VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.



LAB REPORT on

Object Oriented Java Programming (23CS3PCOOJ)

Submitted by

Vaibhav Dhar (1BM23CD068)

in partial fulfillment for the award of the degree of BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



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 **Vaibhav Dhar (1BM23CD068)**, 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.

Dr. Seema Patil	Dr. Jyothi S Nayak
Assistant Professor	Professor & HOD
Department of CSE, BMSCE	Department of CSE, BMSCE

Index

Sl. No.	Date	Experiment Title	Page No.
1	28/10/24	Implement Quadratic Equation	4
2	28/11/24	SGPA Calculation	7
3	04/11/24	Display Book Details	12
4	11/11/24	Using Abstract Class Shape	16
5	11/11/24	Bank Account Storage	20
6	18/11/24	Creating Packages CIE and SEE	29
7	28/11/24	Handling Exceptions in Inheritance Tree	36
8	28/11/24	Threads Creation	40
9	28/11/24	User Interface Creation	43
10	28/11/24	Demonstrating IPC and Deadlock	49

Github Link:

https://github.com/Vaibhav1830/Java-Lab

Program 1

Implement Quadratic Equation

```
Code- import java-utsl. Scanner;
     class Quadratic
     2 sect orbic;
       double n1, n2,d;
       wid getd()
      { Scanner 5 = neur Scanner (System in);
        System.out. println ("Enter coefficients of a,b,c");
       a = 5. rest Ted();
       b= 5. next Int ();
       C = S. nent Ird();
     wid compute ()
     { while (a==0)
       { System.out. println ("Not a quadratic equation");
        System. out println ("Enter non-zero value for a");
        Scanner 5 = new Scanner (System.in):
         a = s-hent Int();
      d= b*b-4*a*c;
      if a == 0)
     { nl=(-b)((2*a);
       System. out. printle ("Roots are real and equal");
       System.out. println ("Root = "+1");
    else if (d>0)
   { 12 | = ((-b)+ (Math. sgrotd(d))) / (double) (2 + a);
     22= ((-b)- (Moth.sqrt(d))) (Houble) (2*a);
     System.out. println ("Roots are read and distinct");
    System. out- println ("Root 1 = "+ n | + "Root 2 = "+ n 2);
  else if (dco)
   [ System. out. brintln ("Roots are imaginary");
    n1= (-b) 1(2+a)!
   2= Math. sgrt(-d)/(2+a);
   ) ystem.out. brintly ("Root!=" +N+"+i"+ 2);
```

```
import java.util.Scanner;
class Quadratic
{ int a, b, c;
   double r1, r2, d;
   void getd()
   {
      Scanner s = new Scanner(System.in);
      System.out.println("Enter the coefficients of a,b,c");
      a = s.nextInt(); b = s.nextInt(); c = s.nextInt();
}
void compute()
{
      while(a==0)
      {
            System.out.println("Not a quadratic equation");
            System.out.println("Enter a non zero value for a:");
            Scanner s = new Scanner(System.in);
}
```

```
a = s.nextInt();
    d = b*b-4*a*c;
    if(d==0)
      r1 = (-b)/(2*a);
      System.out.println("Roots are real and equal");
      System.out.println("Root1 = Root2 = " + r1);
    else if(d>0)
      r1 = ((-b)+(Math.sqrt(d)))/(double)(2*a);
      r2 = ((-b)-(Math.sqrt(d)))/(double)(2*a);
      System.out.println("Roots are real and distinct");
      System.out.println("Roo1 = " + r1 + "Root2 = " + r2);
    else if(d<0)
      System.out.println("Roots are imaginary");
      r1 = (-b)/(2*a);
      r2 = Math.sqrt(-d)/(2*a);
      System.out.println("Root1 = " + r1 + " + i"+r2);
      System.out.println("Root1 = " + r1 + " - i"+r2);
  }
class QuadraticMain
 public static void main(String args[])
    Quadratic q = new Quadratic();
    q.getd();
    q.compute();
  }
 "C:\Program Files\Java\jdk-23\bin\
                                              "C:\Program Files\Java\jdk-23\bin\
                                             Enter the coefficients of a,b,c
 Enter the coefficients of a,b,c
2 3 1
                                             Roots are imaginary
 Roots are real and distinct
                                             Root1 = -1.0 + i1.4142135623730951
Roo1 = -0.5 Root2 = -1.0
                                             Root1 = -1.0 - i1.4142135623730951
 Process finished with exit code 0
                                             Process finished with exit code
```

SGPA Calculation

```
Code- import java. wil. Scanner:
     class Subject;
    { int subMarks, credits, grade;
                System out pointly ("Roots are real and equal")
   class Student
                      System out frietle ( " Hoot = Roots = "+ 11")
   ( Subject subject [];
    String name i
                    (C+5) (Charles Special (S(6) ) ( (do)) = (d-3) = (d-3)
    String usn;
                       12 = ((-6)- (Mothogat (4,2)) (Acolle) (2 %);
    double SGPA;
                        System. out friedla (" Posts one work and
    Scanner Si
             System out printle ("Roots" + m + "Root 25" + n2);
   Student ()
  { int i ;
    Subject = new Subject (9);
   for (i=0; icq; i++) ( compani and tool ) water the
   Subject (i) = new Subject ();
  S= new Scanner (System.in);
```

```
wid get Student Details ()
     { System.out. print ("Enter your name:");
       hame = s. hent();
       System. out print ("Enter your USN:");
       asn = s. next();
     wid getManker()
    ( for (int i=0; i=q;i++)
      { System out pint ("Enter monks for Subject" + (i+1) + ":");
        Subject [i] . subject Marks = s. next Int();
        System out print ("Enter credits for Subject" + (+1)+":");
        subject (i). oredits = s. neut Int();
        Subject (i). grade = (Subject (i). subject Marks / 10) +1;
        of (subject (i). grade == 11)
          subject (i) grade = 10;
       if (subject (i). grade <= 4)
          Subject (i). grade = 0;
   wid compute SCIPA ()
   { int effective Score= 0;
     int total (nedits=0;
                                        I is house you there would
     for (int i=0;icq;i++)
     { effectivescore += (subject (i) grade * subject (i) credits);
        total (redits + = subject (i) oredit;
     SGPA = (double) effective Score / (double) total (redits;
Class Main
{ bublic static A wid main (String args [])
  } Student s = new Student();
    SI- get Student Details ();
    sl.get Marks();
     sl.computesGPA();
    System.out. brindle ("Name:" + sl. name);
   System.out. println ("USN: "+ sl. usn);
 2 System. out. privaler ("SGPA:"+s1. SGPA);
```

```
Output-
   Enter your name: Vaibhar
   Exter your USN: 18M23CD068
   Enter marks for subject 1:80
  Enter oredits for subject 1: 4
  Enter marks for subject 2: 85
  Enter credits for subject 2: 3
  Enter marks for subject 3:89
  Enter credits for subject 3: 4
  Enter marks for subject 4: 90
 Enter credits for subject 4: 3
 Enter marks for subject 5: 91
 Enter mo credits for subject 5: 3
 Enter marks for subject 6: 87
Ender credits for subject 6: 1
Enter marks for subject 7: 89
Enter credits for subject 7:1
                                      Subject (1). grade 0;
Enter marks for subject 8: 88
Enter you credits for subject 8:1
                                           () Agolastadono
Enter mortes for subject 9:92
                                          epartolastantes
Enter credits for subject 9: 1
Name: Vaibhay
USN: LBM2200682 Sept (12 topped) = + and lavetes
SGPA: 9.363636
```

```
import java.util.Scanner;
class Subject
  int subjectMarks;
  int credits;
  int grade;
class Student
```

```
Subject subject[];
String name;
String usn;
double SGPA;
Scanner s;
Student()
{
  int i;
  subject = new Subject[9];
  for(i=0;i<9;i++)
    subject[i] = new Subject();
  s = new Scanner(System.in);
void getStudentDetails()
  System.out.print("Enter your Name: ");
  name = s.next();
  System.out.print("Enter your USN: ");
  usn = s.next();
void getMarks()
  for(int i=0;i<9;i++)
     System.out.print("Enter marks for subject "+(i+1)+":");
     subject[i].subjectMarks = s.nextInt();
     System.out.print("Enter credits for subject "+(i+1)+":");
     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 effectiveScore = 0;
  int totalCredits = 0;
  for(int i=0; i<9; i++)
     effectiveScore += (subject[i].grade*subject[i].credits);
    totalCredits += subject[i].credits;
```

```
SGPA = (double)effectiveScore/(double)totalCredits;
class Student SGPA
 public static void main(String args[])
   Student s1 = new Student();
   s1.getStudentDetails();
   s1.getMarks();
   s1.computeSGPA();
   System.out.println("Name: "+s1.name);
   System.out.println("USN: "+s1.usn);
   System.out.println("SGPA: "+s1.SGPA);
 }
 "C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
Enter your Name: Vaibhav
Enter your USN: 1BM23CD068
Enter marks for subject 1:82
Enter credits for subject 1:4
Enter marks for subject 2:85
Enter credits for subject 2:4
Enter marks for subject 3:91
Enter credits for subject 3:3
Enter marks for subject 4:88
Enter credits for subject 4:3
Enter marks for subject 5:94
Enter credits for subject 5:3
Enter marks for subject 6:99
Enter credits for subject 6:1
Enter marks for subject 7:87
Enter credits for subject 7:1
Enter marks for subject 8:91
Enter credits for subject 8:1
Enter marks for subject 9:87
Enter credits for subject 9:1
Name: Vaibhav
USN: 1BM23CD068
SGPA: 9.380952380952381
Process finished with exit code 0
```

Display Book Details

```
Algorithm:
```

```
import java. will- Scarner;
   class Book &
       String name, author;
        int frice, numbages ;
        Book (String many String author, int price, int num Pages) E
            this name = name;
            this author = author;
            this price: price;
            this numbages a numbages;
  public String to String () {
        String book Details: " Book name: " + this name +" \n"+ "Author name:"
                           +" In" + " Auth His author + " In" + " Price!" +
                            "th"+ this price + "In" + " No. of pages! "+
                            this numbages +" In";
               book Petails ;
public class week 3 {
       PSUM (String [] args) {
         Scanner 5 = new Scanner (System.in);
         System.out. printle ("Enter some o no of books: ");
         int n = S. nentInt ();
         Book [] books = new Book (n)
         for lint i=0; icn; i++) {
             Sop ("Enter name of book "+ (i+1)+":");
             Strong name = S. next ();
            Sol ("Enter author of book" + (i+1) +":");
            String author = s. nent();
             Sop (" Easter price of book "+ (i+1) + 6":");
```

```
int Numlages = 5-next Int ();
           books (i) = new Book (name, author, price, numlager)
       SOP ("In Book Details: ");
       S. close ();
  Outfut -
  Enter number of books: 1
  Enter name of book 1: Arun
  Enter author of book 1: Chetan Bhagat
  Enter no. of pages of book 1:800
Book details:
Book Name: Arun
  Author Name: Chetan Bhagat
  Price: 499
  No. of pager: 800
```

```
import java.util.Scanner ;

class Main{
  public static void main(String args[]){
    int n;
    System.out.print("Enter the number of books:");
    Scanner sc = new Scanner(System.in); n = sc.nextInt();
    sc.nextLine();
    Book books[] = new Book[n];
    for(int i = 0; i < n; i++){
        System.out.print("Enter the book name: ");
        String name = sc.nextLine();
        System.out.print("Enter the author name: ");
</pre>
```

```
String author = sc.nextLine();
       System.out.print("Enter the price of the book: ");
       int price = sc.nextInt();
       System.out.print("Enter the number of pages in the book: ");
       int numPages = sc.nextInt();
       sc.nextLine();
       books[i] = new Book(name,author,price,numPages);
    System.out.println("");
    for(int i = 0; i < n; i++){
       System.out.println(books[i].toString());
    System.out.println("Vaibhav Dhar");
    System.out.print("1BM23CD068");
    sc.close();
  }
class Book{
  String name, author;
  int price, numPages;
  Book(String name, String author, int price, int numPages){
    this.name = name:
    this.author = author;
    this.price = price;
    this.numPages = numPages;
  }
  public String toString(){
    String name ,author , price,numPages ;
    name = "Book name: " + this.name + "\n";
    author = "Author name: " + this.author + "\n";
    price = "Price: " + this.price + "\n";
    numPages = "Number of pages: " + this.numPages + "\n";
    return name + author + price + numPages;
}
```

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files` Enter the number of books:2
Enter the book name: Wings of Fire
Enter the author name: APJ Abdul Kalam

Enter the price of the book: 550

Enter the number of pages in the book: 1002

Enter the book name: Fire and Ice
Enter the author name: Robert Frost
Enter the price of the book: 450

Enter the number of pages in the book: 889

Book name: Wings of Fire

Author name: APJ Abdul Kalam

Price: 550

Number of pages: 1002

Book name: Fire and Ice Author name: Robert Frost

Price: 450

Number of pages: 889

Vaibhav Dhar 1BM23CD068

Process finished with exit code 0

Using Abstract Class Shape

```
import java util. Scanner:
  abstract class Shape:
   int dil;
  int diz;
  public Shape () {
      this dil = 0;
     this. diz = 0;
 public Shape ( int dil, int diz) {
      thir dil=dil;
     this. diz=diz;
 public abstract void printArea()1
Class Rectangle extends Shape &
        public Rectangle (int li int w) {
           diliw;
        public hoid print Area () [
             it area = dil + dizi
           SOP("Area of rectaple:"+ area);
class Triangle extends Shape {
    bublic Triangle (int b, int h) {
           dil=b;
          d'2=h'
```

```
public wind print Area () {
              double crea = 0.5+dil+diz)
              SOPE "Area of triangle ""+ once);
  closs Circle extends Shape {
         public Cincle (int x) {
              dil=n;
              di2=0;
         public wid printArea () {
             double drea = Math. PI * dil *dil;
            SOP ("Ane of while: "+ area);
public class Weeked {
     Psum (String () args)
     { Scanner Scannor: her Scannor (System.in);
          SOP ("Endor length and breadth for rectangle");
          int l= scenner. next Zort ();
          int be Scanner. hent 2nt();
          Shape rectargle = new Rectalgle (lib);
          rectangle- print Area ();
         SOP(" Enter base and height of triangleis;
         int b = scenner. next Int ();
         int h = Scanner. next Int();
         Shape triangle = new Triangle (bh);
         triangle print Aree ();
         Sop ("Enter radius of circle!");
         int r = sconner - next Int ();
        Shape circle = new Circle (n);
        Cincle printArea ();
         Scarner. close (1) 3
```

```
Enter length and breadth of rectangle:

12
10
Anes of rectangle: 120
Enter bare and height of transgle:

4
Anes of triangle: 12.0
Enter radius of circle:

Anes of circle: 153.93804
```

```
import java.util.Scanner;
class Main{
  public static void main(String[] args){
    Rectangle ob2 = new Rectangle();
    Triangle ob1 = new Triangle();
    Circle ob3 = new Circle();
    ob2.printArea();
    ob1.printArea();
    ob3.printArea();
    System.out.println("Vaibhav Dhar");
    System.out.print("1BM23CD068");
  }
}
abstract class Shape{
  Scanner sc = new Scanner(System.in);
  int dimension1, dimension2;
  abstract void printArea();
class Rectangle extends Shape{
  Rectangle(){
    System.out.println("Enter the dimensions of the rectangle(Length and Breadth): ");
    dimension1 = sc.nextInt();
    dimension2 = sc.nextInt();
```

```
void printArea(){
    System.out.print("The area of the rectangle is = ");
    System.out.println(dimension1*dimension2);
  }
}
class Triangle extends Shape{
  Triangle(){
    System.out.println("Enter the dimensions of the triangle(base and height): ");
    dimension1 = sc.nextInt();
    dimension2 = sc.nextInt();
  void printArea(){
    System.out.print("The area of the Triangle is = ");
    System.out.println(0.5*dimension1*dimension2);
  }
}
class Circle extends Shape{
  Circle(){
    System.out.println("Enter the dimension of the circle(radius): ");
    dimension1 = sc.nextInt();
  }
  void printArea(){
    System.out.print("The area of the Circle is = ");
    System.out.println(3.1415926535897*dimension1*dimension1);
  }
 "C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
 Enter the dimensions of the rectangle(Length and Breadth):
 10 20
 Enter the dimensions of the triangle(base and height):
15 20
 Enter the dimension of the circle(radius):
 The area of the rectangle is = 200
 The area of the Triangle is = 150.0
 The area of the Circle is = 615.7521601035811
 Vaibhav Dhar
 1BM23CD068
 Process finished with exit code 0
```

Bank Account Storage

```
Algorithm:
            juin wil Scorer;
       import
       class Account {
             Strong cur tomer name;
             int acc-no;
             double balance;
             public Account (String customer-name, int account
                           double balance ) {
                 this. Customer-name = customer-name !
                 this acc-no = acc-no;
                this balance = balance;
            public double getBalance () {
                return balance;
           3
          public void deporit (double amount) {
              if Comment su) {
                    balance + = amount;
                    Sop ("Deposited: "tamount);
              else {
                   SOP ("Deposit amount must be positive");
          3
        public withdraw (double amount)
               if (amount c=getBalance ()) {
                     balance samount;
               Sop ("Withdrow: "+ amount + "balance is: "+
                      belone);
       0
              else {
                 SOP ("Insufficient freds');
```

```
public word display Balance () [
                    SOP ("Corrent balance: "+ balance);
                   Savings Account extends Account {
                      double interest Rate;
                      public Savings Account (String customer name,
                       int account Mumber, double initial Ralance, double
                         interest Rate) {
                              Super (Customer Name, account Number,
                                    initial Palance );
                              this interst Rate = interest Rate;
                    public wild compute and Defosit Interest () {
                    double interest = get Balance () interest Rete /100);
                   deporit (interest);
            class Current Account extends Accounts
                 double minimum Balance, services Charge i
              public current Account (String Customer Name, int
                  account Number double initial Balance, double minBalance
                 double service charge) {
                  Super (Customer Name, account Number, initiallale
                      this. minimum Balance = minimum Balance;
                  this. service Charge = service Charge ;
   "The format panel manight pine
bublic wid
      Check Minimum Balana () {
            if (getBalance () a minimum Balance) {
            Sopl " Belone is below minimum ");
           balance - = Service Charge ;
```

```
SOP(" reducted Service Change!" + service Change );
        Sop(" Ralance after deduction;" + balance);
public class Bank &
     krom (Strong (3 args) {
        Scanner Sc= new Scanner (System. in);
         Sop(" Enter curtomer name: ");
         String name = sc. nortline;
         SOP (" Enter arc no: ");
      int accno = sc- west (ine (); hentInd();
         SOP(" Forker initial balance:");
      double Palence = Sc. next Pouble ();
        Sop (" Enter min balence " ");
         double minimum-balance = sc. next Pouble ();
         Sop (" Enter interest rate: ");
        double interest-notes sc. next Pouble ();
        SOP("Enter Service charge!");
        double service-charge = SC. next Pouble();
       Sop ("Enter choice: In 1. Current arc la 2. Savingsacc"
     int che Sc. next Int();
      Sop("Customer name is!" + name + "In Account
       number: "tarc-not" In Vaibhar - 1 BM 23 Coof8 ");
 Switch (ch) {
   Care (1):
    Sop (" Account is current type ");
          Current account ca : new Current Account (name acc No.
          balance minimum-balance service Charge );
            SOP ("Forter choice: In 1. deposit) In 2 Withdraw /a
            3. Deposit balance ");
      int c = sc. hent Int();
            Ca. chechtinimum Balence ();
```

```
if (c==DE
        SOP (" Enter amount to be deported!");
        double amount = Sc. next Pouble ();
         Ca. deposit (ant) 13
    elre if (c==2) {
      SOP(" Enter amount to be withdrawn: ");
       double ant = rc. next Double();
      ca. withdraw (ant); }
  ele if (c== 3) {
     co. display balance (1; 3
 elre
    System exit (0);
 3 while (true);
Case (2);
 Sop(" Account is sains type ");
Savings Account sa: new Savings Account (name, acc-no, balance,
do {
   Sop(" Enter choice: In 1. Deposit In 2. Withdrew In 3. Disple
  int d = sc. next Int ()!
  if (c1==1) {
  Sop ("Enter amount to be deposited:")
  double amt = sc. hent Pouble ();
  Sa. deposit (ant); 3
  else if (ct=2)
 { SOP (" Enter amount to withdrawi")
 double ant: Sc. next Double ();
 Sa. withdraw (and); 3
 else if (cl==3)
 E sa. compute And Deposit Interest ();
    Sa · display Balance (); }
```

```
else

Esystem exit (0); }

Juhile (true);

Teperited: 500:0

Ralane: 500.0

Interest: added: 25.0

Withdran: 200.0

Balance: 300:0

Current Account:

Oct orded: 1000:0

Withdran: 1000:0

Withdran: 1000:0

Withdran: 1000:0

Withdran: 1000:0

Balance: 0.0
```

import java.util.Scanner; public class Bank { static Scanner sc = new Scanner(System.in); Account ob1; void createAccount() { String customer; int account; String type; int initBal; System.out.print("Enter the customer name: "); customer = sc.nextLine(); System.out.print("Enter account Number: "); account = sc.nextInt(); sc.nextLine(); // Consume the newline System.out.print("Enter Account type (Savings or Current): "); type = sc.nextLine(); System.out.print("Enter the initial Balance: "); initBal = sc.nextInt();

```
if (type.equals("Savings")) {
       ob1 = new Savings(customer, account, initBal);
     } else {
       ob1 = new Current(customer, account, initBal);
  }
  public static void main(String[] args) {
    Bank bank = new Bank();
    bank.createAccount();
    while (true) {
       System.out.println("-----");
       System.out.println("1. Deposit 2. Withdraw");
       System.out.println("3. Compute interest");
       System.out.println("4. Display account details");
       System.out.println("5. exit ");
       int choice = sc.nextInt();
       switch (choice) {
         case 1:
            bank.ob1.deposit();
            break;
         case 2:
            bank.ob1.withdraw();
            break;
         case 3:
            if (bank.ob1 instanceof Savings) {
              ((Savings) bank.ob1).computeInterest();
            } else {
              System.out.println("Interest computation is only available for Savings accounts.");
            break;
         case 4:
            bank.ob1.display();
            break;
         case 5:
            break;
         default:
            System.out.println("Invalid choice. Please try again.");
       if(choice == 5) break;
class Account {
  String customerName;
  int accountNumber;
  int balance;
```

```
Account(String customer, int accountNum, int bal) {
     customerName = customer:
    accountNumber = accountNum;
    balance = bal:
  }
  void deposit() {
    System.out.print("Enter the amount to deposit: ");
    int amt = Bank.sc.nextInt();
    balance += amt:
    System.out.println("Deposited: " + amt + ", New Balance: " + balance);
  void withdraw() {
     System.out.print("Enter the amount to withdraw: ");
    int amt = Bank.sc.nextInt();
    if (balance - amt < 0) {
       System.out.println("Insufficient Balance to withdraw the given amount.");
     } else {
       balance -= amt;
       System.out.println("Amount of " + amt + " withdrawn successfully. Current Balance is " +
balance);
  }
  void display() {
    System.out.println("The Balance in the account is " + balance);
  }
class Savings extends Account {
  double interestPercent;
  Savings(String customer, int accountNum, int bal) {
     super(customer, accountNum, bal);
    System.out.print("Enter the interest percentage on the account: ");
    interestPercent = Bank.sc.nextDouble();
  void computeInterest() {
    balance += balance * (interestPercent / 100);
    System.out.println("Amount after applying interest is: " + balance);
  }
class Current extends Account {
  int minBalance = 1000;
  Current(String customer, int accountNum, int bal) {
     super(customer, accountNum, bal);
```

```
}
 void withdraw() {
   System.out.print("Enter the amount to withdraw: ");
   int amt = Bank.sc.nextInt();
   if (balance - amt < minBalance) {
     System.out.println("Insufficient Balance to maintain the minimum required.");
   } else {
     balance -= amt;
     System.out.println("Amount of " + amt + " withdrawn successfully. Current Balance is " +
balance);
   }
 }
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
Enter the customer name: Vaibhav
Enter account Number: 12
Enter Account type (Savings or Current): Current
Enter the initial Balance: 1500
-----

    Deposit
    Withdraw

3. Compute interest
4. Display account details
5. exit
Enter the amount to deposit: 200
Deposited: 200, New Balance: 1700
-----
1. Deposit 2. Withdraw
3. Compute interest
4. Display account details
5. exit
Enter the amount to withdraw: 400
Amount of 400 withdrawn successfully. Current Balance is 1300
-----MENU-----
1. Deposit 2. Withdraw
3. Compute interest
4. Display account details
5. exit
The Balance in the account is 1300
```

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
Enter the customer name: Vaibhav
Enter account Number: 11
Enter Account type (Savings or Current): Savings
Enter the initial Balance: 1000
Enter the interest percentage on the account: 15
-----MENU------
1. Deposit 2. Withdraw
3. Compute interest
4. Display account details
5. exit
Amount after applying interest is: 1150
-----MENU-----
1. Deposit
            2. Withdraw
3. Compute interest
4. Display account details
5. exit
Enter the amount to deposit: 400
Deposited: 400, New Balance: 1550
-----MENU-----
1. Deposit 2. Withdraw
3. Compute interest
4. Display account details
5. exit
The Balance in the account is 1550
-----MENU------
1. Deposit 2. Withdraw
3. Compute interest
4. Display account details
5. exit
Enter the amount to withdraw: 200
Amount of 200 withdrawn successfully. Current Balance is 1350
------
1. Deposit 2. Withdraw
3. Compute interest
4. Display account details
5. exit
Process finished with exit code 0
```

Creating Packages CIE and SEE

```
pachage
         cie j
   public class Standard 1 &
       String name;
       i'nt sem;
       public Student (String usn, String having int sem) {
            this usneusn;
            this name = name;
       public word display Student Info () {
           Sout ("USN: "+ usn);
          Sout ("Name:" theme);
          Sout ("Semester:"+ sem);
package (ZE;
public class Internals extends student/ {
    int () internal Marks : new int (5); int () internal Mark
    public Internals (String was, String hameint sen) {
        Super (wn, have, sem);
        this. internal Marks: internal Marks;
            14 () sofered Marter (60,00, 56,65,50);
```

```
public wid display Indernal Marks () &
                     Sout ("Internal marks: ");
                     for (int Menti: internal Ments) [
              Sout (Marks + " ");
                      Sout ();
       package see;
        import cie. Studentl;
       public class Entornal extends Student 1 8
         int () enternal Merks = new int (5);
          public Entered (String us, String name, int sem, int () externs
          super (us, name, son);
              this. external Merks = external Merks 6;
        public roid display Enternal Merks () {
            Sout ("Enternal Morks: ");
            for (int mark: external Merks ) [
                Sout (mark + " ");
             Sout(); (Case " : stynol" ) took
        3
 3
import cie. Internals;
import see. Extends; 3 habit about a second was sided
public Class Main 1 & Colon was short another Colon
   ps vm (string () args) {
     ind h=1;
     int () internal Menhs 1 = { 20, 30, 25, 28, 223;
     int () enternal Merks 1 = $60,70, 56,65,503;
```

```
student / Entonal 2 new Entonal
                           ("18123 0068", "Vaibhae", 3, internel Marks);
                  int () internel ments 2 = { 18,25, 20,23,283;
                  int[) external mortes 2: { 50,65, 60,58,453;
                  Internals Student 2 Internal: new Internals
                           (" (BM2) C5001", "Aadit", 2, intornal Merhs);
                  Sout (" Student I Info: ");
                  Student Internal . display Student Info ():
                  Student I Internal. display Internal Morhol);
                  Student 1 Enternal . display Inter Enternal Murhs ();
                  -17
                  int () final Marks 1 = calculate Final Marks
3(1
                       (Student | Internal internal Marks, Student | External. Enternal marks);
                   display Final Marks ( Ginel Marks 1);
            public Static 14 (7 calculate Finel Marks
            (int [] interval Marks, int[? Enternal Marks)
              { int () finel Mushs e new int(5);
              for (int i=0; i < 5; i++) {
                  final Menhs G): internal Menhs G) + external Menhs G);
               reduce find Merks;
     public state void display Final Marks (int () final Marks) {
         Sout ( " final marks (Intonal + Enternal): ");
        for Cint Merh! finel Merhs) {
         Sout () )
```

31

```
Output

Enter no. of students: 1

Enter details of students:

USN: 18M23CDOG8

Name: Vaibbour

Sen: 3

Enter C2E ments for S courses:

18 20 15 19 17

Enter SEE ment for S se courses:

40 38 45 42 39

Final mater of Students:

Student 1 - USN: 18M23CDOG8

Name: Vaibbour Sen: 3

Final Marty: 38 39 37 40 36
```

package CIE;

```
import java.util.Scanner;

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

// Method to input student details
    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();
    }
}
```

```
// Method to display student details
  public void displayStudentDetails() {
     System.out.println("USN: " + usn);
     System.out.println("Name: " + name);
     System.out.println("Semester: " + sem);
  }
}
package CIE;
import java.util.Scanner;
public class Internals extends Student {
  protected int[] marks = new int[5];
  // Method to input internal marks
  public void inputCIEmarks() {
     Scanner s = new Scanner(System.in);
     System.out.println("Enter internal marks for 5 subjects:");
     for (int i = 0; i < 5; i++) {
       System.out.print("Enter marks for subject " + (i + 1) + ": ");
       marks[i] = s.nextInt();
  }
package SEE;
import CIE.Internals;
import java.util.Scanner;
public class Externals extends Internals {
  protected int[] marks = new int[5];
                                        // SEE marks
  protected int[] finalMarks = new int[5]; // Final marks
  // Constructor to initialize the marks arrays
  public Externals() {
     marks = new int[5];
     finalMarks = new int[5];
  }
  // Method to input SEE marks
  public void inputSEEmarks() {
     Scanner s = new Scanner(System.in);
```

```
System.out.println("Enter SEE marks for 5 subjects:");
     for (int i = 0; i < 5; i++) {
       System.out.print("Enter SEE marks for subject " + (i + 1) + ": ");
       marks[i] = s.nextInt();
     }
  }
  // Method to calculate final marks (internal + external)
  public void calculateFinalMarks() {
     for (int i = 0; i < 5; i++) {
       finalMarks[i] = marks[i] + this.marks[i]; // Final marks = internal + external
     }
  }
  // Method to display final marks
  public void displayFinalMarks() {
     displayStudentDetails(); // Display student details (inherited from Student)
     System.out.println("Final Marks:");
     for (int i = 0; i < 5; i++) {
       System.out.println("Subject " + (i + 1) + ": " + finalMarks[i]);
  }
import SEE.Externals;
import java.util.Scanner;
class Main {
  public static void main(String args[]) {
     Scanner s = new Scanner(System.in);
    // Input number of students
     System.out.print("Enter number of students: ");
     int n = s.nextInt();
     s.nextLine(); // Consume newline
     Externals[] students = new Externals[n];
    // Input details for each student
     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].inputCIEmarks();
       students[i].inputSEEmarks();
```

```
students[i].calculateFinalMarks();
}

// Display final marks for each student
System.out.println("\nDisplaying final marks for all students:");
for (int i = 0; i < n; i++) {
    students[i].displayFinalMarks();
}

s.close();
}
</pre>
```

Handling Exceptions in Inheritance Tree

```
import java util. Scannor;
        class Wrong AgeException extends Exception E
             public Wrong Age Enception (String message) [
      class fether f
           int age i
          public father (int age) throws Wrong Aze Exception [
            Ef (age 50) {
                throw new B. Wrong Age Enception ("Wrong age ");
class Son entends father & int son Age;
  hubble Son (int father Age, int son Age)
      throws Wrong Age Exception, Son Age Enception
      { Supor (father Age);
        if (son Age) forther Age) {
         throw new SonAgeEncepton ("Son's age eas't be
```

```
this. son Age = son Age ;
         public int getson Age () {
             return SonAge;
    public class Main 7 {
         bsvm (String ags ()) {
          while (true) {
             Scenner sc: her hour Sconner (Tystem.in);
            Sout ("Enter father's age:");
            int Fage = insc. next int ();
             Sout (" Enter son's age: ");
            int S-Age = Sc. neut Int();
             try {
                 Son sone new son (f-Age S-Age);
                 Sout ( " Accepted successfully ");
              cotch (Wrong Age Buception e) {
                   Sout (e. get Message (v);
               Catch (SonAge Exception e) {
                 Sout ( e. get Mersage );
Output-
Enter Fother's age: 45
Enter son's age: 18
Accepted successfully
 Would you like to re-enter details: y
```

```
Erder fother's age: -1

Broter son's age: 18

Wrong Age

Would you like to re-ender details:
```

```
import java.util.Scanner;
class WrongAge extends Exception {
  public WrongAge() {
     super("Age Error");
  public WrongAge(String message) {
    super(message);
  }
class InputScanner {
  Scanner s = new Scanner(System.in);
class Father extends InputScanner {
  int fatherAge;
  public Father() throws WrongAge {
     System.out.print("Enter Father's age: ");
    fatherAge = s.nextInt();
    if (fatherAge < 0) {
       throw new WrongAge("Age cannot be negative");
  }
  public void display() {
    System.out.println("Father's age: " + fatherAge);
  }
class Son extends Father {
  int sonAge;
  public Son() throws WrongAge {
     super();
     System.out.print("Enter Son's age: ");
     sonAge = s.nextInt();
    if (sonAge >= fatherAge) {
       throw new WrongAge("Son's age cannot be greater than or equal to father's age");
     } else if (sonAge < 0) {
       throw new WrongAge("Age cannot be negative");
  }
```

```
System.out.println("Son's age: " + sonAge);
    super.display(); // This calls the Father's display method
  }
public class Exception_Handling{
 public static void main(String[] args) {
    try {
      System.out.println("Vaibhav Dhar\n1BM23CD068");
      Son son = new Son();
      son.display();
    } catch (WrongAge e) {
      System.out.println("Exception: " + e.getMessage());
  }
                                          "C:\Program Files\Java\jdk-23\bin
"C:\Program Files\Java\jdk-23\bin\
                                          Vaibhav Dhar
Vaibhay Dhar
                                          1BM23CD068
1BM23CD068
                                          Enter Father's age: 23
Enter Father's age: 56
                                          Enter Son's age: -1
Enter Son's age: 27
                                          Exception: Age cannot be negative
Son's age: 27
Father's age: 56
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Fi
Vaibhay Dhar
1BM23CD068
Enter Father's age: 43
Enter Son's age: 46
Exception: Son's age cannot be greater than or equal to father's age
```

public void display() {

Program 8

Threads Creation

```
Algorithm:
```

```
RMSCollege extends Thread &
class
   @ override
   public wid run () {
      try {
          while (true) {
         soften ("BMs college of Engineering");
         Thread sleep (10000);
         3
      3 cotch (Interrupted Exception e) {
                 sople (e);
        3
 class (SEThread extends Thread {
     @ Override
     public wid run () {
        try { while (true) {
           20 pln (, (ZE.,);
            Thread. sleep (2000);
      } catch (Interrupted Exception e) {
               Sopen("e);
         3
   3
```

```
public class Threads {
  static class BMSDisplayThread extends Thread {
     public void run() {
       try {
          while (true) {
            System.out.println("BMS College of Engineering");
            Thread.sleep(10000); // Sleep for 10 seconds
       } catch (InterruptedException e) {
          System.out.println(e);
     }
  }
  static class CSEDisplayThread extends Thread {
     public void run() {
       try {
          while (true) {
            System.out.println("CSE");
```

```
Thread.sleep(2000);
      } catch (InterruptedException e) {
        System.out.println(e);
   }
  }
 public static void main(String[] args) {
    BMSDisplayThread bmsThread = new BMSDisplayThread();
    CSEDisplayThread cseThread = new CSEDisplayThread();
    System.out.println("Vaibhav Dhar");
    System.out.println("1BM23CD068");
    bmsThread.start();
    cseThread.start();
  }
}
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
Vaibhav Dhar
1BM23CD068
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
```

Process finished with exit code 130

Program 9

User Interface Creation

Algorithm:

```
import javax- swing . +;
import javax aut +;
import java aut. event. ";
class Switz Demo {
    Suring Demo () [
     I Frame ifrom: new I Frame ("Divider App");
      j Frm. setsia (275, 150);
     i from. set Default Close Operation ();
     Thehel jlab: new Itahel ("Enter divider and dividend");
      I Text field ajtf: new I Text field (8);
      Textfield ajtf = new JText Field (1);
     Thutton new = new JButton (" Celculate");
      Ilabel err: new Ilabel ();
      Nobel alab = her Thehel();
      Tlabel blab = new Tlabel();
      Ilabel andlab = her Ilabel();
      ifrom. add (err);
      ifron. add (jleb);
     ifrom. add (aftf);
     ifrm. add (bjtf);
     jfron. add (aleb)
     jfrm. Badd (blob);
     ifrom. add (arsleb);
```

```
ajth. addAction listener (1) [
 bitf. add Action Usterer (1);
button. add Action listener (new Action listener C) E
      public rold actionPerformed (Action brent est) ?
           Sopla ("Action event from a fext field");
gitf. add Action listener (1);
bitf. add Action (isterns (1);
button, add Astim listener (new Action listener ()) {
  public wid action Performed (Action Event ent) {
      try { int as Integer. parse Int (a) the get Tout (1) ]
             int b = Integer parse Int (bjtf. get Text());
             ind arralbi
            alab. set Tent (" \nA = " +a);
             blab. set Tent (" In B = " +b))
            andob set Text ("In Aw: "taw);
       Catch (Numferformet Exception e) {
           alds. setText ("");
          bleb. set Text (" ")
          ander. set Toxt ("In Ahr: "tans);
     ofm. set Visible (true);
  prom (string ago(s) {
       public wid run () (
           new Suring Demo (1;
3;
```

```
Outhot-

O Num!: 10

Num: 2

Result: 5

O Num!: 10.5

Num: 2

"Plane entor valid integer"

(3) Num!: 10

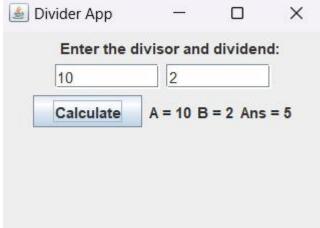
Num: 0

"Division by zero not allowed"
```

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class SwingDemo {
  SwingDemo() {
    JFrame jfrm = new JFrame("Divider App");
    ifrm.setSize(275, 200);
    jfrm.setLayout(new FlowLayout());
    jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    JLabel jlab = new JLabel("Enter the divisor and dividend:");
    JTextField ajtf = new JTextField(8);
    JTextField bitf = new JTextField(8);
    JButton button = new JButton("Calculate");
    JLabel err = new JLabel();
    JLabel alab = new JLabel();
    JLabel blab = new JLabel();
    JLabel anslab = new JLabel();
    ifrm.add(err);
    jfrm.add(jlab);
    jfrm.add(ajtf);
    jfrm.add(bjtf);
```

```
ifrm.add(button);
    jfrm.add(alab);
    jfrm.add(blab);
    jfrm.add(anslab);
     button.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);
            err.setText("");
          } catch (NumberFormatException e) {
            alab.setText("");
            blab.setText("");
            anslab.setText("");
            err.setText("Enter Only Integers!");
          } catch (ArithmeticException e) {
            alab.setText("");
            blab.setText("");
            anslab.setText("");
            err.setText("B should be NON zero!");
     });
    jfrm.setVisible(true);
  public static void main(String args[]) {
     SwingUtilities.invokeLater(new Runnable() {
       public void run() {
          new SwingDemo();
     });
} import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class SwingDemo {
  SwingDemo() {
```

```
JFrame ifrm = new JFrame("Divider App");
jfrm.setSize(275, 200);
jfrm.setLayout(new FlowLayout());
ifrm.setDefaultCloseOperation(JFrame.EXIT ON CLOSE);
JLabel ilab = new JLabel("Enter the divisor and dividend:");
JTextField aitf = 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();
ifrm.add(err);
jfrm.add(jlab);
ifrm.add(ajtf);
jfrm.add(bjtf);
ifrm.add(button);
ifrm.add(alab);
jfrm.add(blab);
jfrm.add(anslab);
button.addActionListener(new ActionListener() {
  public void actionPerformed(ActionEvent evt) {
     try {
       int a = Integer.parseInt(ajtf.getText());
       int b = Integer.parseInt(bitf.getText());
       int ans = a / b;
       alab.setText("A = " + a);
       blab.setText("B = " + b);
       anslab.setText("Ans = " + ans);
       err.setText("");
     } catch (NumberFormatException e) {
       alab.setText("");
       blab.setText("");
       anslab.setText("");
       err.setText("Enter Only Integers!");
     } catch (ArithmeticException e) {
       alab.setText("");
       blab.setText("");
```



Program 10 a)

Demonstrating IPC

Algorithm:

```
class Q {
     int ni
     boolean value set : false)
     Synchronized int get () {
      while (! value Set) {
         boy [ System. out privaler (" Consumor waiting ");
        Catch (Intoruffed Exception e) [
               System out protein (" Interrupted exception caught");
         System out printler ("Cost"+ 4);
         value set = false :
         Sopla (" Intereste Producer ");
          notify ();
         neturn h; had not a sufficient
 3
 synchronized with put (int h) {
 While (value set) [
     try {
          System out privale (" In Producer waiting In");
          wreit();
          3 catch (Interrupted Exception e) {
             System. out. printle ("Put! "+4);
             System. out. printle (" Put:"+n);
          3 hotify(1);
Class Producer implements Punneble 60 {
   Qqi
   Produce (dg) {
        this. q =q i
         new Thread (this, "Produce"), stent ();
   public wid run() {
      int := 0;
      while (i=15)
          9. put (ita)!
```

```
public wid nun () }
       int 1=0;
      while (i=15) {
            int n = q. get ();
            System out frintle ("Enrumed");
  class PCFined {
       public static cold main (String [] args) {
               Qq = new QCJ;
               new Producer (g):
                 Sopla ( Pres Control + (1);
Owtput -
Put:1
Cet:1
Pat:2
Get:2
Put: 3
Cret: 3
Put: 4
Cret: 4
R4:5
Cretis
```

```
class Q {
  int n;
  boolean valueSet = false;

synchronized int get() {
    while(!valueSet)
    try {
        System.out.println("\nConsumer waiting\n");
        wait();
    } catch(InterruptedException e) {
```

```
System.out.println("InterruptedException caught");
       }
     System.out.println("Got: " + n);
     valueSet = false;
     System.out.println("\nIntimate Producer\n");
     notify();
    return n;
  }
  synchronized void put(int n) {
     while(valueSet)
       try {
          System.out.println("\nProducer waiting\n");
          wait();
       } catch(InterruptedException e) {
         System.out.println("InterruptedException caught");
    this.n = n;
     valueSet = true;
     System.out.println("Put: " + n);
    System.out.println("\nIntimate Consumer\n");
     notify();
  }
class Producer implements Runnable {
  Qq;
  Producer(Q q) {
     this.q = q;
     new Thread(this, "Producer").start();
  public void run() {
     int i = 0;
     while(i < 15) {
       q.put(i++);
class Consumer implements Runnable {
  Qq;
  Consumer(Q q) {
     this.q = q;
     new Thread(this, "Consumer").start();
  public void run() {
    int i=0;
```

```
while(i<15) {
    int r=q.get();
    System.out.println("consumed:"+r);
    i++;
    }
} class IPC {
    public static void main(String args[]) {
        Q q = new Q();
        System.out.println("Vaibhav Dhar\n1BM23CD068");
        new Producer(q);
        new Consumer(q);
        system.out.println("Press Control-C to stop.");
    }
}</pre>
```

<pre>"C:\Program Files\Java\j</pre>	Producer waiting	Intimate Producer
Vaibhav Dhar		
1BM23CD068	Got: 2	consumed:4
Put: 0		Put: 5
	Intimate Producer	and the same
Intimate Consumer	10.1	Intimate Consumer
	consumed:2	
	Put: 3	
Producer waiting		Producer waiting
	Intimate Consumer	
Press Control-C to stop.		Got: 5
Got: 0		Intimate Producer
601. 0	Producer waiting	Intimate Producer
T 1: 1 D 1		consumed:5
Intimate Producer	Got: 3	Put: 6
	Intimate Producer	7327 3
Put: 1	Intimate Producer	Intimate Consumer
100	consumed:3	
Intimate Consumer	Put: 4	
	700. 4	Producer waiting
	Intimate Consumer	¥2
Producer waiting	THE TIME CONSOLIE!	Got: 6
consumed:0	Producer waiting	Intimate Producer
Got: 1		
	Got: 4	consumed:6
Intimate Producer		Put: 7

Intimate Consumer	Got: 9	consumed:11 Put: 12
Producer waiting	Intimate Producer	Intimate Consumer
Got: 7	consumed:9 Put: 10	
Intimate Producer	Intimate Consumer	Producer waiting
	Tire Time to Consolie	Got: 12
consumed:7 Put: 8	Producer waiting	Intimate Producer
Intimate Consumer	Got: 10	consumed:12 Put: 13
	Intimate Producer	Intimate Consumer
Producer waiting	consumed:10	The Time to Consolie
Got: 8	Put: 11	Producer waiting
Intimate Producer	Intimate Consumer	Got: 13
consumed:8 Put: 9	Producer waiting	Intimate Producer
Intimate Consumer	Got: 11	consumed:13 Put: 14
Producer waiting	Intimate Producer	Intimate Consumer

Got: 14

Intimate Producer

consumed:14

Process finished with exit code 0

Program 10 b)

Demonstrating Deadlock

Algorithm:

```
import java. util. *;
   clas A E
    synchronized wid for (0 b) {
     String home = Thread. current Thread ()- get Name ();
      Sopla ( name + "entered A. foo");
      try {
         Thread sleep (10000);
     } catch (Exception e) {
       Sopen (" A interupped ");
     Sopler (name + " trying to call P. last ()");
     bilast ();
     wid lest() {
       Sopla ("Inside A.lost");
don B {
Synchroprized boil foo (A a) [
  String hame: Thread. current Thread () get Name ():
  Soften ( name + " entered B. bur ");
 try [
     Thread. sleep (1000);
 } cotch (Eucephon e) {
    Soply (" B interrupted ");
3
```

```
Sofela ("name + "trying to call A-last());
     a. lost();
  Loid last () {
    Sopla (" Inside Alast");
 Class Deadlock implements Kunnable
  A a = new A(1;
  B'b= new B();
 Deadlock ( ) {
  Thread. current Thread () - set Name ("Main Thread");
 Thread t= new Thread (this, "Rocing Thread");
  t. sterd ();
  a. foo (b);
                           3 (a till had done his morning
  Sopla ("Back in main thread");
         " (" of gothern makes of " I what of more or with the ");
public word me run() {
 b. ber(a);
Sopla ("Back in other thread");
 public static void main (String args []) {
      her Peadlock ();
Main Thread entered A-foo
Racing Threed entered Bill box
MainThreed trying to call B.lest()
Inside A.lost
Inside A-lest
 Back in other thread
```

```
Code:
import java.util.*;
class A {
  synchronized void foo(B b) {
     String name = Thread.currentThread().getName();
     System.out.println(name + " entered == A.foo");
    try {
       Thread.sleep(1000);
     } catch (Exception e) {
       System.out.println("A Interrupted");
     System.out.println(name + " trying to call B.last");
     b.last();
  }
  synchronized void last() {
     System.out.println("Inside A.last");
  }
class B {
  synchronized void bar(A a) {
     String name = Thread.currentThread().getName();
     System.out.println(name + " entered B.bar");
     try {
       Thread.sleep(1000);
     } catch (Exception e) {
       System.out.println("B Interrupted");
     System.out.println(name + " trying to call A.last");
     a.last();
  }
  synchronized void last() {
     System.out.println("Inside B.last");
```

```
}
class Deadlock implements Runnable {
  A = new A();
  B b = new B();
  Deadlock() {
    Thread.currentThread().setName("MainThread");
    Thread t = new Thread(this, "RacingThread");
    t.start();
    a.foo(b);
    System.out.println("Back in main thread");
  public void run() {
    b.bar(a);
    System.out.println("Back in other thread");
  }
  public static void main(String args[]) {
    new Deadlock();
}
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
MainThread entered == A.foo
RacingThread entered B.bar
MainThread trying to call B.last
RacingThread trying to call A.last
```