## VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.



LAB REPORT on

# Object Oriented Java Programming (23CS3PCOOJ)

Submitted by

Chadive Muralidhar Reddy (1BM23CS072)

in partial fulfillment 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

#### Sep-2024 to Jan-2025

#### 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 Chadive Muralidhar Reddy (1BM23CS072), who is bonafide student of B.M.S. College of Engineering. It is in partial fulfillment 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.

Swathi Sridharan Assistant Professor Department of CSE, BMSCE Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE

## Index

Sl. No.	Date	Experiment Title	Page No.
1	1/10/2024	Roots of Quadratic Equations	4-6
2	8/10/2024	SGPA Calculation	6-9
3	15/10/2024	Method Overriding	9-13
4	22/10/2024	Abstract Class	14-16
5	29/10/2024	Bank Account	16-22
6	19/11/2024	Packages	23-27
7	26/11/2024	Interfaces	27-30
8	26/12/2024	Exception Handling	31-33
9	3/12/2024	Threads	33-34
10	3/12/2024	Calculator	35-36

Github Link:
<a href="mailto:murali1231944/JAVA\_LAB">murali1231944/JAVA\_LAB</a>

Program 1 Implement Quadratic Equation

Algorithm:

```
Quadractic Expression:
import java-util - Scanner;
public class Main {
     public static void main (string[/] args(1) {
      Scanner input = new Scanner (System-in);
      System out println ("Enter number 1 and number 2");
  and no = input - next Int();
     int numb = input next Int();
   double Pleat de
              c = input · next Int();
 show lint
       double d = 6 +6 - 4 + a + C = 11
       if (d==0) { 100 = 100 = 3/41
             int 1001 = - b/(2+a);
              System. out-print In ("Roots are equal and the
                value of root is " + root 1);
    else if (dro) {
     int root 1 = (+6 + Ja) (2xa);
                int 100t2 = (-b - \( \tau d \)
                System-out println (" Roots are real and
                 distinct and the roots are "+ 10011
                                            + 100t2);
         enter a, b, 1
Output: 123,41,5
          roots are imaginary
```

package LAB1; import java.util.Scanner; public class Quad { public static void main(String [] args) { Scanner input = new Scanner(System.in); System.out.println("enter a b c constants in the quad equaion"); int a = input.nextInt();

```
C:\Users\mural\OneDrive\Desktop\java_lab\LAB1>javac Quad.java
C:\Users\mural\OneDrive\Desktop\java_lab\LAB1>java Quad
Error: Could not find or load main class Quad
Caused by: java.lang.NoClassDefFoundError: LAB1/Quad (wrong name: Quad)
C:\Users\mural\OneDrive\Desktop\java_lab\LAB1>java Quad.java
enter a b c constants in the quad equaion
1
2
3
roots are imaginary
C:\Users\mural\OneDrive\Desktop\java_lab\LAB1>|
```

#### 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 1 of
           String name:
           Storing USN;
            int subjects :
            int totalcredits = 0.
             Student (name, USN) {
                -this-rame = rame;
                 this - USN = USN ;
              void set Manks (int Subjects) {
                  Scanner input = new Scanner (System-in);
                   for (int it o: ix Bubjects; it ) {
                      System out print In ("Enter sub" + "" +
                                          monts and credits);
                        total credits
                              manks (i) = input next Int();
                               eredits [i] = input next Int().
                               total credits += [manks[i])+1
                                                  10
                                                * credits(i);
                System-out. print In (total credits/20);
                System out print (n (this name);
                 System out println (this usn) -
public class
              Student &
       students si= new Students ("Kumas", "IRM"
             181 - set Manks (6);
```

```
package LAB2;
import java.util.Scanner;
public class Main {
  public static void main(String[] args) {
     Scanner sc = new Scanner(System.in);
     System.out.println("Enter the number of students:");
     int n = sc.nextInt();
     sc.nextLine(); // Consume leftover newline
     Student[] students = new Student[n];
     for (int i = 0; i < n; i++) {
       System.out.println("Enter the name and the USN:");
       String name = sc.nextLine();
       String USN = sc.nextLine();
       System.out.println("Enter the number of subjects:");
       int n1 = sc.nextInt();
       int[] marks = new int[n1];
```

```
int[] credits = new int[n1];
       for (int j = 0; j < n1; j++) {
          System.out.println("Enter the marks for subject " + (j + 1) + ":");
          marks[i] = sc.nextInt();
          System.out.println("Enter the credits for subject " + (j + 1) + ":");
          credits[i] = sc.nextInt();
        sc.nextLine(); // Consume leftover newline
       students[i] = new Student(name, USN, credits, marks);
     }
     // Display SGPA for each student
     for (Student student : students) {
        student.s_gpa();
     sc.close();
}
class Student {
  String name;
  String USN;
  int[] credits;
  int[] marks;
  Student(String name, String USN, int[] credits, int[] marks) {
     this.name = name;
     this.USN = USN;
     this.credits = credits;
     this.marks = marks;
  }
  public void s_gpa() {
     int totalCredits = 0;
     int totalWeightedMarks = 0;
     for (int i = 0; i < credits.length; <math>i++) {
        totalCredits += credits[i]:
        totalWeightedMarks += marks[i] * credits[i];
     }
     double sgpa = (double) totalWeightedMarks / totalCredits;
     System.out.println("NAME = " + name +
          "\nUSN = " + ÙSN +
          \nSGPA = " + sgpa);
```

```
C:\Users\mural\OneDrive\Desktop\java_lab\LAB2>javac Main.java
C:\Users\mural\OneDrive\Desktop\java_lab\LAB2>java Main.java
Enter the number of students:
1
Enter the name and the USN:
m
23
Enter the number of subjects:
3
Enter the marks for subject 1:
23
Enter the credits for subject 1:
2
Enter the marks for subject 2:
23
Enter the credits for subject 2:
24
Enter the credits for subject 3:
25
Enter the credits for subject 3:
27
NAME = m
USN = 23
SGPA = 23.0
```

#### 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.

Algorithmn:

```
import Java-util- Examer;
elass Book f
   private Storing name :
    prevate string purther,
     private int price;
private int num pages;
     public Book (Stiring name, eting author, int price, int
              -this-author = author;
                -this price = price ;
                this - num-pages = num-pages ;
      public void setName (String name) {
this name = name;
        public string getName() {
return this name;
        public void set Author (String author) {
               this author = author;
         public string getAuthor() of
         public void setPrice(int price) {
                this price = price ;
     public int getPrice () {
        return this price ?
      public indid setMuonPages (int nuon-pages) {
       public int get Mum Pages (In) }
              return this num pages
      @ Override
      public strong tostronger f
return "Nome = " + this - name + ee
                       'Author = " + this - outhor +"
                         Price = " + this-price + "
                        MumPages = " + this - num pages
```

```
public class BeakInform {

public static void main (String() args) {

B Scanner sc = new Scanner(Eystern-in);

int() book = new int();

sc nextline;

int() book = new int(n);

for (int i=0; i < n; i++) {

System out println("Enter book name");

System out println("Enter book name");

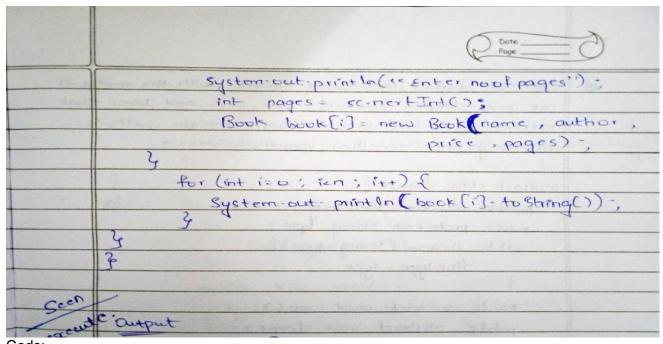
System out println("Enter author");

System out println("Enter Enter Price");

System out println("Enter Price");

System out println("Enter Price");

System out println("Enter Price");
```



Code: package LAB3;

```
import java.util.Scanner;
class Book{
  private String name;
  private String author;
  private int num pages;
  private int price;
  public Book(String name, String author, int 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 void setAuthor(String author){
     this.author =author;
  public void setPrice(int price){
     this.price =price;
  public void setName(int num_pages){
     this.num pages = num pages;
  public String getName(){
     return this.name;
  public String getAuthor(){
     return this.author;
  public int getPrice(){
     return this.price;
  public int getNumPages(){
     return this.num_pages;
  public String toString(){
     return "Name= "+this.name+" Author= "+this.author+" Price= "+this.price+" Numpages=
"+this.num_pages;
public class BookInform{
  public static void main(String[] args){
     Scanner sc = new Scanner(System.in);
     System.out.println("Enter no of books");
     int n= sc.nextInt();
     sc.nextLine();
     Book[] book=new Book[n];
     for (int i=0;i< n;i++){
       System.out.println("Enter book name");
       String name=sc.nextLine();
       System.out.println("Enter author name");
       String author =sc.nextLine();
```

```
System.out.println("Enter Price");
int price = sc.nextInt();
System.out.println("Enter no of book pages");
int pages =sc.nextInt();
book[i]=new Book(name,author,price,pages);
sc.nextLine();

}
for (int i=0;i<n;i++){
    System.out.println(book[i].toString());
}
}</pre>
```

```
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB3/BookInform.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB3.BookInform
Enter no of books
Enter book name
harrypotter
Enter author name
jk rowling
Enter Price
120
Enter no of book pages
679
Enter book name
lord of the rings
Enter author name
stephenwolf
Enter Price
456
Enter no of book pages
980
Name= harrypotter Author= jk rowling Price= 120 Numpages= 679
Name= lord of the rings Author= stephenwolf Price= 456 Numpages= 980
```

#### 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 contains only the method **printArea()** that prints the area of the given shape

Algorithm:

abolistract class Shape 80 {
private
int a ;
ind b;
public Shape (int a , int b) {
this 10=9
+this b = b
7
public abstract Word printArca()
1 5
class Rectangle extends Shape f
public Rectongle (Int length, But breadth) of
super (length, breadth);
4
public void print Area () {
System out. printin("Area of Rectangle: "+ this a  this b);
+ this by
Character Thanks of the State o
class Triongle extends Bapapers
public Triangle ( out booke, int height) = euper (boxe, height);
}
public void printArm() {
Egetern out printle ("Avea Triangle: "+
05+this a x this b)
\$ 05 ETAGE & X 10003 37
le.
I class Circle extends Shape &
public Circle ( but radius) &
super (radius, radius);
}
public void print Area () {
System out print lin ( · Area Circle = " +
3 14 * this. axithis.a)

```
public class Bircon &

public static void male (String... args) &

Rectange & - view Rectangle (2,3)

Riongle + - now Triangle (2,15);

+-print Area();

Citcle e = new Circle (10);

Capaint Area();
```

```
package LAB4;
abstract class Shape{
int a:
 int b;
 public Shape(int a,int b){
   this.a =a:
   this.b =b;
 public abstract void printArea();
class Triangle extends Shape{
 public Triangle(int base,int height){
   super(base,height);
 public void printArea(){
   System.out.println("Area of the Triangle "+this.a*this.b*0.5);
class Rectangle extends Shape{
 public Rectangle(int breadth,int lenght){
   super(breadth,lenght);
 public void printArea(){
   System.out.println("Area of the Rectangle "+this.a*this.b);
class Circle extends Shape{
 public Circle(int radius){
```

```
super(radius,0);
}

public void printArea(){
    System.out.println("Area of the Circle "+this.a*this.a*3.14);
}

public class Area{
    public static void main(String... args){
        Rectangle r = new Rectangle(2,4);
        r.printArea();
        Triangle t = new Triangle(2,5);
        t.printArea();
        Circle c= new Circle(5);
        c.printArea();
}
```

```
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB4/Area.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB4.Area
Area of the Rectangle 8
Area of the Triangle 5.0
Area of the Circle 78.5
```

#### **Program 5**

**Develop a Java program to create a class Bank** that maintains two kinds of accounts 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.

Algorithm:

8012	class Account
	private string Cust Name;
	marcha China Acct No:
	Drivate string Balance: etving
	public Account (String Cout Name, Acct No, double bolance)
	this. Cust Name = Cust Name;
	3
	outstic void cel (double balance) {
	this balance = balance;
	public double get Balance () {
	return this balance
	4
الماطوس	e vold deposit (double amt) for holding
of two	- (# (aut 20) { = () } 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	His dolonce 4 = aut;
-	private desiring Balance; etring  public parasest (String CoutName, AcctNo, double balance)  this. GutName = GutName;  this. AcctNo = AcctNo;  this. balance = balance;  public void cetB (double balance) f  this. balance = balance;  public alouble get Balance;  public double get Balance;  }  return ethic balance;
	sout (" Amount should not be negative ") -
person	there is completely and the se
	ź ,
	The state of the s

class Savinge Aco of 1984 19 19 19 30 198
private double Puterest Rate;
private Account a;
lange to a second to produce to the court of
public (String couthlance, String AceNo, double Balance,
double Interest Rate, Account a) {
gapet (contitaire; Acreto, Balance);
this Interest Rate = Interest Rate;
- Ais a : new Account (cust Nome AccNo, Bolow Ce)
-ce)
public void addInterest (200)
int interest = anget Balance ( ) + Interest Rate
100
a · deposit (interest);
Cons Colone State Side Sides
The service of the se
records .
class Current Ace {
private double min Balance;
orivate Account 6:
public ( String Cust Name , String Acc No, double belonce,
double Minbalance, Account b) {
super (Costolo min Balance = min Balance; this - is = new Account (CustName, AccNo, Balance);

```
Public void withdraw (double ant) &
          if (ant >0 & (Bogotlodance() - ant) > mings
                  b. getbalance () - = amt;
                  sout ( b. get balance).;
           else f
                 sout ( " withdraw is not possible ").
                  check Book (double ant) {
     public void
           if (amt >0 & (b) getbalance ()-amt) > mindlance
                  bogetbalance() - = amt;
                   sout (b. get bolance):
                   east ("The Amount is drawn through
                                           check ").
            else &
public
                                                  16000);
                        new Current Acc ( Murali
                  15000, 1000, new Account
           d - withdraw (12000) =
```

package LAB5;

import java.util.Scanner;

class Account {
 private String custName;
 private String accNo;
 private double balance;

public Account(String custName, String accNo, double balance) {

```
this.custName = custName:
     this.accNo = accNo:
     this.balance = balance;
  }
  public double getBalance() {
     return this.balance:
  public void deposit(double amount) {
     if (amount > 0)
       this.balance += amount;
       System.out.println("The current balance is " + this.balance);
     } else {
       System.out.println("Amount should not be negative");
  }
  public void withdraw(double amount) {
     if (amount > 0 && (balance - amount) >= 0) {
       this.balance -= amount;
       System.out.println("Withdraw successful. Current balance: " + this.balance);
     } else {
       System.out.println("Withdraw is not possible");
  }
class SavingsAccount {
  private double interestRate;
  private Account account;
  public SavingsAccount(String custName, String accNo, double balance, double interestRate) {
     this.interestRate = interestRate;
     this.account = new Account(custName, accNo, balance);
  }
  public void addInterest() {
     double interest = account.getBalance() * this.interestRate;
     account.deposit(interest);
  }
  public Account getAccount() {
     return account;
}
class CurrentAccount {
  private double minBalance;
  private Account account;
  public CurrentAccount(String custName, String accNo, double balance, double minBalance) {
     this.minBalance = minBalance;
```

```
this.account = new Account(custName, accNo, balance);
  }
  public void withdraw(double amt) {
     if (amt > 0 && (account.getBalance() - amt) >= minBalance) {
       account.withdraw(amt);
     } else {
       System.out.println("Withdraw is not possible");
  }
  public Account getAccount() {
     return account;
}
public class Bank {
  public static void main(String[] args) {
     Scanner sc = new Scanner(System.in);
     System.out.println("Enter the name:");
     String name = sc.nextLine();
     System.out.println("Enter the account number:");
     String accnt = sc.nextLine():
     while (true) {
       System.out.println("Enter your choice:");
       System.out.println("1. Savings Account");
       System.out.println("2. Current Account");
       System.out.println("3. Exit");
       int choice = sc.nextInt();
       switch (choice) {
          case 1:
            System.out.println("Enter initial balance:");
            double savingsBalance = sc.nextDouble();
            System.out.println("Enter the interest rate:");
            double interestRate = sc.nextDouble():
            SavingsAccount savingsAccount = new SavingsAccount(name, accnt, savingsBalance,
interestRate);
            savingsAccount.addInterest();
            break:
          case 2:
            System.out.println("Enter initial balance:");
            double currentBalance = sc.nextDouble();
            System.out.println("Enter minimum balance:");
            double minBalance = sc.nextDouble();
            CurrentAccount currentAccount = new CurrentAccount(name, accnt, currentBalance,
minBalance):
            System.out.println("entr the amount to be withdraw");
            double q = sc.nextInt();
            currentAccount.withdraw(q);
```

```
System.out.println("Account created. Current balance: " + currentAccount.getAccount().getBalance()); break;

case 3:
    System.out.println("Exiting..."); sc.close(); return;

default:
    System.out.println("Invalid choice. Please try again.");
}

}
}
```

```
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB5/Bank.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB5.Bank
Enter the name:
murali
Enter the account number:
123
Enter your choice:
1. Savings Account
2. Current Account
   Exit
Enter initial balance:
1000
Enter the interest rate:
The current balance is 3000.0
Enter your choice:
1. Savings Account
2. Current Account
3. Exit
Enter initial balance:
3000
Enter minimum balance:
1000
entr the amount to be withdraw
Withdraw successful. Current balance: 1000.0 Account created. Current balance: 1000.0 Enter your choice:

1. Savings Account
2. Current Account
    Fxit
Exiting...
```

#### 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 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.

Algorithm:

```
package CIE;
 public class Student f.
      statue vens
      String name:
      String sen;
      public Student (String um, String ware, String sen):
          this un = usn;
          this name = nome;
          this sem = sem;
    20
 4
                      a where deposit on the state of
package CIE;
 public class Internals extends Student of
       public int () masks = new int (s):
       Super (ush, name, sem);
        te's marks: marks;
 3
package SEC;
import are Student:
public class Externals extends Student (
      public fate() marks = new rut(s);
      public Externals (String un, String name String sem,
       super (usn, name, sem); int () magnes) of
       ethis marks: marks;
```

```
import CIE +x;
import SEC . x ;
import java util . Sconner:
public class Inform &
        public static void main ( string[ ? args) &
             Scanner sc = new Scanner (System in);
              Eystemout pointle (ernor of students");
               int n = sc vext(ut();
               Internals () firsternals = new Internals (m);
              Externals () Externals: new Externals (n);
              -for (int i=0: icn; i+){
                      Eystemout print (12USN");
                       String usn = ec next(me();
                      System out - printly (CENAME");
                       String name = sc. next Line();
                        Cyctemout - priorth (" SEM >>);
                          String sem = sc-nextline();
                          Systemout. print Ly (ceEnter CIE macks").
                          "ut () (marks = new int (5);
                          int ( ] Bmaiks = new int (5);
                            For (int j=0; j=5; j+1) {.

Charles(i) = sc next[let();
                           For (int j: 0: j=5: j++) {
Smarks (i) = scinextlutes;
                           internals(i) = new Internals (usn, name sen,
                           externals[i]: new loternals(usu, name,
                          +Bi(mi=0; j=5; j+1) {

int final = internals(i) - masks(j) +

externals(i) - masks(i)
                                          externals(i) - masts(i)
                             gout ( " Bub" + ( it) + " marks "4
```

```
Code:
package LAB6.CIE;

public class Student1 {
    String usn;
    String name;
    String sem;

public Student1(String usn, String name, String sem) {
    this.usn = usn;
    this.name = name;
```

```
this.sem = sem;
package LAB6.CIE;
public class Internals extends Student1 {
  public int[] marks = new int[5];
  public Internals(String usn, String name, String sem, int[] marks) {
     super(usn, name, sem);
     this.marks=marks;
     System.out.println("CIE MARKS -----" +
       "\nS1 : " + this.marks[0] +
       \normalfont{"}\normalfont{NS2}: "+ this.marks[1] +
       "\nS3 : " + this.marks[2] +
       "\nS4 : " + this.marks[3] +
       "\nS5:" + this.marks[4]);
}
package LAB6.SEE;
import LAB6.CIE.Student1;
public class Externals extends Student1{
 public int[] marks=new int[5];
  public Externals(String usn,String name,String sem,int[] marks){
     super(usn,name,sem);
     this.marks = marks;
     System.out.println("SEE MARKS -----"+
     "\nS1 : "+this.marks[0]+
     \nnS2: "+this.marks[1]+
     \nnS3: "+this.marks[2]+
     \n '\nS4 : "+this.marks[3]+
     '' \nS5 : '' + this.marks[4]);
  }
}
package LAB6;
import LAB6.CIE.*;
import LAB6
.SEE.*;
import java.util.Scanner;
public class Inform{
  public static void main(String[] args) {
   Scanner sc =new Scanner(System.in);
   System.out.println("no of Students");
```

```
int n=sc.nextInt();
   Internals[] internals=new Internals[n];
   Externals[] externals = new Externals[n];
   for(int i=0;i< n;i++)
     System.out.println("USN");
     String usn = sc.nextLine();
sc.next();
     System.out.println("NAME");
     String name=sc.nextLine();
sc.next();
     System.out.println("SEM");
     String sem =sc.nextLine();
     sc.next();
     System.out.println("Eneter CIE marks");
     int[] Cmarks=new int[5];
     int[] Smarks=new int[5];
    for(int j=0; j<5; j++){
      Cmarks[j]=sc.nextInt();
     System.out.println(" Enter SEE marks");
    for(int j=0; j<5; j++){
       Smarks[j]=sc.nextInt();
     internals[i]=new Internals(usn,name,sem,Cmarks);
     externals[i]=new Externals(usn,name,sem,Smarks);
    System.out.println("Final Marks");
     for(int j=0; j<5; j++){
       int final1=internals[i].marks[j]+externals[i].marks[j];
       System.out.println("Sub"+(i+1)+"marks "+final1);
   }
```

```
no of Students
USN
072
NAME
murali
SEM
Eneter CIE marks
89
89
89
89
Enter SEE marks
89
89
89
89
89
CIE MARKS ---
S1: 89
S2: 89
S3 :
     89
S4: 89
S5: 89
SEE MARKS -----
S1: 89
S2: 89
S3: 89
S4: 89
S5 : 89
Final Marks
Sub1marks 178
Sub1marks 178
Cubimanle 170
```

#### Program 7

We have created an interface named Polygon. It includes a default method getPerimeter() and an abstract method getArea(). We can calculate the perimeter of all polygons in the same manner so we implemented the body of getPerimeter() in Polygon. Now, all polygons that implement Polygon can use getPerimeter() to calculate the perimeter. However, the rule for calculating the area is different for different polygons. Hence, getArea() is included without implementation. Any class that implements Polygon must provide an implementation of getArea().

```
import java ub 1 . scanner :
          interface Polygon of
             Scannel co - new Cannel Crystem in):
             void get Area();
             void
             default voil gettermeter () [
                  System out printent" sites size and league");

Ent size = co next lates;

Ent length: sc next lates;
         3
                Square implements Polygon &
         class
                int size -
                int long to ;
                Square ( int size , int langter) {
                    -16il-size = size;
                     this legth tength;
                public wild get Area() {
                     Systemout printle ("ADEA = " + length > length);
                public word get Permeteccif
                      Eystemout . printly ( " PERIMETED: " +4 *
                                                        lengter) =
      class
              Hexagon
                       implements Polygon &
                int 8 12c ;
                int length,
                Hexagon ( ent size, ent lengths) {
                          - Hic size = size
                           - Ris length = length ,
public void get Areal) &
                                             (34 Math & 59.14(3) the)
             -this size =
        System out println ("CAREA:"
public
                getPerimeter() {
          System out print la ("PERIMETER: "+ 6x lengte);
public class Mainy &
        public static void main (Storages) {
              Polygon p new Equae (4.7);
            p get Areal >-,
               p-getPerimeterc);
              Polygon pi = new Heragon (5,9);
               Progethieses;
              Pl getPerimeter ();
                  Coping the sale cold
```

```
package LAB7;
import java.util.Scanner;
interface Polygon {
  Scanner sc = new Scanner(System.in);
  void getArea();
  default void getPerimeter() {
     System.out.println("Enter the size and enter the length");
     int size = sc.nextInt();
     int length = sc.nextInt();
     System.out.println("Perimeter: " + size * length);
}
class Square implements Polygon {
  int size;
  int length;
  Square(int size, int length) {
     this.size = size;
     this.length = length;
  }
  public void getArea() {
     System.out.println("AREA: " + length * length);
  public void getPerimeter() {
     System.out.println("Perimeter: " + 4 * length);
  }
class Hexagon implements Polygon {
  int size;
  int length;
  Hexagon(int size, int length) {
     this.size = size;
     this.length = length;
  }
  public void getArea() {
     System.out.println("AREA: " + (3 * Math.sqrt(3) * length * length) / 2);
```

```
public void getPerimeter() {
    System.out.println("Perimeter: " + 6 * length);
}

public class Main4 {
    public static void main(String[] args) {
        Polygon p = new Square(4, 7);
        p.getArea();
        p.getPerimeter();

    Polygon p1 = new Hexagon(5, 9);
        p1.getArea();
        p1.getPerimeter();
}
```

```
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB7/Main4.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB7.Main4
AREA: 49
Perimeter: 28
AREA: 210.4441731196186
Perimeter: 54
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB7/Main.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB7.Main
Dog barks
Dog eats bones
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB7/Main1.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB7.Main1
Sedan is starting
Sedan is driving
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB7/Main2.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB7.Main2
Dog barks
Dog eats bones
C:\Users\mural\OneDrive\Desktop\java_lab>
```

#### **Program 8**

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 uses both father and son's age and throws an exception if son's age is >=father's age.

Algorithm: Class AgeException extends Runtime Exception (
AgeException (string van 1) & Supa (vags); class Stathers intage; fathe, (intage) throws Alex Exception f if (age (20) throw new AgeException ("Age >0"). this age = age; System out printin (" fother age is " + this age); 3 3 class Son extends Father { Son (int Page, int Sage) throws Age Exception ( super (fage); if (Sage > = Fage) throw new Age Exception ( " Fother age , son age "): System out printin (" Fother age is" + fage +" and Son age is 11 4 Sage) public class public static void main (string () angs) catch ( Exception & f System out print In (e);

### Code:

package LAB8;

```
class AgeException extends RuntimeException{
 AgeException(String msg){
 super(msg);
class Father{
 int age;
 Father(int age) throws AgeException{
   if (age<=0) throw new AgeException("Age should be greater than zero");
   else{
     this.age=age;
     System.out.println("Father age is "+this.age);
class Son extends Father{
 Son(int Fage,int Sage)throws AgeException{
 super(Fage);
 if(Sage>=Fage) throw new AgeException("Father age should be greater than Son age");
System.out.println("Father age is "+ Fage+ " and Son age is "+Sage);
public class Main{
public static void main(String[] args){
     Son son1 = new Son(23,25);
  catch(Exception e){
     System.out.println(e);
```

```
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB8/E1.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB8.E1
File test.txt is missing
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB8/E3.java
C:\Users\mural\OneDrive\Desktop\java_lab>java LAB8.E3
java.lang.ArithmeticException: / by zero
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB8/E4.java
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB8.E4
/ by zero
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB8/Main.java
C:\Users\mural\OneDrive\Desktop\java_lab>javac LAB8.Main
Father age is 23
LAB8.AgeException: Father age should be greater than Son age
C:\Users\mural\OneDrive\Desktop\java_lab>
```

#### **Program 7**

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.

Algorithm: public class Mainte & public static void main ( string ( ? angs) { Thread to = naoThread (0) > f try & for (" o: 120; 143; 141) { System out println ("BMSCE"); Thread . Sleep (10, 1000); } catch ( Interrupted Exception e) { System out printen (" to was interrupted"); 3); Thread to new Thread (1) of try E for (int iso; ic 3; in) { " System out privato ("CSF"); Thread steap (3 x 1000); 3 color (Interrupted Exception e) { Systemout prior In ( 100 was interrupted"); 3): to chart 3 +2" Stait; 3 orap BMSCE CSC CSC

Code:

```
package LAB9;
class BMSCollegeThread extends Thread {
 public void run() {
   try {
      while (true) {
        System.out.println("BMS College of Engineering");
        Thread.sleep(10000);
   } catch (InterruptedException e) {
     System.out.println("BMS College Thread Interrupted");
 }
class CSEThread extends Thread {
 public void run() {
   try {
     while (true) {
        System.out.println("CSE");
        Thread.sleep(2000);
   } catch (InterruptedException e) {
     System.out.println("CSE Thread Interrupted");
 }
public class TwoThreads {
 public static void main(String[] args) {
   BMSCollegeThread bmsThread = new BMSCollegeThread();
   CSEThread cseThread = new CSEThread();
   bmsThread.start();
   cseThread.start(); }
}
```

#### <u>Program 8</u>

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.

Algorithm: public static void main (String() arigs) { INFORMATION MESSAGE String si = Joption Pane show Input malog Strong (2 = JOption Pane show lapout Dialog try ? int mi = Integer place Int (si); int n2 = Integer - parse lat (se); realt = mi/ (Plant) ms. JOphionPane - show Massage Dialog (mull, "Result" result, "Result" JUPHORPARE-INFORMATION MESSAGE); & catch (Anthometic Exception e) { (null, "can't divide by zero", to JOPHON . INFORMATION MESSA Granch (Exception e) { is Run : Palse , System out print In ( " invalid inputs) 4

Code: package LAB10;

import javax.swing.JOptionPane;

```
public class add {
  public static void main(String[] args) {
     boolean isRun = true;
     JOptionPane.showMessageDialog(null, "Caluclator", "Division by zero",
JOptionPane.INFORMATION MESSAGE);
     while (isRun) {
       String s1 = JOptionPane.showInputDialog("enter a no ");
       String s2 = JOptionPane.showInputDialog("enter second no");
          int n1 = Integer.parseInt(s1);
          int n2 = Integer.parseInt(s2);
         float result = n1 / (float) n2;
          JOptionPane.showMessageDialog(null, "result:" + result, "Result",
JOptionPane.INFORMATION_MESSAGE);
       } catch (ArithmeticException e) {
         JOptionPane.showMessageDialog(null, "cant divide by zero:", "Exception",
              JOptionPane.INFORMATION_MESSAGE);
       } catch (Exception e) {
          isRun = false;
          System.out.println("invalid inputs ");
       }
    }
}
```

