VISVESVARAYA TECHNOLOGICAL UNIVERSITY

”JnanaSangama” Belgaum -590014, Karnataka.

LAB REPORT
on
Object Oriented Java Programming
(23CS3PCOOJ)

Submitted by:

CHANDRAKANT (1BM25CS454-T)

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 **CHANDRAKANT (1BM25CS454-T),** who is bonafide student of **B.M.S. College of Engineering.** It is in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements in respect of an Object-Oriented Java Programming (23CS3PCOOJ) work prescribed for the said degree.

Dr. Seema Patil Associate Professor Department of CSE, BMSCE	Dr. Kavitha Sooda Professor & HOD Department of CSE, BMSCE

Index

GitHub Link:

https://github.com/chandrakant-cs/Chandrakant_JAVA-LAB-Programs.git

Program 1

Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2 -4ac is negative, display a message stating that there are no real solutions.

Code:

```
import java.util.Scanner;

class quadratic
{
   public static void main (String args[])
   {
      int a,b,c,d;
      Scanner in =new Scanner(System.in);
      System.out.print("enter co-efficient of of A:");
      a=in.nextInt();

      System.out.print("enter co-efficient of of B:");
      b=in.nextInt();

      System.out.print("enter co-efficient of of C:");
      c=in.nextInt();

      if( a==0){
         System.out.println("Not a quadratic equation");
      }

      else{
         d=b*b-4*a*c;
         if (d==0){
            int r1;
```

```
            r1=(-b)/(2*a);
            System.out.println("Roots are real and equal");
            System.out.println("Root1 and Root2: "+r1);
        }
        else if (d>0) {
        double r1,r2;
            r1 = ((-b) + (Math.sqrt(d)))/(double)(2*a);
            r2 = ((-b) - (Math.sqrt(d)))/(double)(2*a);
            System.out.println("Root1: "+r1);
            System.out.println("Root2: "+r2);

        }

        else if(d<0){
            System.out.println("Roots are imaginary");
            double r1,r2;
            r1 = (-b)/(2*a);
            r2 = Math.sqrt(-d)/(2*a);
            System.out.println("Root1: "+r1);
            System.out.println("Root2:  "+r2);

        }

        }
    }

}
```

Output:

```
PS C:\Users\Ganga\Downloads> java quadratic
enter co-efficient of of A:1
enter co-efficient of of B:-2
enter co-efficient of of C:1
Roots are real and equal
Root1 and Root2: 1
PS C:\Users\Ganga\Downloads> java quadratic
enter co-efficient of of A:3
enter co-efficient of of B:9
enter co-efficient of of C:1
Root1: -0.11556268951365418
Root2: -2.8844373104863457
PS C:\Users\Ganga\Downloads> java quadratic
enter co-efficient of of A:0
enter co-efficient of of B:2
enter co-efficient of of C:4
Not a quadratic equation
PS C:\Users\Ganga\Downloads> java quadratic
enter co-efficient of of A:12
enter co-efficient of of B:9
enter co-efficient of of C:3
Roots are imaginary
Root1: 0.0
Root2: 0.33071891388307384
```

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.

Code:

```
import java.util.Scanner;

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

class Student {
    String name;
    String usn;
    double SGPA;
    Subject subject[];
    Scanner s = new Scanner(System.in);

    Student() {
        subject = new Subject[8];
        for (int i = 0; i < 8; i++) {
            subject[i] = new Subject();
        }
    }

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

    void getMarks() {
        for (int i = 0; i < 8; i++) {
            System.out.println("\nEnter marks of subject" + (i + 1)+"::");
```

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

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

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

    void computeSGPA() {
        int Score = 0, Credits = 0;
        for (int i = 0; i < 8; i++) {
            Score += (subject[i].grade * subject[i].credits);
            Credits += subject[i].credits;
        }
        SGPA = (double) Score / Credits;
    }

    void display() {
        System.out.println("Name: " + name);
        System.out.println("USN: " + usn);
        System.out.printf("SGPA: %.2f\n", SGPA);
    }
}

public class studentSGPA {
    public static void main(String[] args) {
        int n;
        System.out.println("\nenter number of students:: ");
        Scanner s=new Scanner(System.in);
        n=s.nextInt();
        while (n>0){
        Student st = new Student();
        st.getStudentDetails();
        st.getMarks();
        st.computeSGPA();
```

```
st.display();
n-=1; }}}
```

Output:

```
PS C:\Users\Ganga\Downloads> java studentSGPA

enter number of students:: 2
Enter Student Name: Chandrakant
Enter Student USN: 1BM25CS454

Enter marks of subject1::78
Enter Credits: 3

Enter marks of subject2::98
Enter Credits: 4

Enter marks of subject3::90
Enter Credits: 2

Enter marks of subject4::94
Enter Credits: 2

Enter marks of subject5::98
Enter Credits: 3

Enter marks of subject6::80
Enter Credits: 5

Enter marks of subject7::95
Enter Credits: 2

Enter marks of subject8::99
Enter Credits: 2
Name: Chandrakant
USN: 1BM25CS454
SGPA: 9.52
```

```
Enter Student Name: Deepak
Enter Student USN: 1BM25CS244

Enter marks of subject1::97
Enter Credits: 3

Enter marks of subject2::94
Enter Credits: 3

Enter marks of subject3::79
Enter Credits: 4

Enter marks of subject4::86
Enter Credits: 4

Enter marks of subject5::86
Enter Credits: 3

Enter marks of subject6::90
Enter Credits: 4

Enter marks of subject7::94
Enter Credits: 3

Enter marks of subject8::87
Enter Credits: 4
Name: Deepak
USN: 1BM25CS244
SGPA: 9.32
```

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

Code:

```
import java.util.Scanner;


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

    Books(String name, String author, int price, int numPages){
        this.name=name;
        this.author=author;
        this.price=price;
        this.numPages=numPages;
    }

    public String getName(){
        return this.name;
    }

    public String getAuthor(){
        return this.author;
    }
    public int getNumPages(){
        return this.numPages;
    }

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

```
    }

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

    public void setPrice(int price){
        this.price=price;
    }

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

    @Override
     public String toString() {
        String details = "Book name: " + this.name + "\n" +
                         "Author name: " + this.author + "\n" +
                         "Price: " + this.price + "\n" +
                         "Number of pages: " + this.numPages + "\n";
        return details;
    }
}
public class library {
    public static void main(String[] args) {
        Scanner s=new Scanner(System.in);
        System.out.print("Enter the number of books: ");
        int n=s.nextInt();
        s.nextLine();

        Books[] b=new Books[n];

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

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

            System.out.print("Enter author: ");
            String author = s.nextLine();
```

```
            System.out.print("Enter price: ");
            int price=s.nextInt();

            System.out.print("Enter number of pages: ");
            int numPages=s.nextInt();
            s.nextLine();

            b[i] = new Books(name,author,price,numPages);
        }
        System.out.println("Book Details:");

        for(int i=0;i<n;i++){
            System.out.println("Book "+(i+1));
            System.out.println(b[i]);
        }
    }
}
```

Output:

```
Enter the number of books: 2

Enter details for Book 1:
Enter name: java-programming
Enter author: james gosling
Enter price: 200
Enter number of pages: 100

Enter details for Book 2:
Enter name: python-programming
Enter author: denis ritchie
Enter price: 300
Enter number of pages: 120
```

```
Book Details:
Book 1
Book name: java-programming
Author name: james gosling
Price: 200
Number of pages: 100

Book 2
Book name: python-programming
Author name: denis ritchie
Price: 300
Number of pages: 120
```

Program 4

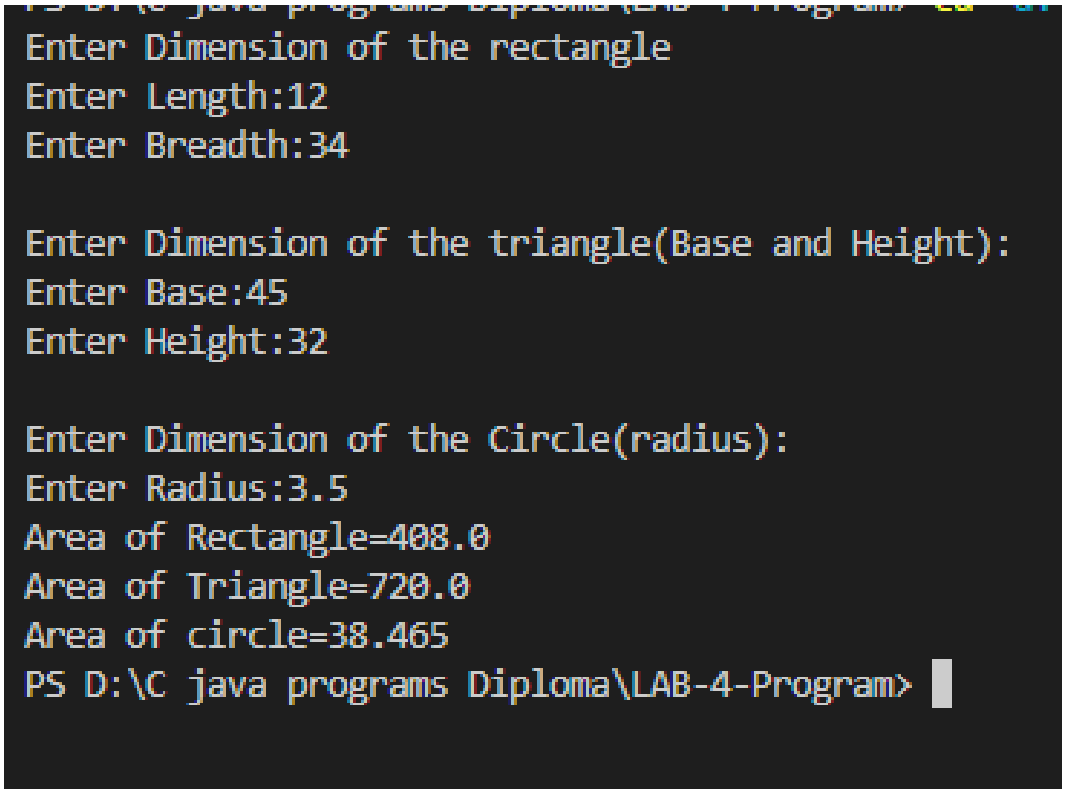
Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

Code:

```
import java.util.Scanner;


abstract class Shape {
   double dim1;
   double dim2;

   public Shape(double dim1,double dim2){
      this.dim1=dim1;
      this.dim2=dim2;
   }
   public abstract void printArea();
}

class Rectangle extends Shape{
   public Rectangle(double length,double width){
      super(length,width);
   }
   public void printArea(){
      double area=dim1*dim2;
      System.out.println("Area of Rectangle="+area);
   }
}
class Triangle extends Shape{
   public Triangle(double base,double height){
      super(base,height);
   }
   public void printArea(){
      double area= 0.5* dim1* dim2;
```

```
        System.out.println("Area of Triangle="+area);
    }
}

class Circle extends Shape{
    public Circle(double radius){
        super(radius,0);
    }
    public void printArea(){
        double area = 3.14*dim1*dim1;
        System.out.println("Area of circle="+area);
    }
}

public class ShapeDemo{
    public static void main(String[] args) {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter Dimension of the rectangle");
        System.out.print("Enter Length:");
        double recLen=s.nextDouble();
        System.out.print("Enter Breadth:");
        double recWidt=s.nextDouble();
        Shape rect=new Rectangle(recLen,recWidt);


        System.out.println("\nEnter Dimension of the triangle(Base and Height):");
        System.out.print("Enter Base:");
        double triBase=s.nextDouble();
        System.out.print("Enter Height:");
        double triHeight=s.nextDouble();
        Shape tri=new Triangle(triBase,triHeight);

        System.out.println("\nEnter Dimension of the Circle(radius):");
        System.out.print("Enter Radius:");
        double r=s.nextDouble();
        Shape circle=new Circle(r);

        rect.printArea();
        tri.printArea();
        circle.printArea(); }}
```

Output:


Enter Dimension of the rectangle
Enter Length:12
Enter Breadth:34
Enter Dimension of the triangle(Base and Height):
Enter Base:45
Enter Height:32
Enter Dimension of the Circle(radius):
Enter Radius:3.5
Area of Rectangle=408.0
Area of Triangle=720.0
Area of circle=38.465
PS D:\C java programs Diploma\LAB-4-Program>

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.

Code:

```
import java.util.Scanner;

class Account {
    String name;
    int accno;
    String type;
    double balance;

    Account(String n, int a, String t, double b) {
        name = n;
        accno = a;
        type = t;
        balance = b;
    }

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

```
    }

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


class SavAccount extends Account {
    SavAccount(String n, int a, double b) {
        super(n, a, "saving", b);
    }

    void interest() {
        double rate = 0.04; // 4% interest
        double intr = balance * rate;
        balance += intr;
        System.out.println("Interest added: " + intr + ". Updated balance: " + balance);
    }

    void withdraw(double amt) {
        if (amt <= balance) {
            balance -= amt;
            System.out.println("Withdrawn: " + amt + ". Updated balance: " + balance);
        } else {
            System.out.println("Insufficient balance.");
        }
    }
}


class CurAccount extends Account {
    CurAccount(String n, int a, double b) {
        super(n, a, "current", b);
    }

    void withdraw(double amt) {
```

```
        if (amt <= balance) {
            balance -= amt;
            System.out.println("Withdrawn: " + amt + ". Updated balance: " + balance);
            checkMinBalance();
        } else {
            System.out.println("Insufficient balance.");
        }
    }

    void checkMinBalance() {
        double min = 2000;
        if (balance < min) {
            balance -= 100;
            System.out.println("Service charge imposed! Updated balance: " + balance);
        }
    }
}


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

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

        System.out.print("Enter account number: ");
        int accno = s.nextInt();

        System.out.print("Enter account type (saving/current): ");
        String type = s.next();

        System.out.print("Enter initial balance: ");
        double bal = s.nextDouble();

        if (type.equalsIgnoreCase("saving")) {
            SavAccount sa = new SavAccount(name, accno, bal);
            menu(sa, s);
        } else {
            CurAccount ca = new CurAccount(name, accno, bal);
```

```
        menu(ca, s);
    }
}

static void menu(Account acc, Scanner s) {
    int choice;
    do {
        System.out.println("\n------MENU------ ");
        System.out.println("1. Deposit");
        System.out.println("2. Withdraw");
        System.out.println("3. Compute Interest for Savings Account");
        System.out.println("4. Display Account Details");
        System.out.println("5. Exit");
        System.out.print("Enter your choice: ");
        choice = s.nextInt();

        switch (choice) {
            case 1:
                System.out.print("Enter the deposit amount: ");
                double d = s.nextDouble();
                acc.deposit(d);
                break;
            case 2:
                System.out.print("Enter the withdrawal amount: ");
                double w = s.nextDouble();
                if (acc instanceof SavAccount)
                    ((SavAccount) acc).withdraw(w);
                else if (acc instanceof CurAccount)
                    ((CurAccount) acc).withdraw(w);
                break;
            case 3:
                if (acc instanceof SavAccount)
                    ((SavAccount) acc).interest();
                else
                    System.out.println("Interest not available for current account.");
                break;
            case 4:
                acc.display();
                break;
            case 5:
```

```
                System.out.println("Exiting...");
                break;
            default:
                System.out.println("Invalid choice.");
        }
    } while (choice != 5);
  }
}
```

Output:

```
Enter customer name: chandu
Enter account number: 12345
Enter account type (saving/current): saving
Enter initial balance: 4500

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 1
Enter the deposit amount: 500
Deposited: 500.0. Updated balance: 5000.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 2
Enter the withdrawal amount: 2000
Withdrawn: 2000.0. Updated balance: 3000.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 3
Interest added: 120.0. Updated balance: 3120.0
```

```
------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 4
Customer name: chandu
Account number: 12345
Type of Account: saving
Account Balance: 3120.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 5
Exiting...
PS D:\C java programs Diploma\LAB-5-Program>
```

```
Enter customer name: chandu
Enter account number: 3456
Enter account type (saving/current): current
Enter initial balance: 4029

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 1
Enter the deposit amount: 30004
Deposited: 30004.0. Updated balance: 34033.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 1
Enter the deposit amount: 30004
Deposited: 30004.0. Updated balance: 34033.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
5. Exit
Enter your choice:
```

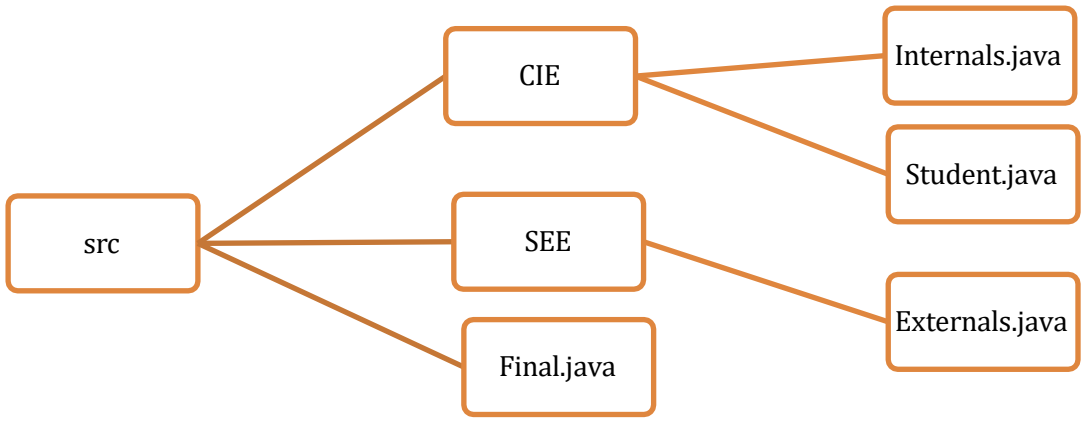
```
Enter your choice: 2
Enter the withdrawal amount: 23456
Withdrawn: 23456.0. Updated balance: 10577.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 3
Interest not available for current account.

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 4
Customer name: chandu
Account number: 3456
Type of Account: current
Account Balance: 10577.0

------MENU------
1. Deposit
2. Withdraw
3. Compute Interest for Savings Account
4. Display Account Details
5. Exit
Enter your choice: 5
Exiting...
```

Program 6

Create a package CIE which has two classes - Personal 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 Personal. 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.

Code:

File Structure



Internals.java

```
package CIE;

import java.util.Scanner;

public class Internals extends Student {
    protected int[] internalMarks = new int[5];

    public void inputInternalMarks() {
        Scanner s = new Scanner(System.in);
```

```
        System.out.println("Enter Internal Marks (out of 50) for 5 subjects:");
        for (int i = 0; i < 5; i++) {
            System.out.print("Subject " + (i + 1) + ": ");
            internalMarks[i] = s.nextInt();
        }
    }

    public void displayInternalMarks() {
        System.out.println("Internal Marks:");
        for (int i = 0; i < 5; i++) {
            System.out.println("Subject " + (i + 1) + ": " + internalMarks[i]);
        }
    }
}
```

Student.java

```
package CIE;

import java.util.Scanner;

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

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

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

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

Externals.java

```
package SEE;

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

public class Externals extends Internals {
    protected int[] seeMarks = new int[5];
    protected double[] finalMarks = new double[5];

    public Externals() {
        for (int i = 0; i < 5; i++) {
            seeMarks[i] = 0;
            finalMarks[i] = 0.0;
        }
    }

    public void inputSEEmarks() {
        Scanner s = new Scanner(System.in);
```

```
        System.out.println("Enter SEE Marks (out of 100) for 5 subjects:");
        for (int i = 0; i < 5; i++) {
            System.out.print("Subject " + (i + 1) + ": ");
            seeMarks[i] = s.nextInt();
        }
    }

    public void calculateFinalMarks() {
        for (int i = 0; i < 5; i++) {
            finalMarks[i] = internalMarks[i] + (seeMarks[i] / 2.0);
        }
    }

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

Final.java

```
import SEE.*;
import java.util.*;

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

        System.out.print("Enter number of students: ");
        int n = sc.nextInt();
```

```
        Externals[] students = new Externals[n];

        for (int i = 0; i < n; i++) {
            students[i] = new Externals();
            System.out.println("\nEnter Details for Student " + (i + 1) + "");
            students[i].inputStudentDetails();
            students[i].inputInternalMarks();
            students[i].inputSEEmarks();
            students[i].calculateFinalMarks();
        }

        System.out.println("\n========== FINAL RESULTS ==========");
        for (int i = 0; i < n; i++) {
            students[i].displayFinalMarks();
        }
    }
}
```

Output :

```
Enter number of students: 1

Enter Details for Student 1
Enter USN: 1BM25CS454
Enter Name:  Chandrakant
Enter Semester: 3
Enter Internal Marks (out of 50) for 5 subjects:
Subject 1: 45
Subject 2: 50
Subject 3: 46
Subject 4: 40
Subject 5: 41
Enter SEE Marks (out of 100) for 5 subjects:
Subject 1: 89
Subject 2: 90
Subject 3: 95
Subject 4: 97
Subject 5: 91

========== FINAL RESULTS ==========
USN: 1BM25CS454
Name:  Chandrakant
Semester: 3
Final Marks in 5 Subjects:
Subject 1: 89.5
Subject 2: 95.0
Subject 3: 93.5
Subject 4: 88.5
Subject 5: 86.5
PS D:\C java programs Diploma\LAB-6-Program>
```

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=father’s age

Code:

Father.java

```
import java.util.Scanner;

class WrongAge extends Exception{
   public WrongAge(String msg){
      super(msg);
   }
}

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

class Father extends InputScanner{
   int FatherAge;
   Father() throws WrongAge{
      System.out.print("Enter Father's Age:");
      FatherAge= s.nextInt();

      if (FatherAge<=0){
      throw new WrongAge("Age cannot Be Zero or Negative");
      }
   }
```

```
    void displayFather(){
        System.out.print("Fathers Age is "+FatherAge);
    }}
```

Son.java

```
class Son extends Father{
    int SonAge;

    Son() throws WrongAge{
        System.out.print("Enter Son's Age:");
        SonAge=s.nextInt();

        if(SonAge<=0){
            throw new WrongAge("Son's Age can not be Negative or Zero");
        }
        else if(SonAge>=FatherAge){
            throw new WrongAge("Son's Age cannot be greater than Fathers Age");
        }
    }

    void display(){
        super.displayFather();
        System.out.println("\nSon's Age is:"+SonAge);
    }
}
```

Main.java

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

            Son obj= new Son();
```

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

                    }
          }
}
```

Output

```
PS D:\C java programs Diploma\lab 7> cd "d:\C java programs Diploma\lab 7\"
Enter Father's Age:0
Exception: Age cannot Be Zero or Negative
PS D:\C java programs Diploma\lab 7> cd "d:\C java programs Diploma\lab 7\"
Enter Father's Age:12
Enter Son's Age:0
Exception: Son's Age can not be Negative or Zero
PS D:\C java programs Diploma\lab 7> cd "d:\C java programs Diploma\lab 7\"
Enter Father's Age:12
Enter Son's Age:8
Fathers Age is 12
Son's Age is:8
PS D:\C java programs Diploma\lab 7> cd "d:\C java programs Diploma\lab 7\"
Enter Father's Age:34
Enter Son's Age:78
Exception: Son's Age cannot be greater than Fathers Age
PS D:\C java programs Diploma\lab 7>
```

Program 8

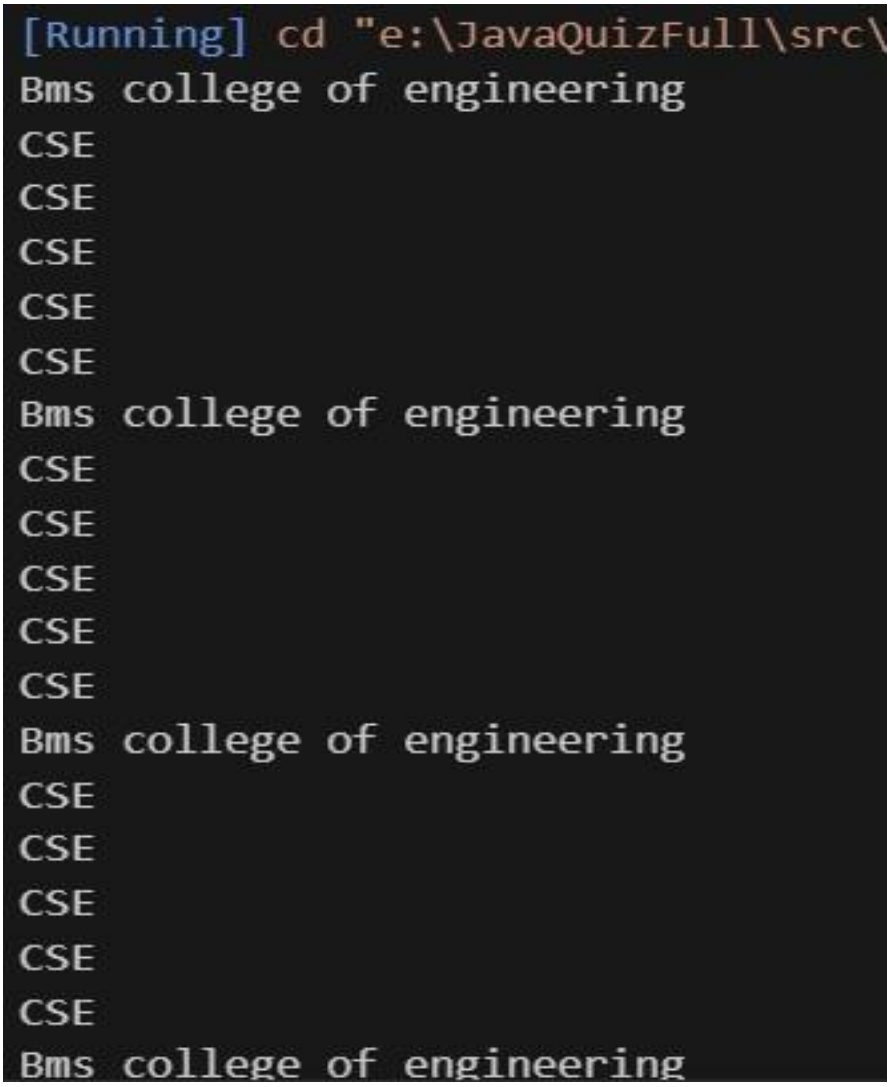
Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

Code:

```
class Bms extends Thread
{
   public void run()
   {
      try
      {
         while (true)
         {
            System.out.println("Bms college of engineering");
            Thread.sleep(10000);
         }
      }
         catch(InterruptedException e)
         {
            System.out.println("thread  inturrupted");
         }

   }
}

class cseprinter extends Thread
{
   public void run()
   {
      try{
         while (true)
         {
```

```
            System.out.println("CSE");
            Thread.sleep(2000);
        }
      }
      catch(InterruptedException e)
      {
        System.out.println("thread interrupt");
      }
   }
}

public class Bmsprinter
{
   public static void main(String[]args)
   {
      Bms t1=new Bms();
      cseprinter t2=new cseprinter();
      t1.start();
      t2.start();
   }
}
```

Output:


[Running] cd "e:\JavaQuizFull\src\
Bms college of engineering
CSE
CSE
CSE
CSE
CSE
Bms college of engineering
CSE
CSE
CSE
CSE
CSE
Bms college of engineering
CSE
CSE
CSE
CSE
CSE
Bms college of engineering