# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

**"JnanaSangama", Belgaum -590014, Karnataka.**

**LAB REPORT**
**on**

# Object Oriented Java Programming

# (23CS3PCOOJ)

***Submitted by***

**SHRAVANI APARAJ (1BF24CS284)**

***in partial fulfillment 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 **Shravani Aparaj (1BF24CS284),** 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<br>Associate Professor<br>Department of CSE, BMSCE | Dr. Kavitha Sooda<br>Professor & HOD<br>Department of CSE, BMSCE |

# Index


| Sl. No. | Date | Experiment Title | Page No. |
|---|---|---|---|
| 1 | 23.9.25 | 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. | 5 |
| 2 | 14.10.25 | 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 | 7 |
| 3 | 28.10.25 | 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 | 10 |
| 4 | 04.11.25 | 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. | 12 |
| 5 | 4.11.25 | 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 | 14 |
| 6 | 18.11.25 | 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 | 20 |

| | | | |
|---|---|---|---|
| | | 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. | |
| 7 | 25.11.25 | 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. | 23 |
| 8 | 09.12.25 | 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. | 25 |
| 9 | | | |
| 10 | | | |

**GitHub Link:**
**https://github.com/shravani101006/Java_1BF24CS284**

**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.*;
public class quadratic
{

   public static void main(String[] args)
   {
      Scanner sc=new Scanner(System.in);
      int a,b,c;
      float r1=0;
      float r2=0;
      System.out.println("Sudhanshu Raj:1BF24CS303");
      System.out.println("Enter value of a");
      a=sc.nextInt();
      System.out.println("Enter the value of b");
      b=sc.nextInt();
      System.out.println("Enter the value of c");
      c=sc.nextInt();
      if(a==0)
      System.out.println("Not Quadratic Equation");
      else
      {
      float d=b*b-4*a*c;
      if(d==0)
      {
         r1=(-b)/(2*a);
         System.out.println("Roots are real and equal");
         System.out.println("Root 1 and Root 2: "+r1);
      }
      else if(d>0)
      {
         r1=((-b)+((float)Math.sqrt(d)))/(2*a);
         r2=((-b)-((float)Math.sqrt(d)))/(2*a);
         System.out.println("Roots are real and unequal");
         System.out.println("First root is: "+r1);
         System.out.println("Second root is: "+r2);
      }
      else
      {
         r1=(-b)/(2*a);
```

```
            r2=(float)Math.sqrt(-d)/(2*a);
            System.out.println("Roots are imaginary");
            System.out.println("The first root is: "+r1+"+"+r2+"i");
            System.out.println("The second root is: "+r1+"-"+r2+"i");
        }

        }


    }
}
```

**Output:**

```
PS C:\Users\Admin\Desktop\Shravani_1BF24CS284> cd "c:\Users\Admin\Desktop\Shravani_1BF24CS284\" ; if ($?) { javac Quadratic.java } ; if ($?) { java Quadratic }
Enter the coefficients a,b,c of the quadratic equation
1
-5
6
Roots are real and different
x1=3.0 x2=2.0
PS C:\Users\Admin\Desktop\Shravani_1BF24CS284> cd "c:\Users\Admin\Desktop\Shravani_1BF24CS284\" ; if ($?) { javac Quadratic.java } ; if ($?) { java Quadratic }
Enter the coefficients a,b,c of the quadratic equation
1
-4
4
Roots are real and same
x=2.0
PS C:\Users\Admin\Desktop\Shravani_1BF24CS284> cd "c:\Users\Admin\Desktop\Shravani_1BF24CS284\" ; if ($?) { javac Quadratic.java } ; if ($?) { java Quadratic }
Enter the coefficients a,b,c of the quadratic equation
0
2
3
Not a quadratic equation
PS C:\Users\Admin\Desktop\Shravani_1BF24CS284>
```

**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.*;
public class student
{    Scanner sc=new Scanner(System.in);
   String sub;
   int credits;
   String usn;
   int grade;
   String name;
   double sgpa;int count;
   void getStudentDetails()
   {
      System.out.println("Enter name");
      this.name=sc.nextLine();
      System.out.println("Enter usn");
      usn=sc.next();
      count++;
      getmarks();


   }
   void getmarks()
    {  int arr[]=new int[8];
      int credit[]=new int[8];
      System.out.println("Enter marks and credit for 8 subjects");
      for(int i=0;i<8;i++)
      { System.out.println("Subject "+(i+1)+":");
      System.out.println("Enter marks");
      arr[i]=sc.nextInt();
       System.out.println("Enter credit");
      credit[i]=sc.nextInt();
      }
      int grade[]=new int[8];
       for(int i=0;i<8;i++)
       {
          grade[i]=(arr[i]/10)+1;
          if(grade[i]==11)
```

```
            grade[i]=10;
            if(grade[i]<=4)
            grade[i]=0;
        }
        computesgpa(grade,credit);
    }
    void computesgpa(int[] grades,int[] credits)
    {   int score=0;int total=0;
        for(int i=0;i<8;i++)
        {
            score=score+(grades[i]*credits[i]);
            total=total+credits[i];
        }
        double ans=(double)score/(double)total;
        System.out.println("For student "+count+":");
        System.out.println("Name is:"+name);
        System.out.println("USN: "+usn);
        System.out.println("sgpa: "+ans);

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

      student student1=new student();
      student student2=new student();
      System.out.println("Calculating for student 1");
      student1.getStudentDetails();
      System.out.println("Calculating for Student 2");
      student2.getStudentDetails();


   }


}
```

## Output:

```
C:\Users\BMSCE\Desktop\Shravani_1BF24CS284> cmd /C "C:\Users\BMSCE\AppData\Roaming\Code\User\globalStorage\pleiades.java-extension-pack-jdk\java\latest\bin\java.exe --enable-preview -XX:+ShowCodeDetails
InExceptionMessages -cp C:\Users\BMSCE\AppData\Roaming\Code\User\workspaceStorage\304474769ea8fc22e2ab2846542fe421\redhat.java\jdt_ws\Shravani_1BF24CS284_751fba45\bin MainSGPA "
Enter USN:
1BF24CS284
Enter Name:
Shravani
Enter marks and credits for 3 subjects:
Subject 1 marks:
85
Subject 1 credits:
4
Subject 2 marks:
70
Subject 2 credits:
3
Subject 3 marks:
90
Subject 3 credits:
3
USN: 1BF24CS284
Name: Shravani
SGPA: 9.0

C:\Users\BMSCE\Desktop\Shravani_1BF24CS284>
```

**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 Book{
   String name,author;
   int price,pages;

   Book(String name,String author,int price,int pages){
      this.name = name;
      this.author = author;
      this.price = price;
      this.pages = pages;
   }

   public String toString(){
      String name,author,price,pages;
      name = "Book name: "+this.name+"\n";
      author = "Author name: "+this.author+"\n";
      price = "Price: "+this.price+"\n";
      pages = "Pages: "+this.pages+"\n";
      return name+author+price+pages;
   }
}

public class Lp3 {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      int n, price,pages;
      String name,author;
      System.out.println("enter no of books");
      n = sc.nextInt();
      sc.nextLine();
      Book b[] = new Book[n];
      for(int i = 0 ;i<n;i++){
         System.out.println("enter name");
         name = sc.nextLine();
         System.out.println("enter author");
         author = sc.nextLine();
         System.out.println("enter price");
         price = sc.nextInt();
```

```
            System.out.println("enter pages");
            pages = sc.nextInt();
            b[i] = new Book(name,author,price,pages);
            sc.nextLine();
        }
        for(int i = 0;i<n;i++)
            System.out.println(b[i]);
    }
}
```

## Output:

```
C:\Users\BMSCE\Desktop\Shravani_1BF24CS284> cmd /C "C:\Users\BMSCE\AppData\Roaming\Code\User\globalStorage\pleiades.java-extension-pack-jdk\java\latest\bin\java.exe --enable-preview -XX:+ShowCodeDetails
InExceptionMessages -cp C:\Users\BMSCE\AppData\Roaming\Code\User\workspaceStorage\304474769ea8fc22e2ab2846542fe421\redhat.java\jdt_ws\Shravani_1BF24CS284_751fba45\bin Main "
Enter number of books: 2

Enter details of Book 1:
Enter book name: Java
Enter author name: James
Enter price: 500
Enter number of pages: 300

Enter details of Book 2:
Enter book name: Python
Enter author name: Guido
Enter price: 450
Enter number of pages: 400

--- Book Details ---

Book 1 Details:
Book name: Java
Author name: James
Price: 500
Number of pages: 300


Book 2 Details:
Book name: Python
Author name: Guido
Price: 450
Number of pages: 400


C:\Users\BMSCE\Desktop\Shravani_1BF24CS284>
```

## Program 4:

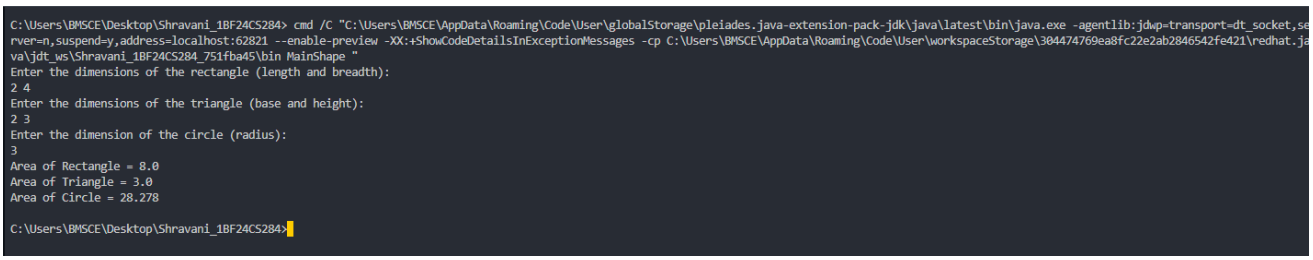
**Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea()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

## Code:

```
import java.util.Scanner;
abstract class Shape{
   int a, b;
   abstract void printArea();
}

class Rectangle extends Shape {
   Rectangle()
   {
      System.out.println("Enter the dimensions of the rectangle (length and breadth):");
      Scanner s = new Scanner(System.in);
      a = s.nextInt();
      b = s.nextInt();
   }

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

class Triangle extends Shape {
   Triangle()
   {   Scanner s = new Scanner(System.in);
      System.out.println("Enter the dimensions of the triangle (base and height):");
      a = s.nextInt();
      b = s.nextInt();
   }

   void printArea() {
      double area = 0.5 * a * b;
      System.out.println("Area of Triangle = " + area);
   }
```

```
}

class Circle extends Shape {
   Circle()
   {    Scanner s = new Scanner(System.in);
      System.out.println("Enter radius of circle:");
      a = s.nextInt();
   }

   void printArea() {
      double area = 3.142 * a * a;
      System.out.println("Area of Circle = " + area);
   }
}

public class MainShape {
   public static void main(String[] args) {
      Rectangle r =new Rectangle();
      Triangle t =new Triangle();
      Circle c =new Circle();
      r.printArea();
      t.printArea();
      c.printArea();
   }
}
```

## Output:



## 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 customerName;
   int accountNumber;
   String accountType;
   double balance;

   void getAccountDetails() {
      Scanner s = new Scanner(System.in);
      System.out.print("Enter customer name: ");
      customerName = s.next();
      System.out.print("Enter account Number: ");
      accountNumber = s.nextInt();
      System.out.print("Enter type of account (saving/current): ");
      accountType = s.next();
      balance = 0;
   }

   void display() {
      System.out.println("Customer name: " + customerName);
      System.out.println("Account number: " + accountNumber);
      System.out.println("Type of Account: " + accountType);
      System.out.println("Balance = " + balance);
   }
}
class Sav_acct extends Account {
   void deposit() {
```

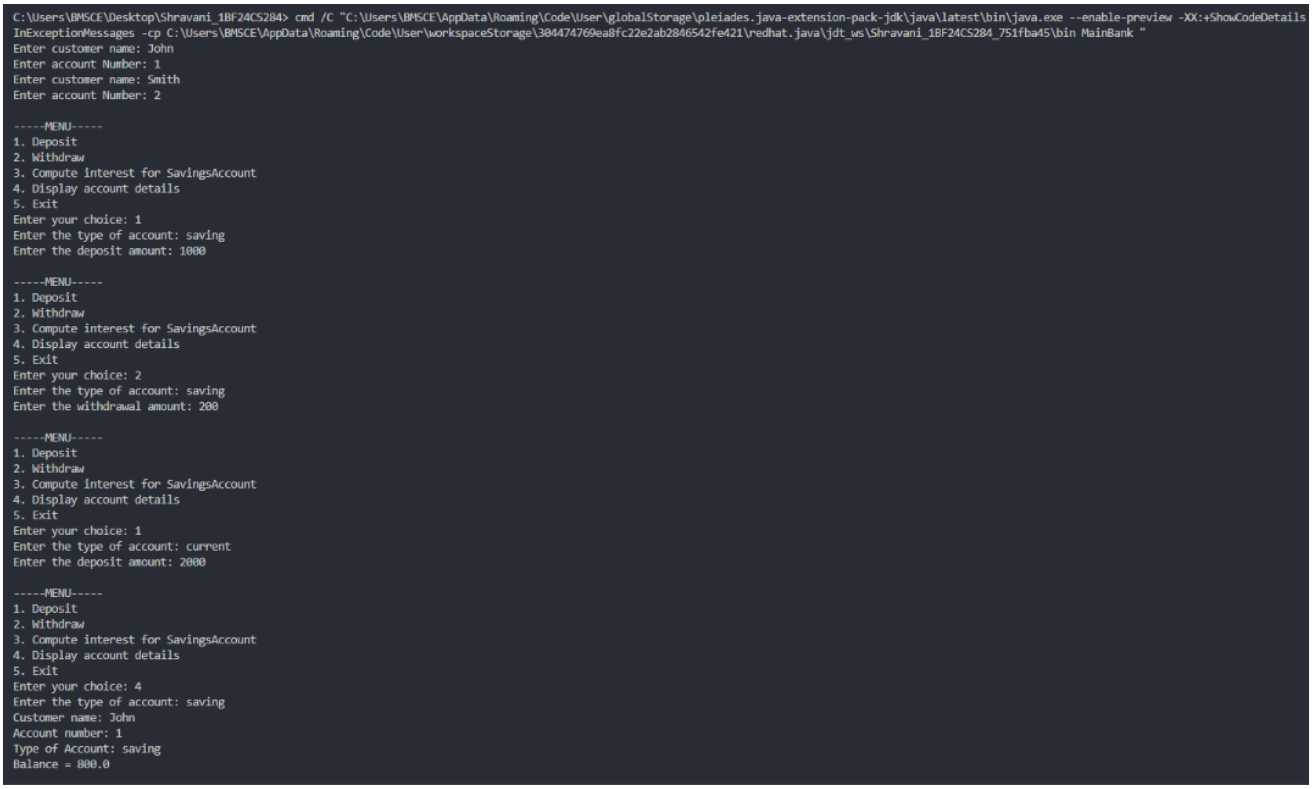
```
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the deposit amount: ");
        double amount = s.nextDouble();
        balance += amount;
    }

    void withdraw() {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the withdrawal amount: ");
        double amount = s.nextDouble();
        if (amount > balance) {
            System.out.println("Insufficient balance!");
        } else {
            balance -= amount;
        }
    }

    void computeInterest() {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the rate of interest: ");
        double rate = s.nextDouble();
        System.out.print("Enter the time period (years): ");
        int time = s.nextInt();

        double interest = balance * Math.pow((1 + rate / 100), time) - balance;
        balance += interest;
        System.out.println("Interest added = " + interest);
    }
}


class Cur_acct extends Account {
    final double minBalance = 500;
    final double serviceCharge = 100;

    void deposit() {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the deposit amount: ");
        double amount = s.nextDouble();
        balance += amount;
    }
```

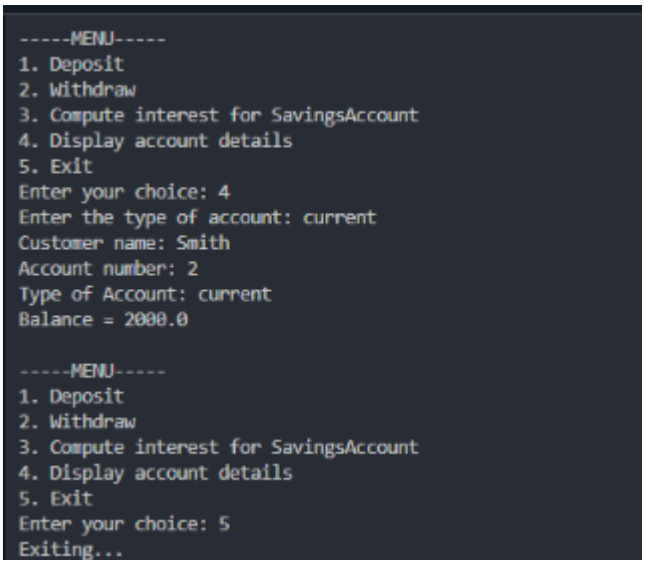
```
    void withdraw() {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the withdrawal amount: ");
        double amount = s.nextDouble();
        if (amount > balance) {
            System.out.println("Insufficient balance!");
        } else {
            balance -= amount;
            checkMinBalance();
        }
    }

    void checkMinBalance() {
        if (balance < minBalance) {
            balance -= serviceCharge;
            System.out.println("Balance below minimum! Service charge of Rs." + serviceCharge + "
imposed.");
        }
    }
}

     public class bank {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        Sav_acct sav = new Sav_acct();
        Cur_acct cur = new Cur_acct();

        System.out.print("Enter customer name for savings account");
        sav.customerName = s.next();
        System.out.print("Enter account Number: ");
        sav.accountNumber = s.nextInt();
        sav.accountType = "saving";

        System.out.print("Enter customer name for current account: ");
        cur.customerName = s.next();
        System.out.print("Enter account Number: ");
        cur.accountNumber = s.nextInt();
        cur.accountType = "current";

        int choice;
```

```
do {
   System.out.println("1. Deposit");
   System.out.println("2. Withdraw");
   System.out.println("3. Compute interest for SavingsAccount");
   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 type of account: ");
         String type = s.next();
         if (type.equalsIgnoreCase("saving"))
            sav.deposit();
         else
            cur.deposit();
         break;

      case 2:
         System.out.print("Enter the type of account: ");
         type = s.next();
         if (type.equalsIgnoreCase("saving"))
            sav.withdraw();
         else
            cur.withdraw();
         break;

      case 3:
         sav.computeInterest();
         break;

      case 4:
         System.out.print("Enter the type of account: ");
         type = s.next();
         if (type.equalsIgnoreCase("saving"))
            sav.display();
         else
            cur.display();
         break;
```

```
            case 5:
                System.out.println("Exiting");
                break;

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

## Output:




-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 4
Enter the type of account: current
Customer name: Smith
Account number: 2
Type of Account: current
Balance = 2000.0
-----MENU-----
1. Deposit
2. Withdraw
3. Compute interest for SavingsAccount
4. Display account details
5. Exit
Enter your choice: 5
Exiting...

## Program 6:

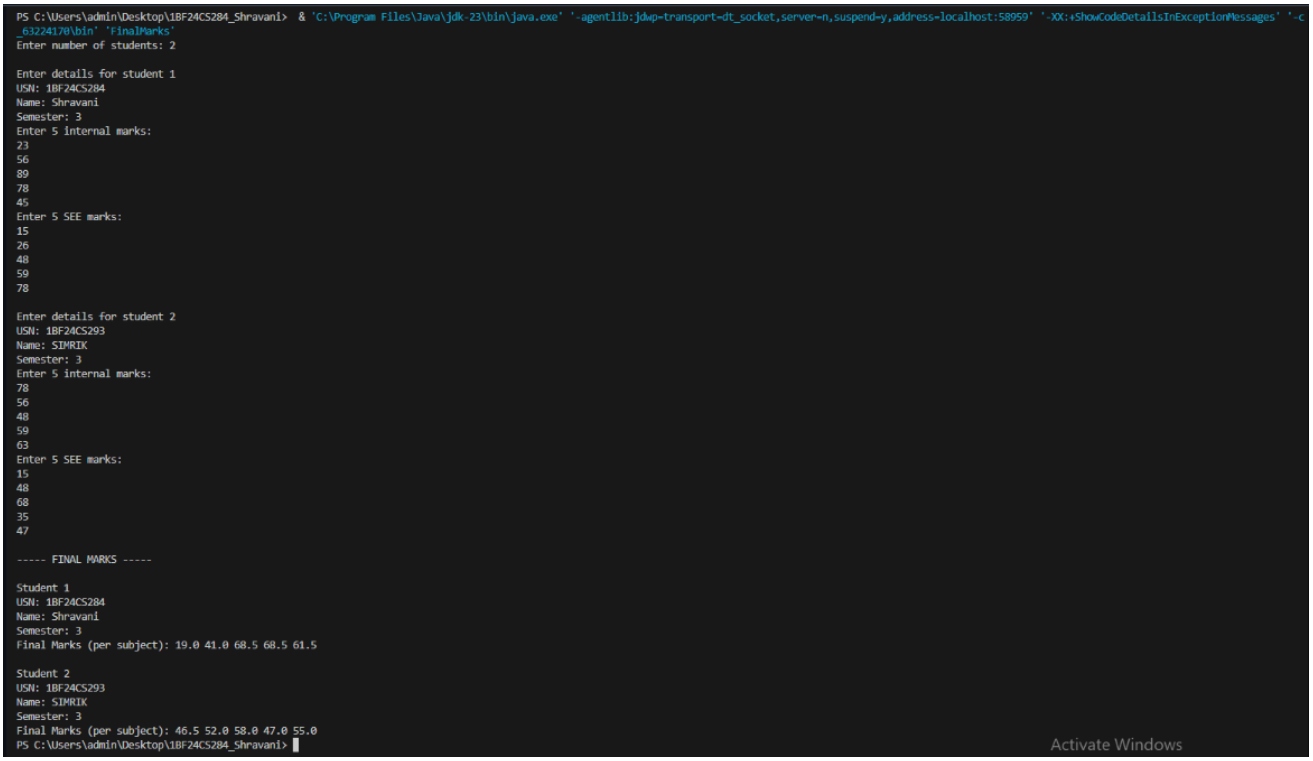
**Create a package CIE which has two classes - Student and Internals. The class Student has members like usn, name, sem. The class Internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External 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.**

**Code:**

```
package CIE;

public class Internals {
   public int[] internalMarks = new int[5];
}


package SEE;
import CIE.Personal;
public class External extends Personal {
   public int[] seeMarks = new int[5];

   public External(String usn, String name, int sem) {
      super(usn, name, sem);
   }
}

package CIE;

public class Personal {
   public String usn, name;
   public int sem;

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

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

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

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

        External[] students = new External[n];
        Internals[] internals = new Internals[n];

        for (int i = 0; i < n; i++) {
            System.out.println("\nEnter details for student " + (i + 1));
            System.out.print("USN: ");
            String usn = sc.next();
            System.out.print("Name: ");
            String name = sc.next();
            System.out.print("Semester: ");
            int sem = sc.nextInt();

            students[i] = new External(usn, name, sem);
            internals[i] = new Internals();

            System.out.println("Enter 5 internal marks:");
            for (int j = 0; j < 5; j++)
                internals[i].internalMarks[j] = sc.nextInt();

            System.out.println("Enter 5 SEE marks:");
            for (int j = 0; j < 5; j++)
                students[i].seeMarks[j] = sc.nextInt();
        }

        System.out.println("\n----- FINAL MARKS -----");
        for (int i = 0; i < n; i++) {
            System.out.println("\nStudent " + (i + 1));
            System.out.println("USN: " + students[i].usn);
            System.out.println("Name: " + students[i].name);
            System.out.println("Semester: " + students[i].sem);

            System.out.print("Final Marks (per subject): ");
            for (int j = 0; j < 5; j++) {
                double finalMark = (internals[i].internalMarks[j] / 2.0)
                                + (students[i].seeMarks[j] / 2.0);
                System.out.print(finalMark + " ");
            }
            System.out.println();
        }
        sc.close();
    }
}
```

## Output:


PS C:\Users\admin\Desktop\1BF24CS284_Shravani> & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:58959' '-XX:+ShowCodeDetailsInExceptionMessages' '-c
_6324170\bin' 'FinalMarks'
Enter number of students: 2
Enter details for student 1
USN: 1BF24CS284
Name: Shravani
Semester: 3
Enter 5 internal marks:
23
56
89
78
45
Enter 5 SEE marks:
15
26
48
59
78
Enter details for student 2
USN: 1BF24CS293
Name: SIMRIK
Semester: 3
Enter 5 internal marks:
78
56
48
59
63
Enter 5 SEE marks:
15
48
68
35
47
----- FINAL MARKS -----
Student 1
USN: 1BF24CS284
Name: Shravani
Semester: 3
Final Marks (per subject): 19.0 41.0 68.5 68.5 61.5
Student 2
USN: 1BF24CS293
Name: SIMRIK
Semester: 3
Final Marks (per subject): 46.5 52.0 58.0 47.0 55.0
PS C:\Users\admin\Desktop\1BF24CS284_Shravani>
Activate Windows

## Program 7:

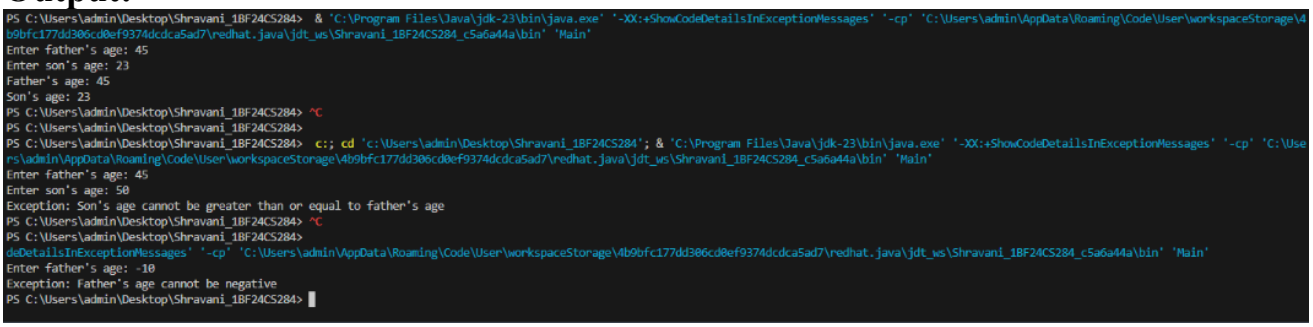
**Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age=father's age.**

**Code:**

```
import java.util.Scanner;

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

class Father {
   int fatherAge;

   Father(int age) throws WrongAge {
      if (age < 0) {
         throw new WrongAge("Father's age cannot be negative");
      }
      fatherAge = age;
   }
}

class Son extends Father {
   int sonAge;

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

      if (sonAge < 0) {
         throw new WrongAge("Son's age cannot be negative");
      }

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

      this.sonAge = sonAge;
   }
}

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

```
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter father's age: ");
        int fAge = sc.nextInt();

        System.out.print("Enter son's age: ");
        int sAge = sc.nextInt();

        try {
            Son obj = new Son(fAge, sAge);
            System.out.println("Son's age is: " + obj.sonAge);
            System.out.println("Father's age is: " + obj.fatherAge);
        } catch (WrongAge e) {
            System.out.println("Error: " + e.getMessage());
        }

        sc.close();
    }
}
```

## Output:



## 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 CollegeThread extends Thread {
   public void run() {
      try {
         for(int i=0;i<4;i++) {
            System.out.println("BMS College of Engineering");
            Thread.sleep(10000); // 10 seconds
         }
      } catch (InterruptedException e) {
         System.out.println("CollegeThread interrupted");
      }
   }
}

class CSEThread extends Thread {
   public void run() {
      try {
          for(int i=0;i<4;i++){
            System.out.println("CSE");
            Thread.sleep(2000); // 2 seconds
         }
      } catch (InterruptedException e) {
         System.out.println("CSEThread interrupted");
      }
   }
}

public class threaddemo {
   public static void main(String[] args) {
      CollegeThread t1 = new CollegeThread();
      CSEThread t2 = new CSEThread();

      t1.start();
      t2.start();
   }
}
```

# Output:

```
PS C:\Users\admin\Desktop\Shravani_1BF24CS284>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\admin\AppData\Roaming\Code\User\workspaceStorage\4b9bfc177dd306cd0f9374dcdca5ad7\redhat.java\jdt_ws\Shravani_1BF24CS284_c5a6a44a\bin' 'ThreadDemo'
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
PS C:\Users\admin\Desktop\Shravani_1BF24CS284>
```