

**B.M.S. COLLEGE OF ENGINEERING**  
Basavanagudi, Bengaluru- 560019  
**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**LAB REPORT**

On

*Object Oriented Java Programming*  
**(23CS3PCOOJ)**

Submitted By:

**ANAGHA BHARADWAJ**  
**1BM22CS038**

*In partial fulfilment of*  
**BACHELOR OF ENGINEERING**  
In  
**COMPUTER SCIENCE AND ENGINEERING**  
2023-24

Faculty-In-Charge  
**Swathi Sridharan**  
Assistant Professor  
Department of Computer Science and Engineering

SL. NO	DATE	TOPIC	PageNo
1	22/12/23	Quadratic Equation	1
2	29/12/23	Student SGPA Calculation	3
3	12/01/24	Book Details	6
4	12/01/24	Area Calculation	9
5	19/01/24	Bank Account Details	12
6	02/02/24	External And Internal Marks	18
7	16/02/24	Exception Handling	22
8	16/02/24	Threads	25

## LAB-1: QUADRATIC EQUATION

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;
public class squareroot {
    public static void main (String [ ] args) {
        Scanner s = new Scanner (System.in);
        System.out.println ("Enter the coefficients : ");
        int a = s.nextInt ();
        int b = s.nextInt ();
        int c = s.nextInt ();
        if (a == 0)
            System.out.println ("Enter valid value for a");
        else
            int d = b * b - 4 * a * c;
            if (d > 0)
                System.out.println ("Roots are real and distinct");
            float r1 = (float) (-b + Math.sqrt (d)) / (2 * a);
            float r2 = (float) (-b - Math.sqrt (d)) / (2 * a);
            System.out.println (r1);
            System.out.println (r2);
    }
}
```

```
else if (d < 0)
    System.out.println("Roots are
                        imaginary");
else
    System.out.println("Roots are
                        real and equal");
    float r = (float) -b / (2 * a);
    System.out.println(r);
```

## **OUTPUT :**

```
C:\Users\Admin\Desktop\1BM22CS038>javac SquareRoot.java

C:\Users\Admin\Desktop\1BM22CS038>java SquareRoot
1BM22CS038
Anagha Bharadwaj
Enter the coefficients:
1 -2 1
Roots are real and equal
Root is: 1.0

C:\Users\Admin\Desktop\1BM22CS038>java SquareRoot
1BM22CS038
Anagha Bharadwaj
Enter the coefficients:
1 -3 2
Roots are real and distinct
Roots are: 2.0 1.0

C:\Users\Admin\Desktop\1BM22CS038>java SquareRoot
1BM22CS038
Anagha Bharadwaj
Enter the coefficients:
1 1 1
Roots are imaginary
```

## LAB-2: STUDENT SGPA CALCULATOR

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

```
import java.util.Scanner;
public class student
{
    String usn;
    String name;
    private static int credit[] = {4, 4, 3, 3, 3, 1, 1, 1};
    private int numofsub = 8;
    int marks[] = new int[numofsub];
    Scanner s = new Scanner(System.in);
    student()
    {
        System.out.println("Enter your details below to calculate your SGPA \n");
        getDetails();
        getMarks();
        display();
    }
    public void getDetails()
    {
        System.out.println("Enter your USN:");
        usn = s.nextLine();
    }
```

```

System.out.println("Enter your name:");
name = s.nextLine();
}

public void setMarks()
{
    System.out.println("Enter your marks:");
    for (int i=0; i<numOfSub; i++)
    {
        marks[i] = s.nextInt();
    }
}

public double sgpa()
{
    double sgpa=0, temp=0;
    for (int i=0; i<numOfSub; i++)
    {
        if (marks[i] >= 40)
            if (marks[i] == 100)
                temp += credit[i] * 10;
            else
                temp += credit[i] * ((int)(marks[i] + 1));
        else
            temp += 0;
    }
}

```

```
sgpa = temp/20;
return sgpa;

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

public static void main(String args[])
{
    student s1 = new student();
}
```

#### OUTPUT :

```
C:\Users\Admin\Desktop\1BM22CS038>javac Sgpa.java

C:\Users\Admin\Desktop\1BM22CS038>java Sgpa
1BM22CS038
Anagha Bharadwaj
Enter your details below to calculate your SGPA

Enter your USN:
123
Enter your name:
anagha
Enter your marks in order
100 98 97 90 92 100 99 98
Name: anagha
USN: 123
SGPA: 10.0
```

## LAB-3: BOOK DETAILS

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.

The image shows handwritten Java code on lined paper. The code defines a class named 'book' with four members: name, author, price, and num\_pages. It includes a constructor to set these values and a self-details method to print them. The code uses Scanner for input and System.out.println for output.

```
import java.util.Scanner;
class book {
    String name;
    String author;
    float price;
    int num_pages;

    void self_details() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter the book name, author name, price and number of pages");
        name = s.next();
        author = s.next();
        price = s.nextFloat();
        num_pages = s.nextInt();
    }
}
```

```

    void get_details() {
        String details = toString();
        System.out.println(details);
    }

    word toString() {
        return "The book " + name + " is
        written by " + author + " consists of "
        + num_pages + " pages" and cost
        around " + price;
    }

public static void main(String args[]) {
    Scanner s = new Scanner(System.in);
    System.out.println("Enter the
    number of books");
    int n = s.nextInt();
    book b[] = new book[n];
    for (int i = 0; i < n; i++) {
        b[i] = new book();
        b[i].set_details();
    }
    for (int i = 0; i < n; i++)
        b[i].get_details();
}

```

**OUTPUT :**

```
C:\Users\Admin\Desktop\1BM22CS038>javac book.java
C:\Users\Admin\Desktop\1BM22CS038>java book
1BM22CS038
Anagha Bharadwaj
enter no of books you want to generate
2
enter bookname,author,price,num_pages
Abc
Mark_Howard
550
240
enter bookname,author,price,num_pages
Xyz
David_psyc
999
154
book details

the book Abc was written by Mark_Howard it consists of 240 pages and costs around 550.0
the book Xyz was written by David_psyc it consists of 154 pages and costs around 999.0
```

## LAB-4: AREA CALCULATION

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  
abstract class shape  
{  
    int a,b;  
    abstract void printArea();  
}  
  
class rectangle extends shape  
{  
    rectangle (int l, int b)  
    {  
        a=l; b=b;  
    }  
    void printArea()  
    {  
        System.out.println (" Area of  
rectangle : "+ a*b);  
    }  
}
```

```

class triangle extends shape
{
    int b, h;
    double a = b * h / 2;
    void printArea()
    {
        System.out.println("Area of
triangle: " + 0.5 * a);
    }
}

```

```

class circle extends shape
{
    int r;
    double a = r * r * 3.14;
    void printArea()
    {
        System.out.println("Area of
circle: " + 3.14 * a);
    }
}

```

```

class E
{
    public static void main(String[] args)
    {
        Scanner s = new Scanner(System.in);
    }
}

```

Page

System.out.println("Enter the length  
 and breadth of rectangle");  
 rectangle rec = new rectangle  
 (s.nextInt(), s.nextInt());  
 rec.printArea();  
 System.out.println("Enter the  
 base and height of triangle");  
 triangle tri = new triangle(s.nextInt(),  
 s.nextInt());  
 tri.printArea();  
 System.out.println("Enter the  
 radius of circle");  
 circle cr = new circle(s.nextInt());  
 cr.printArea();

#### OUTPUT :

```
C:\Users\Admin\Desktop\1BM22CS038>javac AreaCalculation.java
C:\Users\Admin\Desktop\1BM22CS038>java AreaCalculation
1BM22CS038
Anagha Bharadwaj
Enter length and breadth of a rectangle:
10 10
Area of rectangle: 100
Enter base and height of a triangle:
20 5
Area of the triangle: 50.0
Enter the radius of a circle:
15
Area of Circle: 706.5
```

## **LAB-5: BANK ACCOUNT DETAILS**

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- a) Accept deposit from customer and update the balance.
- b) Display the balance.
- c) Compute and deposit interest
- d) Permit withdrawal and update the balance

Check for the minimum balance, impose penalty if necessary and update the balance.

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

```
class Account {  
    String customername;  
    long accno;  
    String acctype;  
    double balance;
```

```
public Account (String customername,  
                long accno, String acctype)
```

```
{  
    this.customername = customername;  
    this.accno = accno;  
    this.acctype = acctype;  
    this.balance = 0.0  
}
```

```
public void displayBalance ()
```

```
{  
    System.out.println ("Customer  
Name " + customername);
```

```
    System.out.println ("Account  
Number " + accno);  
}
```

```
class CurAcc extends Acc
```

```

public void depositInterest()
{
    double interest = balance * interestRate;
    balance += interest;
    System.out.println("Interest deposited");
    In } Current balance : $" + balance);
}

public void compInterest()
{
    compInterest( double initialAmt,
    int term)
    {
        double compInterest = initialAmt
        * Math.pow(1 + interestRate), term)
        - initialAmt;
        balance += compInterest;
        System.out.println("Compound
        interest deposited. Current balance : "
        + balance);
    }
}

public class bank
{
    public static void main( String[] args)
    {
        Scanner p = new Scanner( System.in);
        System.out.println("Choose
        account type ");
        System.out.println(" 1. Current
        2. Savings ");
    }
}

```

Page 12

```

System.out.print("Enter choice (1 or 2)
int ch = s.nextInt();
System.out.print("Enter customer
name. Account number.");
String customerName = s.next();
long accno = s.nextLong();
if (ch == 1)
    CurAcc = curacc = new CurAcc
        (customerName, accno);
System.out.println("Enter
initial balance");
double initialBalance =
s.nextDouble();
curAcc.balance = initialBalance;
System.out.print("Enter with
drawal amt:");
double withdrawalAmount =
s.nextDouble();
curAcc.withdraw(withdrawalAmt);
curAcc.imposeServiceCharge();
curAcc.displayBalance();
}
else if (ch == 2)
    SavAcc = savacc = new SavAcc
        (customerName, accno);
System.out.print("Enter initial
balance");
double initialBalance = s.nextDouble();
savacc.balance = initialBalance;
System.out.print("Enter with
amt");
double withdrawalAmt = s.nextDouble();
    
```

```
System.out.print("Enter interest  
rate");  
double interestRate = s.nextDouble();  
savacc.interestRate = interestRate;  
savacc.displayBalance();  
System.out.print("Enter term for  
compound interest");  
int term = s.nextInt();  
savacc.compInterest(initialBalance,  
term);  
savacc.displayBalance();  
  
System.out.println("Invalid choice");
```

**OUTPUT :**

```
C:\Users\Admin\Desktop\1BM22CS038>javac bank.java

C:\Users\Admin\Desktop\1BM22CS038>java bank
1BM22CS038
Anagha Bharadwaj

Choose account type:
1. Current
2. Savings
Enter choice (1 or 2): 1
Enter customer name: anagha
Enter account number: 1232
Enter initial balance: 1000
Enter withdrawal amount: 300
Withdrawal successful. Current Balance: 700.0
Account Number: 1232
Customer Name: anagha
Account Type: Current
Balance: 700.0

C:\Users\Admin\Desktop\1BM22CS038>java bank
1BM22CS038
Anagha Bharadwaj

Choose account type:
1. Current
2. Savings
Enter choice (1 or 2): 2
Enter customer name: anagha
Enter account number: 1909
Enter initial balance: 1000
Enter withdrawal amount: $300
Withdrawal successful. Current Balance: 700.0
Enter interest rate: 5
Enter term (in years) for compound interest calculation: 5
Compound Interest deposited. Current Balance: 7775700.0
Account Number: 1909
Customer Name: anagha
Account Type: Savings
Balance: 7775700.0
```

## LAB-6: CALCULATION OF MARKS

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;
import java.util.Scanner;
public class Student {
    public int usn;
    public String name;
    public int sem;
    public void accept() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter
        usn, name, sem");
        usn = s.nextInt();
        name = s.nextLine();
        sem = s.nextInt();
    }
}
package CIE;
public class Internals {
    public int imarks[] = new int[5]
```

```

package SEE;
import CIF.student;

public class External extends student {
    public int marks[] = new
        int[5];
}

import java.util.Scanner;
import java.SEE.*;
import CIF.*;

public class Final-marks {
    public static void main(String args[]) {
        int fm[] = new int[5];
        Scanner s = new Scanner(System.in);
        System.out.println("Enter n : ");
        int n = s.nextInt();
        SEE.External st[] = new
            SEE.External[n];
        CIF.Internal si[] = new
            CIF.Internal[n];
        for (int i = 0; i < n; i++) {
            st[i] = new SEE.External();
            si[i] = new CIF.Internal();
            System.out.println("Enter
details " + (i + 1));
            st[i].Accept();
            for (int j = 0; j < 5; j++)

```

syro ("Enter im) and sm if  
sub "+(j+1));  
 $s[i].im[j] = sc.next();$   
 $st[i].sm[j] = sc.next();$   
 $fm[j] = s[i].im[j] + st[i].sm[j];$

syro ("Final marks of "  
+ st[i].name);  
for (int k=0; k<5; k++)  
syro ("Course "+(k+1)+"  
= " + fm[k]);

**OUTPUT :**

```
Name: Anagha Bharadwaj  
USN: 1BM22CS038
```

```
Enter n:
```

```
1
```

```
Enter details 1
```

```
Enter U, N, S:
```

```
12
```

```
Anagha
```

```
2
```

```
Enter im and sm of sub 1
```

```
95 95
```

```
Enter im and sm of sub 2
```

```
96 96
```

```
Enter im and sm of sub 3
```

```
97 97
```

```
Enter im and sm of sub 4
```

```
93 93
```

```
Enter im and sm of sub 5
```

```
92 92
```

```
Final marks of Anagha
```

```
Course 1 = 95
```

```
Course 2 = 96
```

```
Course 3 = 97
```

```
Course 4 = 93
```

```
Course 5 = 92
```

## LAB-7: EXCEPTION HANDLING

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.

```
import java.util.Scanner  
class Wrongage extends Exception  
{ public Wrongage (String message)  
{ super (message);  
}  
  
class Father  
{ int fage;  
public Father (int fage) throws  
Wrongage  
{ if (fage<0){ throw new  
Wrongage ("Age can't be negative");  
}  
this.fage = fage;  
}  
}
```

class son extends father  
 {  
 int fage;  
 public static son (int fage, int sage)  
 throws wrongage  
 {  
 super(father fage);  
 if (sage >= fage)  
 throw new wrongage  
 ("Son's age must be lesser  
 than father's age");  
 this.sage = sage;  
 }  
}  
class Main  
{  
public static void main (String args)  
Scanner s = new Scanner (System.in)  
System.out.println ("Enter father's  
age and son's age");  
int fage = s.nextInt();  
int sage = s.nextInt();  
try  
{  
son p = new son (fage, sage)  
System.out.println ("Father's  
age = " + s.fage + " Son's age = " + s.sage  
catch (wrongage e)  
System.out.println (e.getMessage());  
}

**OUTPUT :**

```
PS C:\Users\Admin\Desktop\1BM22CS038> javac AgeOfSonFather.java
PS C:\Users\Admin\Desktop\1BM22CS038> java AgeOfSonFather
Name:Anagha Bharadwaj
USN:1BM22CS038

Enter fathers and sons age: 40 15
Father's age: 40
Son's age: 15
PS C:\Users\Admin\Desktop\1BM22CS038> java AgeOfSonFather
Name:Anagha Bharadwaj
USN:1BM22CS038

Enter fathers and sons age: 20 30
Son's age must be lesser than fathers age
```

## LAB-8: MULTITHREADING

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 one extends Thread
{
    public void run()
    {
        int i=0;
        while(i<5)
        {
            try
            {
                System.out.println("Bms
College of Eng");
                Thread.sleep(10000);
            }
            catch(Exception e)
            {
                System.out.println(e.toString());
            }
        }
    }
}

class two extends Thread
{
    public void run()
    {
        int i=0;
```

Page 40

```

while (i < 5)
{
    i++;
    try
    {
        System.out.println("(" + i + ")");
        Thread.sleep(5000);
    }
    catch (InterruptedException e)
    {
        System.out.println("e.toString()");
    }
}

class Main
{
    public static void main(String[] args)
    {
        One f1 = new One();
        Two f2 = new Two();
        f1.start();
        f2.start();
        f1.join();
        f2.join();
    }
}

```

**OUTPUT :**

```
PS C:\Users\Admin\Desktop\1BM22CS038> javac Threads.java
PS C:\Users\Admin\Desktop\1BM22CS038> java Threads
Name:Anagha Bharadwaj
USN:1BM22CS038

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