
int h = s.nextInt();
System.out.println("Enter the radius of the circle");
int r = s.nextInt();
rectangle re = new rectangle(l, b);
triangle t = new triangle(h, ba);
circle c = new circle(r);
re.printArea();
t.printArea();
c.printArea();
}
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop>javac Abstract.java

C:\Users\Tushar\OneDrive\Desktop>java Abstract
Enter the length and breadth of the rectangle
10
20
Enter base and height of the triangle
20
30
Enter the radius of the circle
7
Area of rectangle = 200.0
Area of triangle = 300.0
Area of circle = 153.93804002589985

C:\Users\Tushar\OneDrive\Desktop>
```

PROGRAM 5:

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called a 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 a 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 deposits from customers 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 a penalty if necessary and update the balance.

```
class Bank {
    public static void main(String[] args) {
        SavingsAccount savingsAccount = new SavingsAccount("John Doe", "SA1001");
        CurrentAccount currentAccount = new CurrentAccount("Jane Smith", "CA2002");
        savingsAccount.deposit(5000);
        savingsAccount.displayBalance();
        savingsAccount.computeInterest();
        savingsAccount.displayBalance();
        savingsAccount.withdraw(2000);
        savingsAccount.displayBalance();
        currentAccount.deposit(8000);
        currentAccount.displayBalance();
        currentAccount.withdraw(5000);
        currentAccount.displayBalance();
    }
}

class Account {
    protected String customerName;
    protected String accountNumber;
    protected double balance;

    public Account(String customerName, String accountNumber) {
        this.customerName = customerName;
        this.accountNumber = accountNumber;
        this.balance = 0;
    }
```

```
    public void deposit(double amount) {
        balance += amount;
        System.out.println("Deposit of " + amount + " successful");
    }

    public void displayBalance() {
        System.out.println("Account Number: " + accountNumber + "\nBalance: " + balance);
    }
}

class SavingsAccount extends Account {
    public SavingsAccount(String customerName, String accountNumber) {
        super(customerName, accountNumber);
    }

    public void computeInterest() {
        double interestRate = 0.05;
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("Interest of " + interest + " computed and added to the balance.");
    }

    public void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println("Withdrawal of " + amount + " successful");
        } else {
            System.out.println("Insufficient funds for withdrawal");
        }
    }
}
```

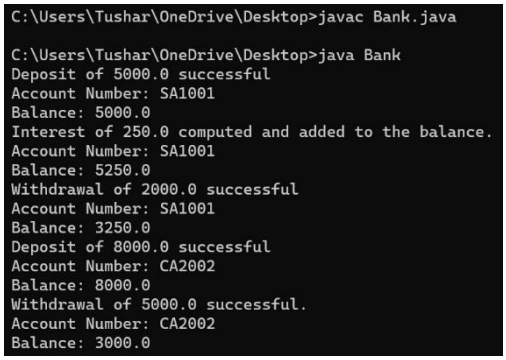
```
class CurrentAccount extends Account {
    private double minimumBalance = 1000;

    public CurrentAccount(String customerName, String accountNumber) {
        super(customerName, accountNumber);
    }

    public void withdraw(double amount) {
        if (balance - amount >= minimumBalance) {
            balance -= amount;
            System.out.println("Withdrawal of " + amount + " successful.");
        } else {
            System.out.println("Insufficient funds. Service charge applied");
            imposeServiceCharge();
        }
    }

    private void imposeServiceCharge() {
        double serviceCharge = 20;
        balance -= serviceCharge;
        System.out.println("Service charge of " + serviceCharge + " imposed.");
    }
}
```

OUTPUT:



PROGRAM 6:

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

```
package CIE;

public class students {
    public String usn;
    public String name;
    public int sem;

    public students(String usn, String name, int sem) {
        this.usn = usn;
        this.name = name;
        this.sem = sem;
    }
}
```

```
package CIE;

public class internals extends students {
    public int[] internalMarks;

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

```
}

package SEE;

import CIE.students;

public class externals extends students {
    public int[] seeMarks;

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

import java.util.Scanner;
import CIE.internals;
import SEE.externals;

public class packages {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of students: ");
        int n = scanner.nextInt();
        internals[] cieStudents = new internals[n];
        externals[] seeStudents = new externals[n];
        for (int i = 0; i < n; i++) {
            System.out.println("Enter details for CIE of student " + (i + 1));
            System.out.print("USN: ");
            String usn = scanner.next();
            System.out.print("Name: ");
            String name = scanner.next();
```

```
        System.out.print("Semester: ");
        int sem = scanner.nextInt();
        int[] cieMarks = new int[5];
        System.out.print("Enter CIE marks for 5 courses: ");
        for (int j = 0; j < 5; j++) {
            cieMarks[j] = scanner.nextInt();
        }
        cieStudents[i] = new internals(usn, name, sem, cieMarks);
    }
    for (int i = 0; i < n; i++) {
        System.out.println("Enter details for SEE of student " + (i + 1));
        System.out.print("USN: ");
        String usn = scanner.next();
        System.out.print("Name: ");
        String name = scanner.next();
        System.out.print("Semester: ");
        int sem = scanner.nextInt();
        int[] seeMarks = new int[5];
        System.out.print("Enter SEE marks for 5 courses: ");
        for (int j = 0; j < 5; j++) {
            seeMarks[j] = scanner.nextInt();
        }
        seeStudents[i] = new externals(usn, name, sem, seeMarks);
    }
    System.out.println("\nFinal Marks of Students:");
    for (int i = 0; i < n; i++) {
        System.out.println("\nDetails of Student " + (i + 1));
        System.out.println("USN: " + cieStudents[i].usn);
        System.out.println("Name: " + cieStudents[i].name);
        System.out.println("Semester: " + cieStudents[i].sem);
        System.out.println("CIE Marks: ");
        for (int j = 0; j < 5; j++) {
```

```
            System.out.print(cieStudents[i].internalMarks[j] + " ");
        }
        System.out.println("\nSEE Marks: ");
        for (int j = 0; j < 5; j++) {
            System.out.print(seeStudents[i].seeMarks[j] + " ");
        }
    }
  }
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop\New folder>javac Final.java

C:\Users\Tushar\OneDrive\Desktop\New folder>java Final
Enter the number of students: 1
Enter details for CIE of student 1
USN: 1bm22
Name: ab
Semester: 3
Enter CIE marks for 5 courses: 10
20
30
40
50
Enter details for SEE of student 1
USN: 1bm22
Name: ab
Semester: 3
Enter SEE marks for 5 courses: 60
80
90
80
90

Final Marks of Students:

Details of Student 1
USN: 1bm22
Name: ab
Semester: 3
CIE Marks:
0 0 0 0 0
SEE Marks:
60 80 90 80 90
```

PROGRAM 7:

Write a program that demonstrates handling of exceptions in inheritance trees. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >father’s age.

```
import java.util.Scanner;

class Father {
    int age;

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

class Son extends Father {
    int sonage;

    public Son(int age, int sonage) {
        super(age);
        if (sonage >= age) {
            throw new IllegalArgumentException("Son's age Cannot be greater than or equal to Father's age");
        }
        this.sonage = sonage;
    }
```

```
}

public class Main {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        try {
            System.out.println("Enter Father's Age:");
            int age = s.nextInt();
            System.out.println("Enter Son's Age:");
            int sonage = s.nextInt();
            Son sn = new Son(age, sonage);
            System.out.println("Father's Age:" + sn.age);
            System.out.println("Son's Age:" + sn.sonage);
        } catch (IllegalArgumentException e) {
            System.out.println("Exception Caught " + e.getMessage());
        }
        s.close();
    }
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop>javac Exception.java

C:\Users\Tushar\OneDrive\Desktop>java Exception
Enter Father's Age:
40
Enter Son's Age:
50
Exception Caught Son's age Cannot be greater than or equal to Father's age
```

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.

```
class DispMessage extends Thread {
    String message;
   int interval; // Interval in milliseconds

    public DispMessage(String message, int interval) {
        this.message = message;
        this.interval = interval;
    }

    public void run() {
        while (true) {
            System.out.println(message);
            try {
                Thread.sleep(interval);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
public class Main {
    public static void main(String[] args) {
    DispMessage bmsThread = new DispMessage("BMS College of Engineering", 10000);
    DispMessage cseThread = new DispMessage("CSE", 2000);

    bmsThread.start();
    cseThread.start();
    }
}
```

OUTPUT:

```
C:\Users\Tushar\OneDrive\Desktop>javac Multithread.java

C:\Users\Tushar\OneDrive\Desktop>java Multithread
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
```