

**VISVESVARAYA TECHNOLOGICAL
UNIVERSITY, JNANASANGAMA,
BELGAUM - 590014, KARNATAKA**



LABORATORY RECORD
ON
Object Oriented Java Programming
(23CS3PCOOJ) *Submitted by*
AMMAR YAZDANI (1BM22CS035)

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING



B. M. S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU – 560019
December-2022 to April-2023

INDEX

Sl. No.	Date	Experiment Title	Page No.
1	18.12.2023	Laboratory Program – 1	3-6
2	01.01.2024	Laboratory Program – 2	7-11
3	01.01.2024	Laboratory Program – 3	12-15
4	08.01.2024	Laboratory Program – 4	16-19
5	08.01.2024	Laboratory Program – 5	20-26
6	22.01.2024	Laboratory Program – 6	27-32
7	22.01.2024	Laboratory Program – 7	33-36
8	05.02.2024	Laboratory Program – 8	37-39
9	19.02.2024	Laboratory Program – 9	40-45

LABORATORY 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;
import java.lang.Math;
class Quad
{
    double Disc(double a,double b,double c)
    {
        return b*b-4*a*c;
    }
    void roots(double a,double b, double c)
    {
        double D = Disc(a,b,c);

        if (D<0)
        {
            double realPart = -b/(2*a);
            double imaginaryPart = Math.sqrt(Math.abs(D))/(2*a);
            System.out.println("The Quadratic Equation has Conjugate Imaginary
roots:");
            System.out.printf("Root 1: %.5f + %.5fi%n",realPart,imaginaryPart);
            System.out.printf("Root 2: %.5f - %.5fi%n",realPart,imaginaryPart);
        }
        else if (D>0)
        {
            System.out.println("The Quadratic Equation has Two Distinct Real Roots:");
            double r1=(-b+Math.sqrt(D))/(2*a);
            double r2=(-b-Math.sqrt(D))/(2*a);
            System.out.printf("Root 1: %.5f%n",r1);
            System.out.printf("Root 2: %.5f%n", r2);
        }
        else
        {
            System.out.println("The Quadratic Equation has Equal and Real Root:");
            double r1=(-b)/(2*a);
            System.out.printf("Both Root 1 and Root 2: %.5f%n",r1);
        }
    }

    class QuadEqn
    {
        public static void main(String sx[])
        {
        }
    }
}
```

```

{
    Scanner S1 = new Scanner(System.in);
    System.out.println("Enter the Coefficients of Quadratic Equation :");
    double a = S1.nextDouble();
    double b = S1.nextDouble();
    double c = S1.nextDouble();
    if (a==0)
    {
        System.out.println("Since the Coefficient of x^2 is Zero, it's not a Quadratic
Equation");
    }
    else
    {
        Quad quadratic=new Quad();
        quadratic.Disc(a,b,c);
        quadratic.roots(a,b,c);
    }
}
}

```

OUTPUT

```

D:\NotePad++\Java>javac QuadEqn.java

D:\NotePad++\Java>java QuadEqn
Enter the Coefficients of Quadratic Equation :
23
24
26
The Quadratic Equation has Conjugate Imaginary roots:
Root 1: -0.52174 + 0.92640i
Root 2: -0.52174 - 0.92640i

D:\NotePad++\Java>java QuadEqn
Enter the Coefficients of Quadratic Equation :
1
2
1
The Quadratic Equation has Equal and Real Root:
Both Root 1 and Root 2: -1.00000

D:\NotePad++\Java>java QuadEqn
Enter the Coefficients of Quadratic Equation :
6
8
0
The Quadratic Equation has Two Distinct Real Roots:
Root 1: 0.00000
Root 2: -1.33333

```

Q) Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read 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.

A) import java.util.Scanner;

```
public class QuadraticSolver {
```

```
    public static void main
```

```
(String[] args) {
```

```
    Scanner scanner = new Scanner
```

```
(System.in);
```

```
    System.out.println("Enter the  
coefficient for quadratic equation");
```

```
    System.out.println("Enter a:");
```

```
    double a = scanner.nextDouble();
```

```
    System.out.println("Enter b:");
```

```
    double b = scanner.nextDouble();
```

```
    System.out.println("Enter c:");
```

```
    double c = scanner.nextDouble();
```

```
    double discriminant = b * b - 4 * a * c;
```

```
    if (discriminant >= 0) {
```

```
        double root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
```

```
        double root2 = (b + Math.sqrt(discriminant)) / (2 * a);
```

```
        System.out.println("Real  
solutions:");
```

```
        System.out.println("Root1: " + root1);
```

```
        System.out.println("Root2: " + root2);
```

```
        else {  
            System.out.println ("No real  
            solutions");  
        }  
    } scanner.close();  
}
```

Output:-

Enter no-coefficient of quadratic equation
Enter a:

4

Enter no-coefficient of $\alpha \cdot b$:

6
Enter b+c:

2

The roots are -0.5 and -1.0

LABORATORY 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.

```
import java.util.Scanner;

class Student
{
    String usn;
    String name;
    int[] credits;
    int[] marks;

    public void acceptDetails() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter USN: ");
        usn = sc.nextLine();
        System.out.println("Enter Name: ");
        name = sc.nextLine();
        System.out.println("Enter number of subjects: ");
        int n = sc.nextInt();
        credits = new int[n];
        marks = new int[n];
        for (int i = 0; i < n; i++) {
            System.out.println("Enter credits for subject " + (i + 1) + ": ");
            credits[i] = sc.nextInt();
            System.out.println("Enter marks for subject " + (i + 1) + ": ");
            marks[i] = sc.nextInt();
        }
    }

    public void displayDetails() {
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
        System.out.println("Marks: ");
        for (int i = 0; i < marks.length; i++) {
            System.out.println("Subject " + (i + 1) + ": " + marks[i]);
        }
        System.out.println("Credits: ");
        for (int i = 0; i < credits.length; i++) {
            System.out.println("Subject " + (i + 1) + ": " + credits[i]);
        }
    }

    public double calculateSGPA() {
```

```

double totalGrade = 0;
int totalCredit = 0;
for (int i = 0; i < credits.length; i++) {
    totalGrade += getGrade(marks[i]) * credits[i];
    totalCredit += credits[i];
}
return totalGrade / totalCredit;
}

private double getGrade(int marks) {
    if (marks >= 90) {
        return 10;
    } else if (marks >= 80) {
        return 9;
    } else if (marks >= 70) {
        return 8;
    } else if (marks >= 60) {
        return 7;
    } else if (marks >= 50) {
        return 6;
    } else if (marks >= 40) {
        return 5;
    } else {
        return 0;
    }
}

public static void main(String[] args) {
    Student student = new Student();
    student.acceptDetails();
    student.displayDetails();
    System.out.println("SGPA: " + student.calculateSGPA());
}
}

```

OUTPUT

```
D:\NotePad++\Java>javac Student.java

D:\NotePad++\Java>java Student
Enter USN:
035
Enter Name:
Ammar
Enter number of subjects:
3
Enter credits for subject 1:
4
Enter marks for subject 1:
97
Enter credits for subject 2:
3
Enter marks for subject 2:
98
Enter credits for subject 3:
2
Enter marks for subject 3:
99
USN: 035
Name: Ammar
Marks:
Subject 1: 97
Subject 2: 98
Subject 3: 99
Credits:
Subject 1: 4
Subject 2: 3
Subject 3: 2
SGPA: 10.0
```

Lab - 2

- Q) 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.

① import java.util.*;

```
class student {  
    String name, usn;  
    int credits[];  
    double marks[];  
    int m, i;  
    float total = 0;  
    int totalcredits = 0;
```

```
Scanner sc = new Scanner(System.in);  
void input() {
```

```
System.out.println("Enter your name");
```

name =

System.out

n = sc.

marks =

credits =

for(int i

System.out

marks'

marks[

System.out

credits

credits[

calc(m

}

}

void

int

totalc

```

create
and
and
of a
stem::)
    name = sc. nextLine();
    System.out.println("Enter number of subjects");
    n = sc.nextInt();
    marks = new double[n];
    credits = new int[n];
    for(int i=0; i<n; i++) {
        System.out.println("Enter " + (i+1) + " Subject marks");
        marks[i] = sc.nextDouble();
        System.out.println("Enter " + (i+1) + " Subject credits");
        credits[i] = sc.nextInt();
        calc(marks[i], credits[i], i);
    }
}

void calc(double marks, int credits,
        int i) {
    totalCredits = totalCredits + credits;
}

```

```

if (marks >= 90 && marks <= 100)
    total = total + (10 * credits);
else if (marks >= 80 && marks < 90)
    total = total + (9 * credits);
else if (marks >= 70 && marks < 80)
    total = total + (8 * credits);
else if (marks >= 60 && marks < 70)
    total = total + (7 * credits);
else if (marks >= 50 && marks < 60)
    total = total + (6 * credits);
else if (marks >= 40 && marks < 50)
    total = total + (5 * credits);
else {
    System.out.println("Subject " + (i+1) +
        " Failed");
}
}

```

```

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

class Student {
    public static void main(String args[]) {
        Student s1 = new Student();
        s1.input();
        s1.display();
    }
}

```

Mark

Output:-

Enter USN:
035
Enter Name:
Ammar
Enter number of subjects:
3
Enter credits for subject 1:
4
Enter marks for subject 1:
97
Enter credits for subject 2:
3
Enter marks for subject 2:
98
Enter credits for subject 3:
2
Enter marks for subject 3:
99
USN: 035
Name: Ammar

LABORATORY 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;
    double price;
    int num_pages;

    public Book(String name, String author, double price, int num_pages)
    {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }

    public void setName(String name)
    {
        this.name = name;
    }

    public String getName()
    {
        return name;
    }

    public void setAuthor(String author)
    {
        this.author = author;
    }

    public String getAuthor()
    {
        return author;
    }

    public void setPrice(double price)
    {
```

```

        this.price = price;
    }

    public double getPrice()
    {
        return price;
    }

    public void setNumPages(int num_pages)
    {
        this.num_pages = num_pages;
    }

    public int getNumPages()
    {
        return num_pages;
    }

    public String toString()
    {
        return "Name: " + name + "\nAuthor: " + author + "\nPrice: " + price + "\nNumber of Pages: " +
num_pages;
    }

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of books: ");
        int n = sc.nextInt();
        Book[] books = new Book[n];
        for (int i = 0; i < n; i++){
            System.out.println("Enter details for book " + (i + 1) + ":" );
            System.out.println("Enter name: ");
            String name = sc.next();
            System.out.println("Enter author: ");
            String author = sc.next();
            System.out.println("Enter price: ");
            double price = sc.nextDouble();
            System.out.println("Enter number of pages: ");
            int num_pages = sc.nextInt();
            books[i] = new Book(name, author, price, num_pages);
        }
        for (int i = 0; i < n; i++){
            System.out.println("Details of book " + (i + 1) + ":" );
            System.out.println(books[i].toString());
        }
    }
}

```

OUTPUT

```
D:\NotePad++\Java>java Book
Enter the number of books:
2
Enter details for book 1:
Enter name:
Rog
Enter author:
Zephyrus
Enter price:
560
Enter number of pages:
1536
Enter details for book 2:
Enter name:
Power
Enter author:
Murphy
Enter price:
399
Enter number of pages:
256
Details of book 1:
Name: Rog
Author: Zephyrus
Price: 560.0
Number of Pages: 1536
Details of book 2:
Name: Power
Author: Murphy
Price: 399.0
Number of Pages: 256
```

Lab - 3

Program : 3 :-

```

import java.util.*;
class books {
    Scanner sc = new Scanner (System.in);
    String name, author;
    int price, num_pages;
    books () {
        }
        books (String name, String author,
               int price, int num_pages) {
            this.name = name;
            this.author = author;
            this.price = price;
            this.num_pages = num_pages;
        }
}

```

```

void getdata () {
    System.out.println ("Enter the name
of the book");
    name = sc.nextLine ();
    System.out.println ("Enter the name
of the author");
    author = sc.nextLine ();
    System.out.println ("Enter the price");
    price = sc.nextInt ();
    System.out.println ("Enter the number
of pages");
    num_pages = sc.nextInt ();
}

public String toString () {
    return ("Name : " + name + "\n Author :
" + author + "\n Price : " + price + "\n
Number of pages : " + num_pages);
}

```

Class book details

```

class book_details {
    public static void main (String
args[])
    {
        Scanner sc = new Scanner (System.in);
        books b1 = new
books ("Mehta", "neil", 56, 154);
        System.out.println ("Constructor
values: \n" + b1);
        System.out.println ("Enter the
number of object of books");
        int m = sc.nextInt ();
        books s[] = new books [m];
        for (int i = 0; i < m; i++)
        {
            s[i] = new books ();
            System.out.println ("Enter the
details of " + (i+1) + " book");
        }
    }
}

```

```

s[i].getdata ();
for (int i = 0; i < m; i++)
{
    System.out.println ("In Details of
the book " + (i+1));
    System.out.println (s[i]);
}

```

Output:-

Constructor values :

Name : Mehta
Author : neil
Price : 56
Number of pages : 154

Enter the number of object of books

Enter details of 1 book

Enter the name of the author book

abc

Enter the name of the author

Xyz.

LABORATORY 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
{
    abstract void printArea();
    int length,breadth;
}

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

    void printArea()
    {
        int area=length*breadth;
        System.out.println("Area of Rectangle is "+area);
    }
}

class Triangle extends Shape
{
    Triangle(int l,int b)
    {
        length=l;
        breadth=b;
    }

    void printArea()
    {
        double area=0.5*length*breadth;
        System.out.println("Area of Triangle is "+area);
    }
}

class Circle extends Shape
```

```

{
    Circle(int r)
    {
        length=r;
    }

    void printArea()
    {
        double area=3.14*length*length;
        System.out.println("Area of Circle is "+area);
    }
}

class Display
{
    public static void main(String sx[])
    {
        Scanner s1=new Scanner(System.in);
        Rectangle r1=new Rectangle(0,0);
        System.out.println("Enter the Length and Breadth of Rectangle, to get it's Area: ");
        int l=s1.nextInt();
        int b=s1.nextInt();
        r1=new Rectangle(l,b);
        r1.printArea();
        Triangle t1=new Triangle(0,0);
        System.out.println("Enter the base and height of triangle, to get it's Area: ");
        int bs=s1.nextInt();
        int h=s1.nextInt();
        t1=new Triangle(bs,h);
        t1.printArea();
        Circle c1=new Circle(0);
        System.out.println("Enter the Radius of Circle, to get it's Area: ");
        int r=s1.nextInt();
        c1=new Circle(r);
        c1.printArea();
    }
}

```

OUTPUT

```
D:\NotePad++\Java>javac Shape.java

D:\NotePad++\Java>java Display
Enter the Length and Breadth of Rectangle, to get it's Area:
2
3
Area of Rectangle is 6
Enter the base and height of triangle, to get it's Area:
2
4
Area of Triangle is 4.0
Enter the Radius of Circle, to get it's Area:
2
Area of Circle is 12.56
```

Program - 4

```
import java.util.*;
abstract class shape {
    double a, b;
    abstract void printarea();
}

class triangle extends shape {
    void getdata(double x, double y) {
        a = x; b = y;
    }
    void printarea() {
        double area = 0.5 * a * b;
        System.out.println("Area of Triangle = " + area);
    }
}

class rectangle extends shape {
    void getdata(double x, double y) {
        a = x; b = y;
    }
}
```

```
void printarea () {
    double area = a * b;
    System.out.println("Area of rectangle = " + area);
}

class circle extends shape {
    void getdata (double x) {
        a = x;
    }
    void printarea () {
        double area = 3.142 * a * a;
        System.out.println("Area of circle = " + area);
    }
}

class abstract { public
static void main (String args[]) {
    int ch;
    shape si;
    Scanner sc = new Scanner (System.in);
```

rectangle r = new rectangle();
 triangle t = new triangle();
 circle c = new circle();
 System.out.println("1. Area of
 rectangle\n2. Area of triangle\n3. Area
 of circle\nEnter your choice");
 ch = sc.nextInt();
 switch(ch){
 case 1:
 System.out.println
 ("Enter length and breadth:");
 double l = sc.nextDouble();
 double b = sc.nextDouble();
 r.setdata(l, b);
 r.printarea();
 break;
 case 2:
 System.out.println
 ("Enter base and height:");
 double b1 = sc.nextDouble();
 double h = sc.nextDouble();

l. get data (b1, h);
l. printarea ();
break;

Case 3: System.out.println ("Enter
length and breadth: "); radius: ");
double r1 = sc.nextDouble ();
c. getdata (~~r1~~ r1);
c. printarea ();
break;
default: System.out.println
("Invalid input");
}
}

LABORATORY 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.

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 accountNumber;
    String accountType;
    double balance;

    public Account(String customerName, long accountNumber, String accountType, double balance)
    {
        this.customerName = customerName;
        this.accountNumber = accountNumber;
        this.accountType = accountType;
        this.balance = balance;
    }

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

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

```

        System.out.println("Customer Name: " + customerName);
        System.out.println("Account Type: " + accountType);
        System.out.println("Balance: " + balance);
    }
}

class SavAcct extends Account
{
    public SavAcct(String customerName, long accountNumber, double balance)
    {
        super(customerName, accountNumber, "Savings", balance);
    }

    public void computeAndDepositInterest(double rate)
    {
        double interest = balance * rate / 100;
        balance += interest;
        System.out.println("Interest computed and deposited. Updated balance: " + balance);
    }

    public void withdraw(double amount)
    {
        if (amount <= balance)
        {
            balance -= amount;
            System.out.println("Withdrawal successful. Updated balance: " + balance);
        }
        else
        {
            System.out.println("Insufficient funds. Withdrawal failed.");
        }
    }
}

class CurrAcct extends Account
{
    double minimumBalance;
    double serviceCharge;

    public CurrAcct(String customerName, long accountNumber, double balance, double minimumBalance, double serviceCharge)
    {
        super(customerName, accountNumber, "Current", balance);
        this.minimumBalance = minimumBalance;
        this.serviceCharge = serviceCharge;
    }

    private void checkMinimumBalance()

```

```

        {
        if (balance < minimumBalance)
            {
            balance -= serviceCharge;
            System.out.println("Minimum balance not maintained. Service charge imposed. Updated
balance: " + balance);
            }
        }

public void withdraw(double amount)
{
    if (amount <= balance)
        {
        balance -= amount;
        System.out.println("Withdrawal successful. Updated balance: " + balance);
        checkMinimumBalance();
        }
    else
        {
        System.out.println("Insufficient funds. Withdrawal failed.");
        }
}
}

public class Bank
{
    public static void main(String[] args)
    {
        Scanner s1 = new Scanner(System.in);

        System.out.print("Enter customer name for Savings Account: ");
        String SCN = s1.nextLine();
        System.out.print("Enter account number for Savings Account: ");
        long SAN = s1.nextLong();
        System.out.print("Enter initial balance for Savings Account: ");
        double SIB = s1.nextDouble();
        SavAcct SA = new SavAcct(SCN, SAN, SIB);

        System.out.print("Enter customer name for Current Account: ");
        String CCN = s1.next();
        System.out.print("Enter account number for Current Account: ");
        long CAN = s1.nextLong();
        System.out.print("Enter initial balance for Current Account: ");
        double CIB = s1.nextDouble();
        System.out.print("Enter minimum balance for Current Account: ");
        double MB = s1.nextDouble();
        System.out.print("Enter service charge for Current Account: ");
        double SC = s1.nextDouble();
    }
}

```

```
CurrAcct CA = new CurrAcct(CCN, CAN, CIB, MB, SC);

System.out.print("Enter deposit amount for Savings Account: ");
double SDA = s1.nextDouble();
SA.deposit(SDA);

System.out.print("Enter interest rate for Savings Account: ");
double SIR = s1.nextDouble();
SA.computeAndDepositInterest(SIR);

System.out.print("Enter withdrawal amount for Savings Account: ");
double SWA = s1.nextDouble();
SA.withdraw(SWA);

System.out.print("Enter deposit amount for Current Account: ");
double CDA = s1.nextDouble();
CA.deposit(CDA);

System.out.print("Enter withdrawal amount for Current Account: ");
double CWA = s1.nextDouble();
CA.withdraw(CWA);

System.out.println("\nFinal Balances:");
System.out.println("Savings Account:");
SA.displayBalance();

System.out.println("\nCurrent Account:");
CA.displayBalance();

}
```

OUTPUT

```
D:\NotePad++\Java>javac Bank.java
```

```
D:\NotePad++\Java>java Bank
Enter customer name for Savings Account:
Enter account number for Savings Account: 2324
Enter initial balance for Savings Account: 5000
Enter customer name for Current Account: Ammar
Enter account number for Current Account: 2324
Enter initial balance for Current Account: 6000
Enter minimum balance for Current Account: 1000
Enter service charge for Current Account: 100
Enter deposit amount for Savings Account: 2000
Deposit successful. Updated balance: 7000.0
Enter interest rate for Savings Account: 2
Interest computed and deposited. Updated balance: 7140.0
Enter withdrawal amount for Savings Account: 500
Withdrawal successful. Updated balance: 6640.0
Enter deposit amount for Current Account: 1000
Deposit successful. Updated balance: 7000.0
Enter withdrawal amount for Current Account: 750
Withdrawal successful. Updated balance: 6250.0
```

Final Balances:

Savings Account:
Account Number: 2324
Customer Name: Ammar
Account Type: Savings
Balance: 6640.0

Current Account:

Account Number: 2324
Customer Name: Ammar
Account Type: Current
Balance: 6250.0

Program-2
 import java.util.*;
 import java.lang.Math;
 class bank {
 Scanner sc = new Scanner(
 System.in);
 String name;
 int acc_no;
 float bal, si;
 void accept() {
 System.out.println("Enter your name");
 name = sc.nextLine();
 System.out.println("Enter the
 balance amount");
 bal = sc.nextFloat();
 }

bal = bal - service-fee;
 System.out.println("Sc no is " +
 "as service fee");
 }
 System.out.println("withdraw
 amount");
 System.out.println("Current
 balance : " + bal);
 }
 }
 }
 class savings extends bank {
 void cheque() {
 System.out.println("Cheque
 service not available");
 }
 void withdrawal() {
 float amt;

System.out.println("Current
 balance : " + bal);
 }
 }
 }
 class current extends bank {
 int service-fee = 50;
 void cheque() {
 System.out.println("Enter
 the amount to be withdrawn");
 amt = sc.nextFloat();
 if (amt > bal)
 System.out.println("Balance
 insufficient");
 else {
 bal = bal - amt;
 if (bal < 100) {

System.out.println("Enter the
 number of time interest applied");
 int n = sc.nextInt();
 System.out.println("Enter the
 time elapsed");
 int t = sc.nextInt();
 si = bal * (1 + (r/n));
 System.out.println("Compound
 interest is " + (Math.pow(si, n*t)));
 }
 public class account {
 public static void main
 (String args[]) {
 Scanner sc = new Scanner
 (System.in);

```
savings obj1 = new savings();  
current obj2 = new current();  
System.out.println("m1. Savings  
account \n2. Current account");  
int choice = sc.nextInt();  
switch (choice) {
```

case 1:

```
    obj1.accept();  
    obj1.display();  
    obj1.cheque();  
    obj1.deposit();  
    obj1.interest();  
    obj1.withdrawal();
```

break;

case 2:

```
    obj2.accept();  
    obj2.display();  
    obj2.cho...();
```

Output:-

1. Savings account

2. Current account

2.

Enter your name

Rahul

Enter the balance amount

10

Name: Rahul

Cheque service available

Do you want to deposit (1 for yes,
2 for no)

1

Enter the amount to be deposited

1000

Amount in bank insufficient

Current balance : 10.0

Enter the amount to be withdrawn
5

LABORATORY 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 which is a derived class of Student and 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 Internals extends Student
{
    public int[] internalMarks;

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

package CIE;

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

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

package SEE;
import CIE.Student;

public class External extends Student
{
    public int[] seeMarks;
```

```

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

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

public class CalculateFinalMarks
{
    public static void main(String[] args)
    {
        Scanner s1 = new Scanner(System.in);

        System.out.println("Enter the number of students:");
        int n = s1.nextInt();

        Internals[] CS = new Internals[n];
        for (int i = 0; i < n; i++)
        {
            System.out.println("Enter details for CIE student " + (i + 1));
            System.out.print("USN: ");
            String usn = s1.next();
            System.out.print("Name: ");
            String name = s1.next();
            System.out.print("Semester: ");
            int sem = s1.nextInt();
            System.out.println("Enter internal marks for 5 courses:");
            int[] internalMarks = new int[5];
            for (int j = 0; j < 5; j++)
            {
                System.out.print("Course " + (j + 1) + ": ");
                internalMarks[j] = s1.nextInt();
            }
            CS[i] = new Internals(usn, name, sem, internalMarks);
        }

        External[] SS = new External[n];
        for (int i = 0; i < n; i++)
        {
            System.out.println("Enter details for SEE student " + (i + 1));
            System.out.print("USN: ");

```

```

String usn = s1.next();
System.out.print("Name: ");
String name = s1.next();
System.out.print("Semester: ");
int sem = s1.nextInt();
System.out.println("Enter SEE marks for 5 courses:");
int[] seeMarks = new int[5];
for (int j = 0; j < 5; j++)
{
    System.out.print("Course " + (j + 1) + ": ");
    seeMarks[j] = s1.nextInt();
}

SS[i] = new External(usn, name, sem, seeMarks);
}

int[][] finalMarks = new int[n][5];
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < 5; j++)
    {
        finalMarks[i][j] = CS[i].internalMarks[j] + SS[i].seeMarks[j];
    }
}

System.out.println("\nFinal Marks:");
for (int i = 0; i < n; i++)
{
    System.out.print("USN: " + CS[i].usn + ", Name: " + CS[i].name + ", Semester: " + CS[i].sem
    + ", Final Marks: ");
    for (int j = 0; j < 5; j++)
    {
        System.out.print(finalMarks[i][j] + " ");
    }
    System.out.println();
}
}
}

```

OUTPUT

```
D:\NotePad++\Java\Packages>javac CalculateFinalMarks.java

D:\NotePad++\Java\Packages>java CalculateFinalMarks
Enter the number of students:
1
Enter details for CIE student 1
USN: 035
Name: Ammar
Semester: 3
Enter internal marks for 5 courses:
Course 1: 47
Course 2: 48
Course 3: 49
Course 4: 50
Course 5: 49
Enter details for SEE student 1
USN: 035
Name: Ammar
Semester: 3
Enter SEE marks for 5 courses:
Course 1: 48
Course 2: 49
Course 3: 47
Course 4: 50
Course 5: 50

Final Marks:
USN:035 Name: Ammar Semester: 3, Final Marks: 95 97 96 100 99
```

Lab-65 Packages in Java

Package CIE:

```

import java.util.*;
public class internals extends
CIE.student {
    Scanner sc = new Scanner(
        System.in);
    public int clem[] = new int[5];
    public void accept() {
        int i;
        for (i=0; i<5; i++) {
            System.out.println("Enter
CIE marks of subject " + (i+1));
            clem[i] = sc.nextInt();
        }
    }
}

```

Package SEF;

```

import CIE.*;
import java.util.*;
public class externals extends
CIE.student {
    Scanner sc = new Scanner(
        System.in);
    public int seem[] = new int[5];
    public void accept() {
        for (int i=0; i<5; i++) {
            System.out.println("Enter
SEF marks of subject " + (i+1));
            seem[i] = sc.nextInt();
        }
    }
}

```

```

import CIE.*;
import SEF.*;
import java.util.*;
class total {
    public static void main(String
args[]) {
        int i, j, m;
        Scanner sc = new Scanner(System
.in);
        int total[] = new int[5];
        System.out.println("Enter
number of students:");
        m = sc.nextInt();
        CIE.student s[] = new
        CIE.student[m];
        CIE.internals ci[] = new
        CIE.internals[m];
        SEF.externals se[] = new
        SEF.externals[m];

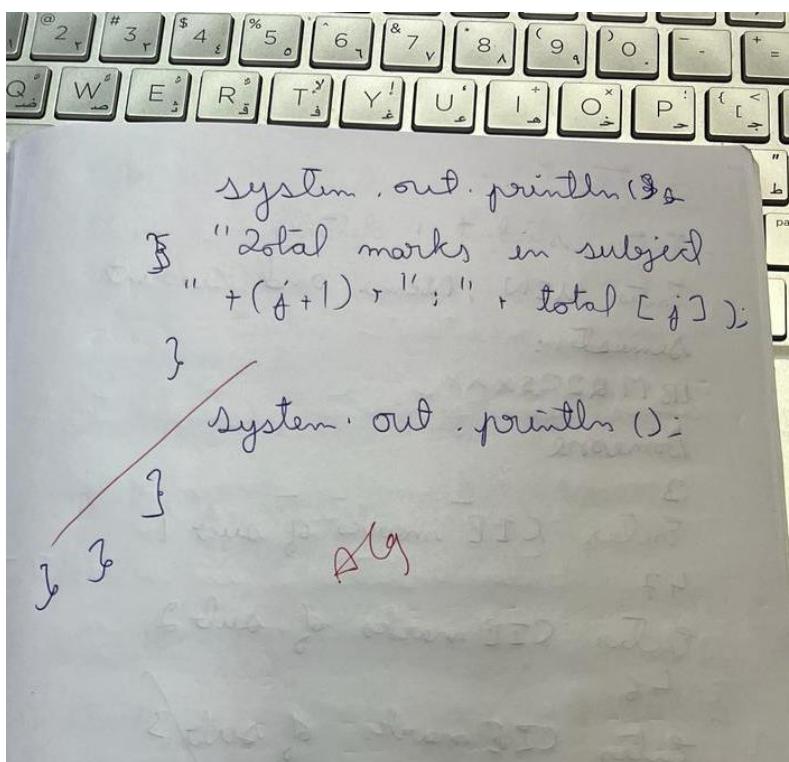
```

```

for (i=0; i<n; i++)
{
    system.out.println("Enter student "+(i+1)+" details");
    S[i] = new CIE.student();
    S[i].accept();
    ci[i] = new CIE.internal();
    ci[i].accept();
    se[i] = new SEE.external();
    se[i].accept();
}

for (i=0; i<n; i++)
{
    system.out.println("Details of student "+(i+1));
    S[i].display();
    for (j=0; j<S>.j++;)
    {
        total[j] = ci[i].seem[j];
        E[i].seem[j];
    }
}

```



Output :-

Enter student 1 details
Enter USN , Name and current
Semester:
1B M Q2CSXX
Someone
3
Enter LIE marks of sub 1
47
Enter CIE marks of sub 2
46
Enter CIE marks of sub 3
45
Enter SEE marks of sub 1
48
Enter SEE marks of sub 2
47
Enter SEE marks of sub 3
44
Details of student 1

LABORATORY 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.

```
import java.util.Scanner;

class WrongAge extends Exception
{
    public WrongAge()
    {
        super("Invalid age! Age cannot be negative nor zero.");
    }

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

class Father
{
    private int age;

    public Father(int age) throws WrongAge
    {
        if (age <= 0)
        {
            throw new WrongAge();
        }
        this.age = age;
    }

    public int getAge()
    {
        return age;
    }
}

class Son extends Father
{
    private int sonAge;
```

```

public Son(int fatherAge, int sonAge) throws WrongAge
{
    super(fatherAge);

    if (sonAge >= fatherAge)
    {
        throw new WrongAge("Son's age should be less than Father's age.");
    }

    this.sonAge = sonAge;
}

public int getSonAge()
{
    return sonAge;
}

public class InheritanceException
{
    public static void main(String[] args)
    {
        try
        {
            Scanner s1 = new Scanner(System.in);

            System.out.print("Enter Father's age: ");
            int fatherAge = s1.nextInt();
            Father f = new Father(fatherAge);

            System.out.print("Enter Son's age: ");
            int sonAge = s1.nextInt();
            Son s = new Son(f.getAge(), sonAge);

            System.out.println("Father's age: " + f.getAge());
            System.out.println("Son's age: " + s.getSonAge());

        } catch (WrongAge e)
        {
            System.out.println("Exception: " + e);
        } catch (Exception e)
        {
            System.out.println("Exception: Invalid input. Please enter valid integer values.");
        }
    }
}

```

OUTPUT

```
D:\NotePad++\Java>javac InheritanceException.java

D:\NotePad++\Java>java InheritanceException
Enter Father's age: 46
Enter Son's age: 56
Exception: WrongAge: Son's age should be less than Father's age.

D:\NotePad++\Java>java InheritanceException
Enter Father's age: 56
Enter Son's age: 24
Father's age: 56
Son's age: 24
```

Lab - 6

Exception in Java:-

```
import java.util.*;
class fatherAgeException extends Exception
{
    public String toString()
    {
        return ("Father's age is less than
        0");
    }
}
class sonAgeException extends
Exception
{
    int a;
    sonAgeException (int age)
    {
        a=age;
    }
    public String toString()
    {
        return ("Father's age is
        less than 0");
    }
}
```

less than 0");
}

class sonAgeException extends
Exception
{
 if (a<0)
 return ("Son's age is less
 than 0");
 else
 return ("Son's age is more
 than father's age");
}

class Father
{
 int age;
 Scanner in = new Scanner
 (System.in);
 Father ()
 {

```
    system.out.println("Enter the age  
of son:");  
    age = in.nextInt();  
}
```

```
void ex1() throws FatherAgeException
```

```
{ if (age < 0)  
    throw new FatherAgeException();
```

```
}
```

```
class Son extends Father {  
    int age;
```

```
Son(){
```

```
    System.out.println("Enter the  
age of son:");
```

```
    age = in.nextInt();
```

```
}
```

```
void ex2() throws SonAgeException {  
    if (age < 0 || age > super.age){  
        throw new SonAgeException(age);  
    }
```

```
}
```

```
public class except {
```

```
    public static void main(String[]  
args) {
```

```
        Son s = new Son();
```

```
        try {
```

```
            s.ex1();
```

```
}
```

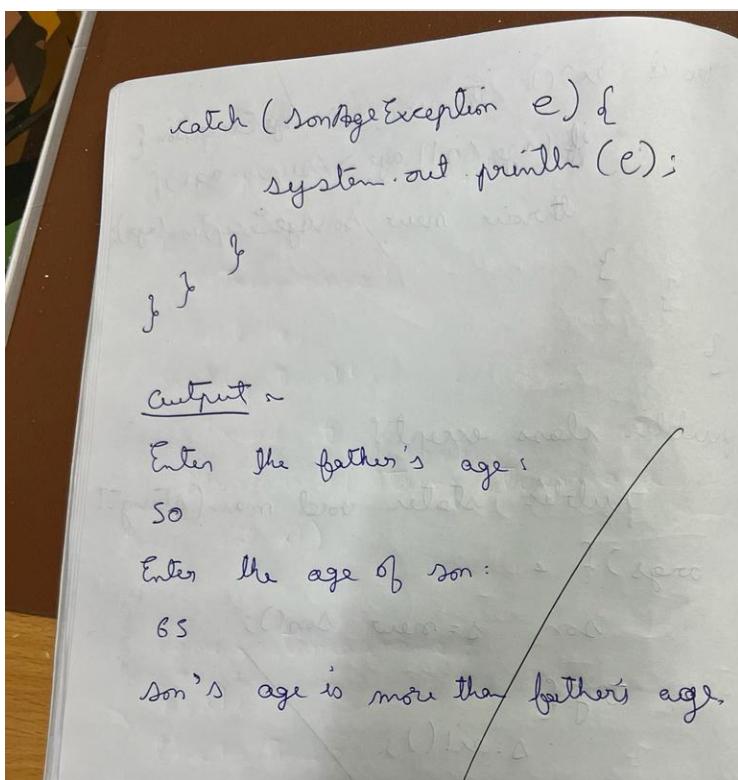
```
        catch (FatherAgeException e) {  
            System.out.println(e);
```

```
}
```

```
        try {
```

```
            s.ex2();
```

```
}
```



LABORATORY 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 BMSThread implements Runnable
{
    public void run()
    {
        while (true)
        {
            try
            {
                System.out.println("BMS College of Engineering");
                Thread.sleep(10000);
            }
            catch (InterruptedException ie)
            {
                System.out.println("BMSThread is Interrupted");
            }
        }
    }
}

class CSEThread implements Runnable
{
    public void run()
    {
        while (true)
        {
            try
            {
                System.out.println("CSE");
                Thread.sleep(2000);
            }
            catch (InterruptedException ie)
            {
                System.out.println("CSEThread is Interrupted");
            }
        }
    }
}

public class Display
{
    public static void main(String[] args)
```

```
{  
    Thread bms = new Thread(new BMSThread());  
    Thread cse = new Thread(new CSEThread());  
    bms.start();  
    cse.start();  
}  
}
```

OUTPUT

```
D:\NotePad++\Java>javac Display.java
```

```
D:\NotePad++\Java>java Display  
BMS College of Engineering
```

```
CSE
```

```
BMS College of Engineering
```

```
CSE
```

```
BMS College of Engineering
```

```
CSE
```

```
CSE
```

```
|
```

Threading :-

```

class NewThread implements Runnable
{
    string name;
    int x;
    long time1;
    Thread t;
    NewThread (string threadname, long
    time, int x1)
    {
        name = threadname;
        x = x1;
        time1 = time;
        t = new Thread (this, name);
        t.start();
    }
    public void run ()
    {
        try {
            for (int i=x; i>0; i--)
            {
                System.out.println (name);
            }
        }
    }
}

```

```

Thread.sleep (time1);
}
}
catch (InterruptedException e)
{
    System.out.println (name + "Interrupted");
}
}
class Threading
{
    public static void main (String
    args[])
    {
        new NewThread ("BMS College of
        engineering", 10000, 2);
        new NewThread ("CSE",
        2000, 1);
    }
}

```

Output :-

BMS College of engineering
 CSE
 CSE
 CSE
 CSE
 CSE
 BMS College of engineering
 CSE
 CSE
 CSE
 CSE
 CSE

LABORATORY PROGRAM - 9

Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.

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

public class DivisionMain1 extends Frame implements ActionListener
{
    TextField num1,num2;
    Button dResult;
    Label outResult;
    String out="";
    double resultNum;
    int flag=0;

    public DivisionMain1()
    {
        setLayout(new FlowLayout());

        dResult = new Button("RESULT");
        Label number1 = new Label("Number 1:",Label.RIGHT);
        Label number2 = new Label("Number 2:",Label.RIGHT);
        num1=new TextField(5);
        num2=new TextField(5);
        outResult = new Label("Result:",Label.RIGHT);

        add(number1);
        add(num1);
        add(number2);
        add(num2);
        add(dResult);
        add(outResult);

        num1.addActionListener(this);
        num2.addActionListener(this);
        dResult.addActionListener(this);
        addWindowListener(new WindowAdapter()
        {
            public void windowClosing(WindowEvent we)
            {
                System.exit(0);
            }
        });
    }
}
```

```

        }
    });

}

public void actionPerformed(ActionEvent ae)
{
    int n1,n2;
    try
    {
        if (ae.getSource() == dResult)
        {
            n1=Integer.parseInt(num1.getText());
            n2=Integer.parseInt(num2.getText());

            /*if(n2==0)
                throw new ArithmeticException();
            out=n1+" "+n2+" ";
            resultNum=n1/n2;
            out+=String.valueOf(resultNum);
            repaint();*/

        }
    }
    catch(NumberFormatException e1)
    {
        flag=1;
        out="Number Format Exception! "+e1;
        repaint();
    }
    catch(ArithmeticException e2)
    {
        flag=1;
        out="Divide by 0 Exception! "+e2;
        repaint();
    }
}

public void paint(Graphics g)
{
    if(flag==0)
        g.drawString(out,outResult.getX()+outResult.getWidth(),outResult.getY()+outResult.getHeight()-8);
    else
        g.drawString(out,100,200);
    flag=0;
}

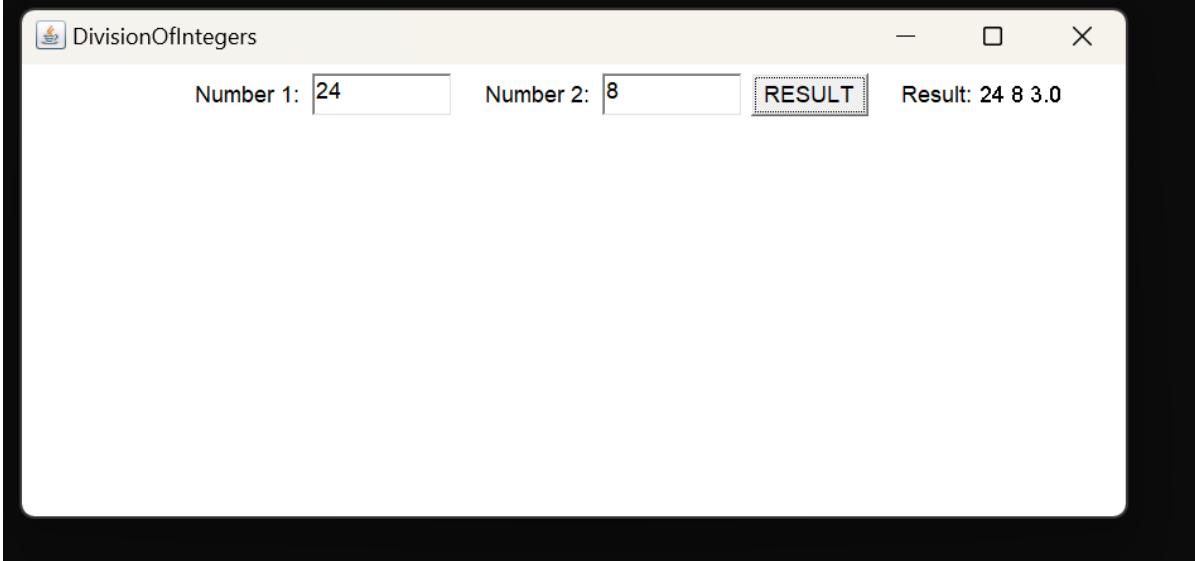
```

```
public static void main(String[] args)
{
    DivisionMain1 dm=new DivisionMain1();
    dm.setSize(new Dimension(800,400));
    dm.setTitle("DivisionOfIntegers");
    dm.setVisible(true);
}
}
```

OUTPUT

```
D:\NotePad++\Java>javac DivisionMain1.java
```

```
D:\NotePad++\Java>java DivisionMain1
```



Lab - program 9 :-

```

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

class SwingDemo{
    swing Demo(){
        JFrame frm = new JFrame("Divide App");
        frm.setSize(275, 150);
        frm.setLayout(new FlowLayout());
        frm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JLabel jlab = new JLabel("Enter the divisor and dividend:");
        JTextField aJtf = new JTextField(8);
    }
}

```

JTextField bJtf = new JTextField(8);
 JButton button = new JButton("Calculate");
 JLabel err = new JLabel();
 JLabel alab = new JLabel();
 JLabel blab = new JLabel();
 JLabel anslab = new JLabel();
 frm.add(err);
 frm.add(jlab);
 frm.add(g aJtf);
 frm.add(bJtf);
 frm.add(button);
 frm.add(alab);
 frm.add(blab);
 frm.add(anslab);

```

ActionListener l = new ActionListener(){
    public void actionPerformed(
        ActionEvent evt) {
        System.out.println("Action event from a text field");
    }
};

aJtf.addActionListener(new ActionListener() {
    public void actionPerformed(
        ActionEvent evt) {
        try {
            int a = Integer.parseInt(
                aJtf.getText());
            int b = Integer.parseInt(bJtf.getText());
            int ans = a/b;
            alab.setText("A = " + a);
        }
    }
});

```

```

        blab.setText("B = " + b);
        anslab.setText("Ans = " + ans);
    }

    catch (NumberFormatException e) {
        alab.setText("");
        blab.setText("");
        anslab.setText("");
        err.setText("Enter only integers!");
    }

    catch (ArithmaticException e) {
        alab.setText("");
        blab.setText("");
        anslab.setText("");
        err.setText("B should be non zero.");
    }
}

```

```
jfm. setVisible(true);  
}  
public static void main  
(String args[]) {  
    SwingUtilities.invokeLater(new  
    Runnable() {  
        public void run() {  
            new SwingDemo();  
        }  
    });  
}  
}
```

Output :-

```
Number1: 24 Number2: 8  
Result: 24 8 3.0
```

