VISVESVARAYA TECHNOLOGICAL UNIVERSITY
"Jnana Sangama", Belgaum -590014, Karnataka.

PROJECT WORK REPORT
On

"OOJ LAB REPORT"

Submitted by

SUDARSHAN KOMAR (1BM22CS291)

Under the Guidance of

SHRAVYA A R
ASSISTANT PROFESSOR

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
Mar 2024

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 project work entitled **"OOJ LAB REPORT"** carried out by **SUDARSHAN KOMAR (1BM22CS291)** who are bonafide students 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 Visveswaraiah Technological University, Belgaum during the year 2024. The project report has been approved as it satisfies the academic requirements in respect of **OOJ LAB REPORT** work prescribed for the said degree.

Signature of the Guide

Shravya A R
Assistant Professor
BMSCE, Bengaluru

Signature of the HOD

Dr. Jyothy S Nayak
Prof & Head of Dept of CSE
BMSCE, Bengaluru

External Viva

Name of the Examiner

Signature with date

B.M.S. COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I, **SUDARSHAN KOMAR (1BM22CS291)** of 3rd Semester, B.E, Department of Computer Science and Engineering, BMS College of Engineering, Bangalore, hereby declare that, this PROJECT entitled **"OOJ LAB REPORT"** has been carried out by me under the guidance of Shravya ma'am, Assistant Professor, Department of CSE, BMS College of Engineering, Bangalore during the academic semester Dec 2023 - Mar 2024.

We also declare that to the best of our knowledge and belief, the development reported here is not from part of any other report by any other students.

Signature

SUDARSHAN KOMAR (1BM22CS291)

TABLE OF CONTENTS

5	Week 5 – **Program 6:** Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of 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.	19-21
6	Week 6 – **Program 7:** Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >=father's age. **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.	22-26
7	Week 7 – **Program 9:** Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.	27-30

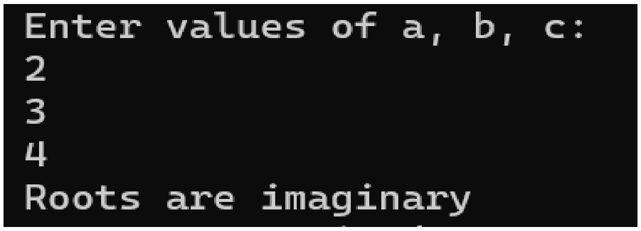
WEEK 1

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 discriminant b2-4ac is negative, display a message stating that there are no real solutions.

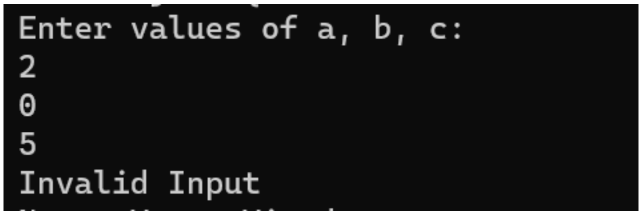
CODE:

```
import java.util.*;
public class QuadEq
{
    public static void main(String args[])
    {
        float a, b, c, d=0.0f, r1=0.0f, r2=0.0f;
        System.out.println("Enter values of a, b, c: ");
        Scanner read= new Scanner(System.in);
        a=read.nextFloat();
        b=read.nextFloat();
        c=read.nextFloat();
        if(a==0||b==0||c==0)
        {
            System.out.println("Invalid Input");
        }
        else
        {
            d=b*b-4*a*c;
            if(d>0)
            {
                r1=(float)(-b+Math.sqrt(d))/(2*a);
                r2=(float)(-b-Math.sqrt(d))/(2*a);
                System.out.println("Roots are real and distinct\nR1= "+r1+"\tR2= "+r2);
            }
            else if(d<0)
            {
                System.out.println("Roots are imaginary");
            }
            else
            {
                r1=-b/(2*a);
                r2=r1;
                System.out.println("Roots are real and equal\nR1= "+r1+"\tR2= "+r2);
            }
        }
    }
}
```

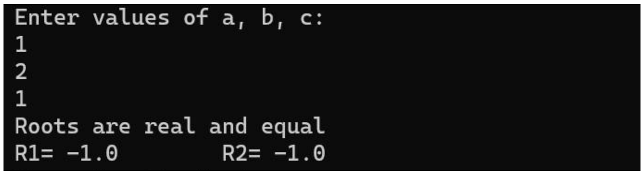
OUTPUT:



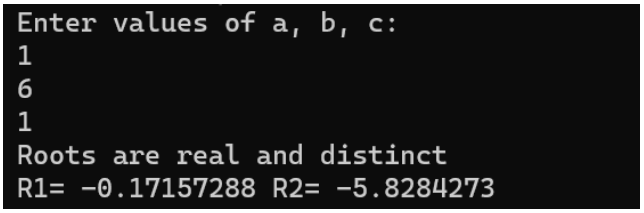
```
Enter values of a, b, c:
2
3
4
Roots are imaginary
```



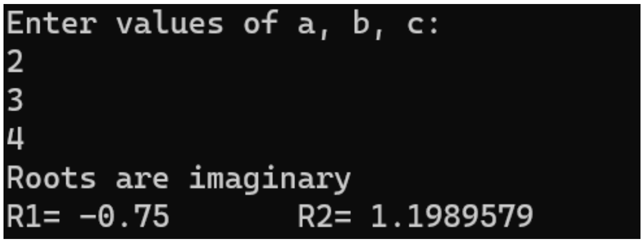
```
Enter values of a, b, c:
2
0
5
Invalid Input
```



```
Enter values of a, b, c:
1
2
1
Roots are real and equal
R1= -1.0        R2= -1.0
```



```
Enter values of a, b, c:
1
6
1
Roots are real and distinct
R1= -0.17157288 R2= -5.8284273
```



```
Enter values of a, b, c:
2
3
4
Roots are imaginary
R1= -0.75       R2= 1.1989579
```

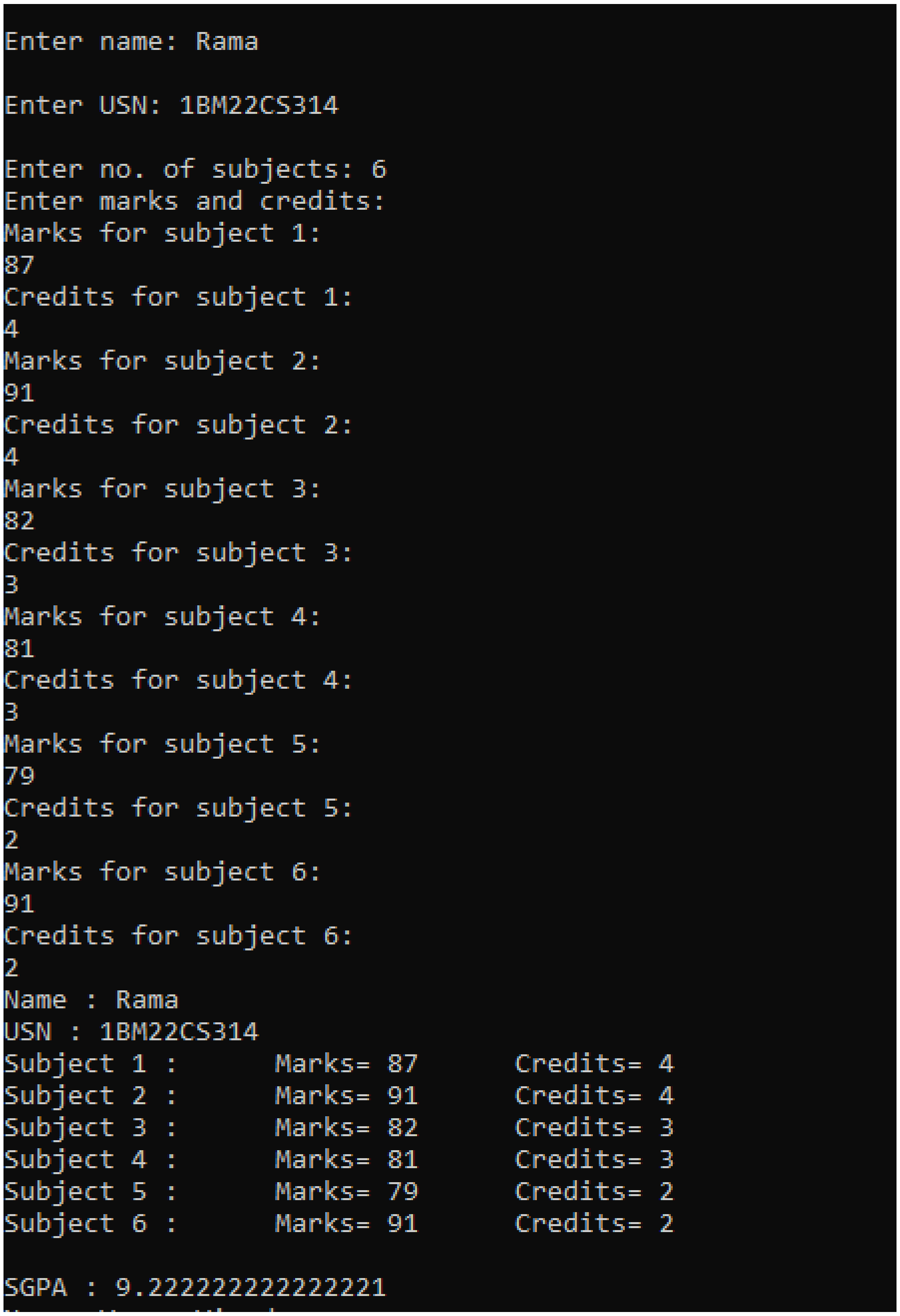
WEEK 2

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 S222GPA of a student.

CODE:

```
import java.util.*;
public class Student {
    String name, usn;
    int credits[], marks[];
    public void display(double res) {
        System.out.println("Name : " + name);
        System.out.println("USN : " + usn);
        for (int i = 0; i < credits.length; i++) {
            System.out.println("Subject " + (i + 1) + " :\t Marks= " + marks[i] + "\tCredits= " +
credits[i]);
        }
        System.out.println("\nSGPA : " + res);
    }
    public double sgpa() {
        double tc = 0;
        double tgp = 0;
        for (int i = 0; i < credits.length; i++) {
            tc += credits[i];
            tgp += calgp(marks[i]) * credits[i];
        }
        return (tgp / tc);
    }
    public double calgp(int m) {
        if (m >= 90) {
            return 10;
        } else if (m >= 80) {
            return 9;
        } else if (m >= 70) {
            return 8;
        } else if (m >= 60) {
            return 7;
        } else if (m >= 50) {
            return 6;
        } else {
            return 0;
        }
    }
    public static void main(String args[]) {
        Scanner read = new Scanner(System.in);
        Student ob = new Student();

        System.out.print("\nEnter name: ");
        ob.name = read.nextLine();

        System.out.print("\nEnter USN: ");
        ob.usn = read.next();

        System.out.print("\nEnter no. of subjects: ");
        int n = read.nextInt();
        ob.credits = new int[n];
        ob.marks = new int[n];

        System.out.println("Enter marks and credits:");
        for (int i = 0; i < n; i++) {
            System.out.println("Marks for subject " + (i + 1) + ": ");
            ob.marks[i] = read.nextInt();
            System.out.println("Credits for subject " + (i + 1) + ": ");
            ob.credits[i] = read.nextInt();
        }

        double res = ob.sgpa();
        ob.display(res);
        }
}
```

OUTPUT:



```
Enter name: Rama

Enter USN: 1BM22CS314

Enter no. of subjects: 6
Enter marks and credits:
Marks for subject 1:
87
Credits for subject 1:
4
Marks for subject 2:
91
Credits for subject 2:
4
Marks for subject 3:
82
Credits for subject 3:
3
Marks for subject 4:
81
Credits for subject 4:
3
Marks for subject 5:
79
Credits for subject 5:
2
Marks for subject 6:
91
Credits for subject 6:
2
Name : Rama
USN : 1BM22CS314
Subject 1 :      Marks= 87      Credits= 4
Subject 2 :      Marks= 91      Credits= 4
Subject 3 :      Marks= 82      Credits= 3
Subject 4 :      Marks= 81      Credits= 3
Subject 5 :      Marks= 79      Credits= 2
Subject 6 :      Marks= 91      Credits= 2

SGPA : 9.222222222222221
```

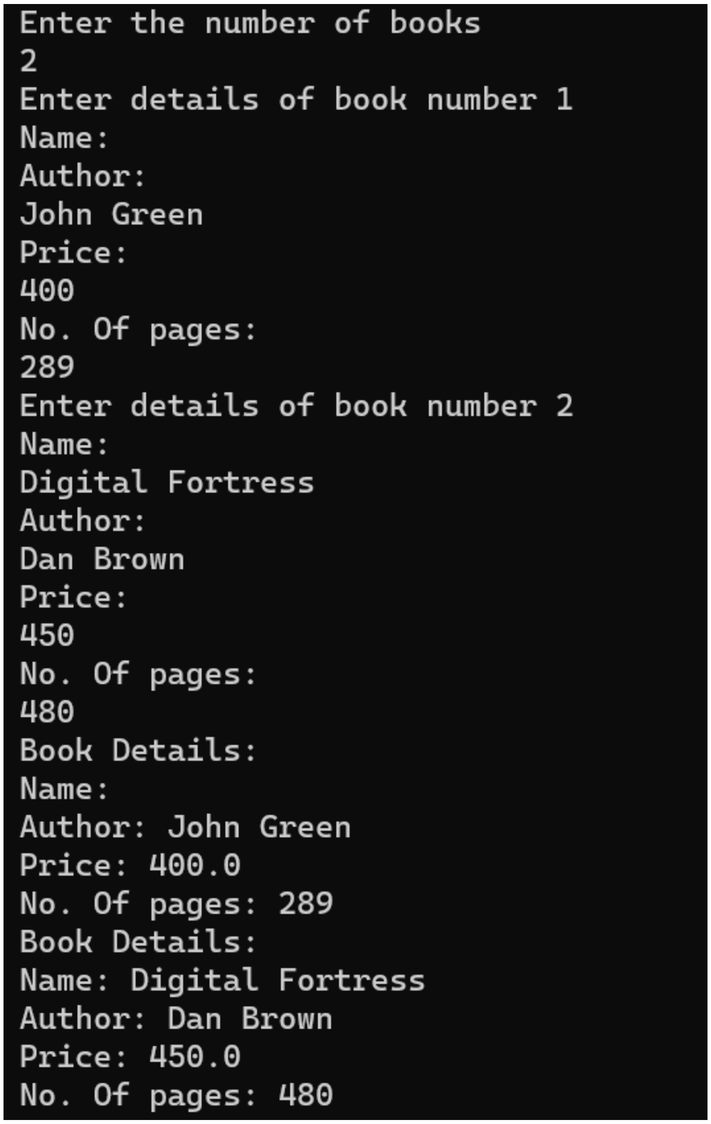
WEEK 3

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.*;
class Book {
    Scanner input = new Scanner(System.in);
    private String name;
    private String author;
    private double price;
    private int pages;
    Book(String name, String author, double price, int pages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.pages = pages;
    }
    void setName(String name) {
        this.name = name;
    }
    String getName() {
        return name;
    }
    void setAuthor(String author) {
        this.author = author;
    }
    String getAuthor() {
        return author;
    }
    void setPrice(double price) {
        this.price = price;
    }
    double getPrice() {
        return price;
    }
    void setPages(int pages) {
        this.pages = pages;
    }
    int getPages() {
        return pages;
    }
    public String toString() {
        return ("Book Details: \nName: " + name + "\nAuthor: " + author + "\nPrice: " + price +
"\nNo. Of pages: " + pages);
    }
}
class BookTest {
    public static void main(String args[]) {
        System.out.println("Enter the number of books");
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        Book book[] = new Book[n];
        for (int i = 0; i < n; i++) {
            System.out.println("Enter details of book number " + (i + 1) + "\nName: ");
            String na = sc.nextLine();
            System.out.println("Author: ");
            String a = sc.nextLine();
            System.out.println("Price: ");
            double p = sc.nextDouble();
            System.out.println("No. Of pages: ");
            int num = sc.nextInt();
            book[i] = new Book(na, a, p, num);
            sc.nextLine(); // consume newline character
        }

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

OUTPUT:



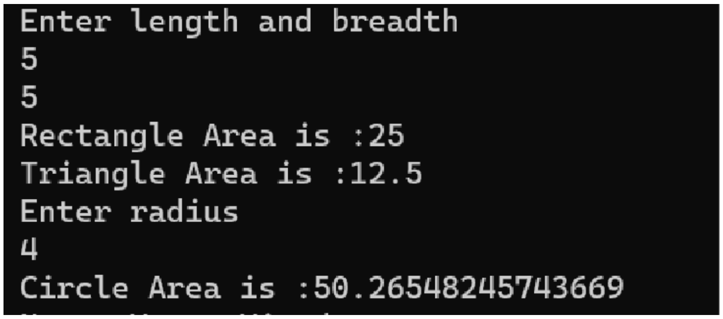
```
Enter the number of books
2
Enter details of book number 1
Name:
Author:
John Green
Price:
400
No. Of pages:
289
Enter details of book number 2
Name:
Digital Fortress
Author:
Dan Brown
Price:
450
No. Of pages:
480
Book Details:
Name:
Author: John Green
Price: 400.0
No. Of pages: 289
Book Details:
Name: Digital Fortress
Author: Dan Brown
Price: 450.0
No. Of pages: 480
```

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.

CODE:

```
import java.util.*;
abstract class Shape
{
   int x,y;
   void printArea()
   {
   }
}
class Rectangle extends Shape
{
   Rectangle(int l,int b)
   {
      x=l;
      y=b;
   }
   void printArea()
   {
       int area=x*y;
       System.out.println("Rectangle Area is :" + area);
   }
}
class Triangle extends Shape
{
   Triangle(int l,int h)
   {
      x=l;
      y=h;
   }
   void printArea()
   {
      double area = 0.5*x*y;
      System.out.println("Triangle Area is :" +area);
   }
}
class Circle extends Shape
{
  Circle(int r)
  {
    x=r;
  }
  void printArea()
  {
    double area= Math.PI*x*x;
    System.out.println("Circle Area is :" +area);
  }
}
class ShapeTest
{
  public static void main(String args[])
  {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter length and breadth");
        int a= s.nextInt();
        int b= s.nextInt();
     Shape ob=new Rectangle(a,b);
     ob.printArea();
     Shape ob1=new Triangle(a,b);
     ob1.printArea();
     System.out.println("Enter radius");
     int r=s.nextInt();
     Shape ob2=new Circle(r);
     ob2.printArea();
  }
}
```

OUTPUT:



```
Enter length and breadth
5
5
Rectangle Area is :25
Triangle Area is :12.5
Enter radius
4
Circle Area is :50.26548245743669
```

WEEK 4

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:

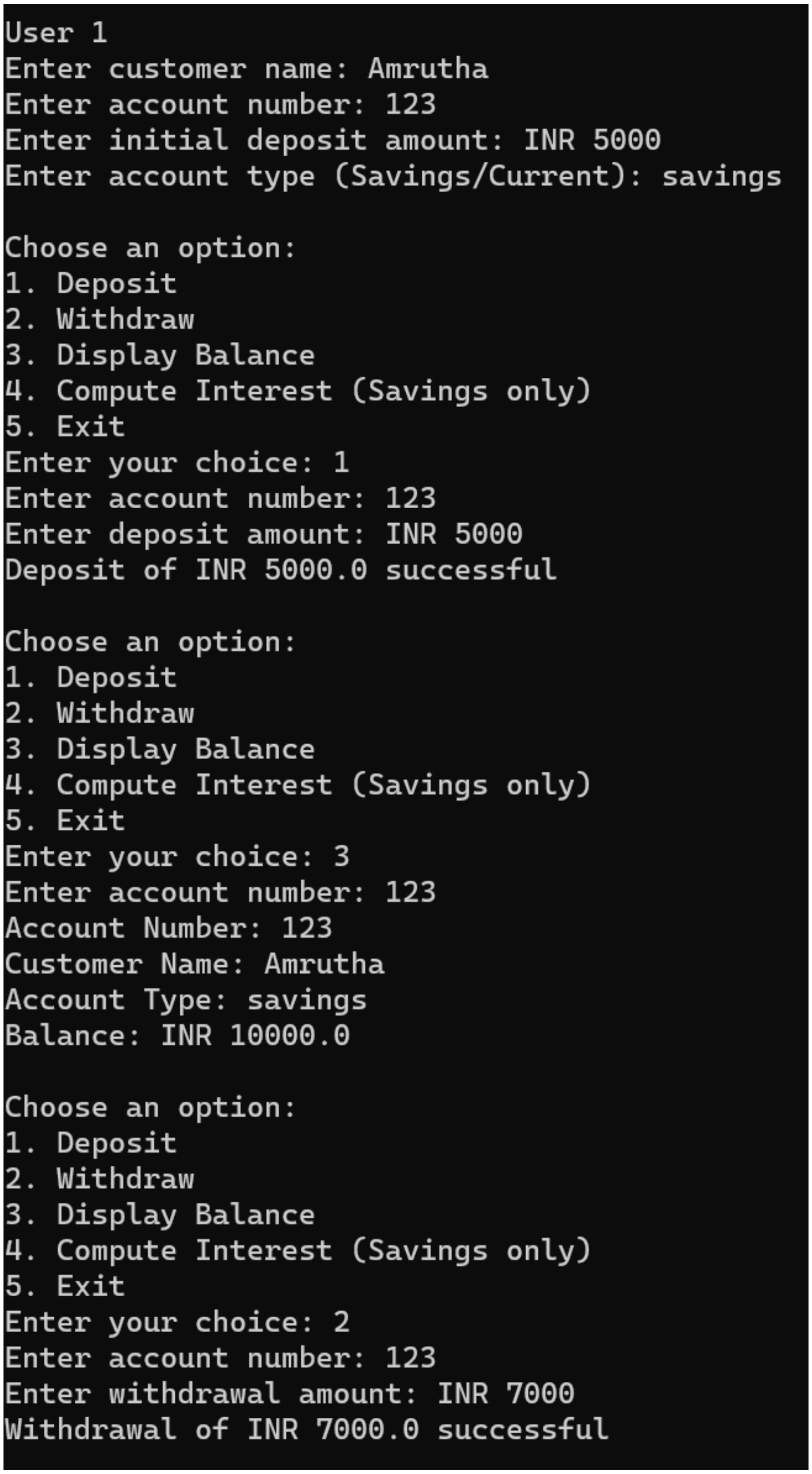
a) Accept deposit from customer and update the balance.
b) Display the balance.
c) Compute and deposit interest
d) Permit withdrawal and update the balance

Check for the minimum balance, impose penalty if necessary and update the balance.

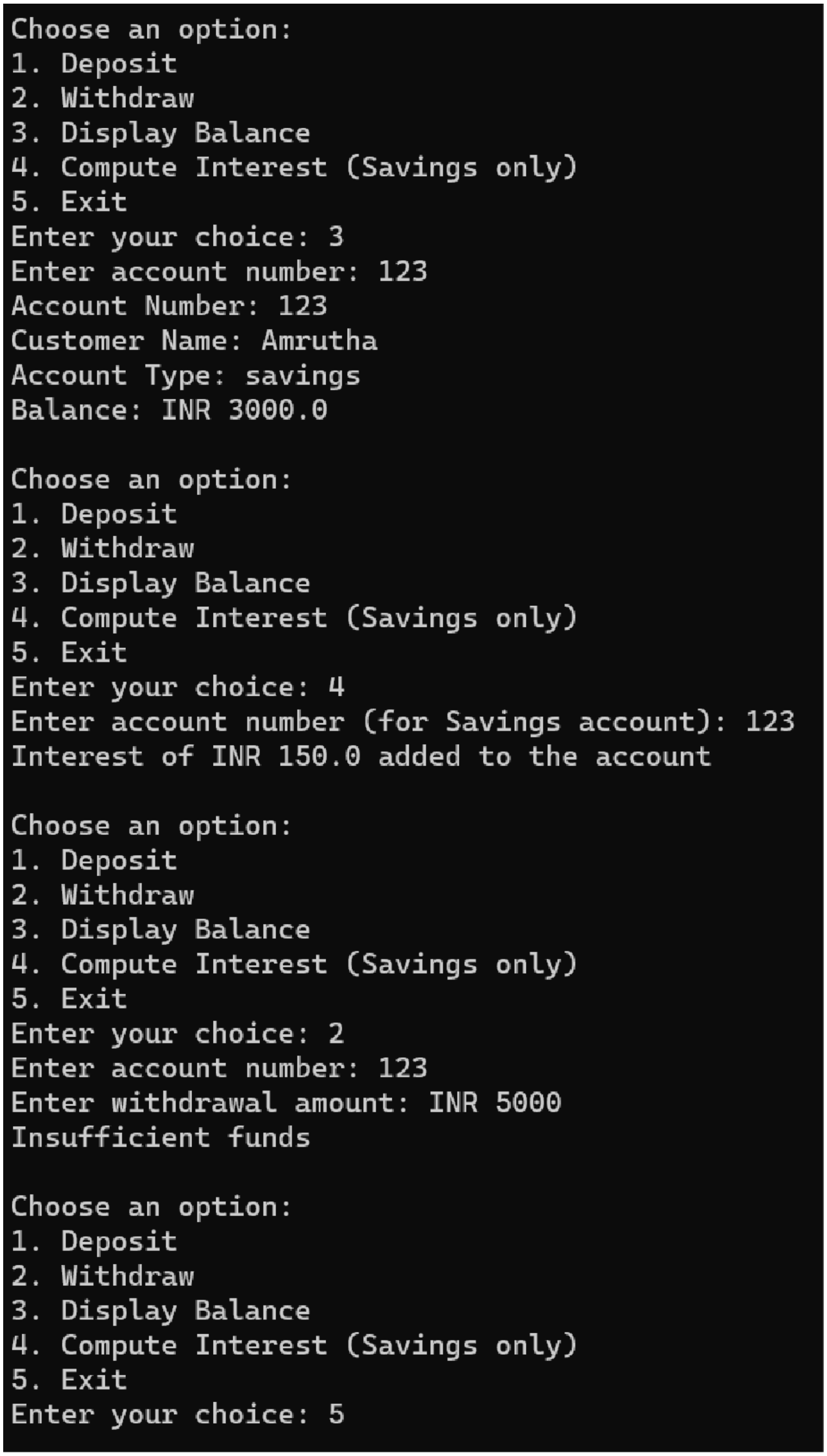
CODE:

```
import java.util.Scanner;

class Account {
    String customerName;
    int accountNumber;
    String accountType;
    double balance;

    Account(String name, int number, String type, double initialBalance) {
        customerName = name;
        accountNumber = number;
 accountType = type;
        balance = initialBalance;
    }

    void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            System.out.println("Deposit of INR " + amount + " successful");
        } else {
            System.out.println("Invalid deposit amount. Please enter a positive value.");
        }
    }

    void displayBalance() {
        System.out.println("Account Number: " + accountNumber);
        System.out.println("Customer Name: " + customerName);
        System.out.println("Account Type: " + accountType);
        System.out.println("Balance: INR " + balance);
    }

    void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println("Withdrawal of INR " + amount + " successful");
        } else {
            System.out.println("Insufficient funds");
        }
    }
void computeInterest() {
 }
    void checkMinimumBalance(double minBalance, double serviceCharge) {
    }
}

class SavAcct extends Account {
    double interestRate = 0.05;

    SavAcct(String name, int number, String type, double initialBalance) {
        super(name, number, type, initialBalance);
    }

    void computeInterest() {
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("Interest of INR " + interest + " added to the account");
    }
}

class CurAcct extends Account {
    double minBalance = 1000;
    double serviceCharge = 50;

    CurAcct(String name, int number, String type, double initialBalance) {
        super(name, number, type, initialBalance);
    }

   void checkMinimumBalance(double minBalance, double serviceCharge) {
        if (balance < minBalance) {
            System.out.println("Service charge of INR " + serviceCharge + " imposed");
            balance -= serviceCharge;
        }
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of users: ");
        int numUsers = scanner.nextInt();

        Account[] accounts = new Account[numUsers];
        for (int i = 0; i < numUsers; i++) {
            System.out.println("\nUser " + (i + 1));
            System.out.print("Enter customer name: ");
            scanner.nextLine();
            String name = scanner.nextLine();
            System.out.print("Enter account number: ");
            int accNumber = scanner.nextInt();
            System.out.print("Enter initial deposit amount: INR ");
            double initialDeposit = scanner.nextDouble();
            System.out.print("Enter account type (Savings/Current): ");
            scanner.nextLine();
    String accType = scanner.nextLine();
            if (accType.equalsIgnoreCase("Savings")) {
                accounts[i] = new SavAcct(name, accNumber, accType, initialDeposit);
            } else if (accType.equalsIgnoreCase("Current")) {
                accounts[i] = new CurAcct(name, accNumber, accType, initialDeposit);
            } else {
                System.out.println("Invalid account type entered. Defaulting to Account.");
                accounts[i] = new Account(name, accNumber, "Account", initialDeposit);
            }
        }

        boolean exit = false;
        while (!exit) {
            System.out.println("\nChoose an option:");
            System.out.println("1. Deposit");
            System.out.println("2. Withdraw");
            System.out.println("3. Display Balance");
            System.out.println("4. Compute Interest (Savings only)");
            System.out.println("5. Exit");
            System.out.print("Enter your choice: ");
            while (!scanner.hasNextInt()) {
                System.out.println("Invalid input. Please enter a number.");
                scanner.next();
            }
            int choice = scanner.nextInt();
            switch (choice) {
                case 1:
                    System.out.print("Enter account number: ");
int accNum = scanner.nextInt();
                    System.out.print("Enter deposit amount: INR ");
                    double depositAmount = scanner.nextDouble();
                    for (Account acc : accounts) {
                        if (acc.accountNumber == accNum) {
                            acc.deposit(depositAmount);
                        }
                    }
                    break;
                case 2:
                    System.out.print("Enter account number: ");
                    accNum = scanner.nextInt();
                    System.out.print("Enter withdrawal amount: INR ");
                    double withdrawAmount = scanner.nextDouble();
            for (Account acc : accounts) {
                if (acc.accountNumber == accNum) {
                    acc.withdraw(withdrawAmount);
                }
            }
            break;
        case 3:
            System.out.print("Enter account number: ");
            accNum = scanner.nextInt();
            for (Account acc : accounts) {
                if (acc.accountNumber == accNum) {
                    acc.displayBalance();
                }
            }
break;
        case 4:
            System.out.print("Enter account number (for Savings account): ");
            accNum = scanner.nextInt();
            for (Account acc : accounts) {
                if (acc.accountNumber == accNum && acc instanceof SavAcct) {
                    ((SavAcct) acc).computeInterest();
                }
            }
            break;
        case 5:
            exit = true;
            break;
        default:
            System.out.println("Invalid choice. Please enter a valid option.");
      }
    }
  }
}
```

OUTPUT:



```
User 1
Enter customer name: Amrutha
Enter account number: 123
Enter initial deposit amount: INR 5000
Enter account type (Savings/Current): savings

Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 1
Enter account number: 123
Enter deposit amount: INR 5000
Deposit of INR 5000.0 successful

Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 3
Enter account number: 123
Account Number: 123
Customer Name: Amrutha
Account Type: savings
Balance: INR 10000.0

Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 2
Enter account number: 123
Enter withdrawal amount: INR 7000
Withdrawal of INR 7000.0 successful
```



```
Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 3
Enter account number: 123
Account Number: 123
Customer Name: Amrutha
Account Type: savings
Balance: INR 3000.0

Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 4
Enter account number (for Savings account): 123
Interest of INR 150.0 added to the account

Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 2
Enter account number: 123
Enter withdrawal amount: INR 5000
Insufficient funds

Choose an option:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit
Enter your choice: 5
```

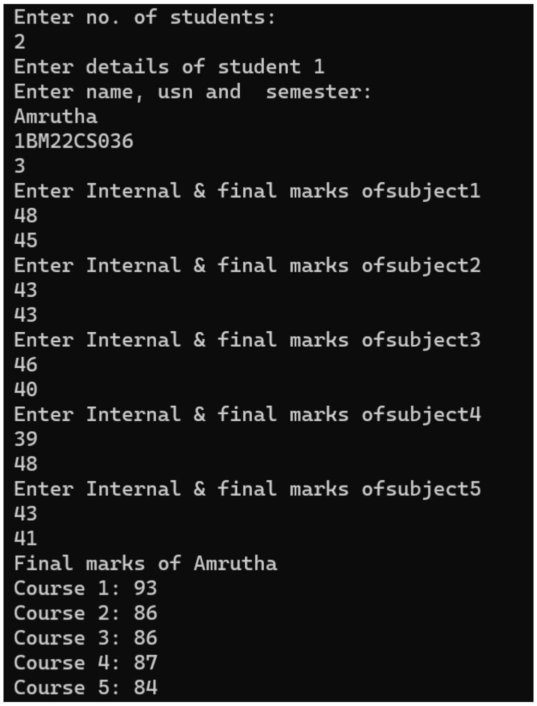
WEEK 5

Program 6: Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of 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.

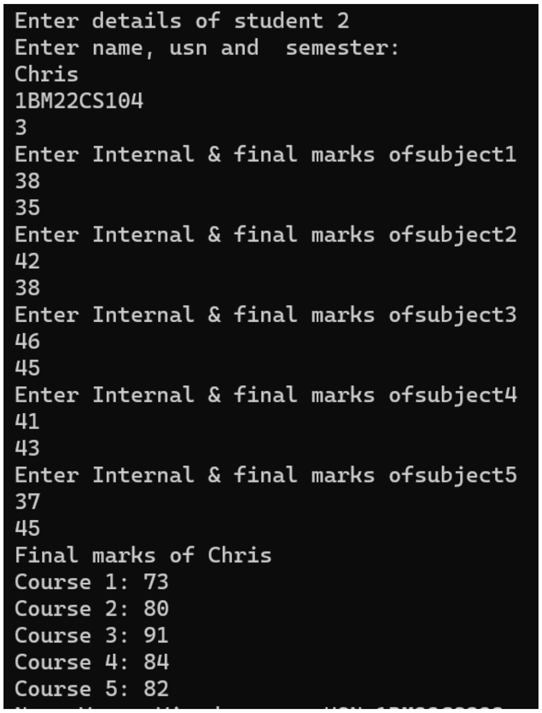
CODE:

```
package CIE;
import java.util.*;
public class Student{
        public String name;
        public String usn;
        public int sem;
        public void accept()
        {
                Scanner input=new Scanner(System.in);
                System.out.println("Enter name, usn and  semester: ");
                name= input.nextLine();
                usn= input.nextLine();
                sem= input.nextInt();
        }
}
package CIE;
public class Internals
{
        public int int_marks[]=new int[5];
}

package SEE;
import CIE.Student;
public class External extends Student
{
        public int ext_marks[]=new int[5];
}
import java.util.*;
import CIE.*;
import SEE.*;
public class FinalMarks
{
    public static void main(String args[])
    {
        int final_marks[]=new int[5];
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter no. of students: ");
        int n=sc.nextInt();
        CIE.Internals obj[]=new CIE.Internals[n];
        SEE.External obj1[]=new SEE.External[n];
        for(int i=0;i<n;i++)
        {
            obj[i]=new CIE.Internals();
            obj1[i]=new SEE.External();
            System.out.println("Enter details of student "+(i+1));
            obj1[i].accept();
            for(int j=0;j<5;j++)
            {
                System.out.println("Enter Internal & final marks ofsubject"+(j+1));
                obj[i].int_marks[j]=sc.nextInt();
                obj1[i].ext_marks[j]=sc.nextInt();
                final_marks[j]=obj[i].int_marks[j]+ obj1[i].ext_marks[j];
            }
            System.out.println("Final marks of "+obj1[i].name);
            for(int k=0;k<5;k++)
            {
                System.out.println("Course "+(k+1)+": "+final_marks[k]);
            }
        }
    }

}
```

OUTPUT:



```
Enter no. of students:
2
Enter details of student 1
Enter name, usn and  semester:
Amrutha
1BM22CS036
3
Enter Internal & final marks ofsubject1
48
45
Enter Internal & final marks ofsubject2
43
43
Enter Internal & final marks ofsubject3
46
40
Enter Internal & final marks ofsubject4
39
48
Enter Internal & final marks ofsubject5
43
41
Final marks of Amrutha
Course 1: 93
Course 2: 86
Course 3: 86
Course 4: 87
Course 5: 84
```



```
Enter details of student 2
Enter name, usn and  semester:
Chris
1BM22CS104
3
Enter Internal & final marks ofsubject1
38
35
Enter Internal & final marks ofsubject2
42
38
Enter Internal & final marks ofsubject3
46
45
Enter Internal & final marks ofsubject4
41
43
Enter Internal & final marks ofsubject5
37
45
Final marks of Chris
Course 1: 73
Course 2: 80
Course 3: 91
Course 4: 84
Course 5: 82
```

WEEK 6

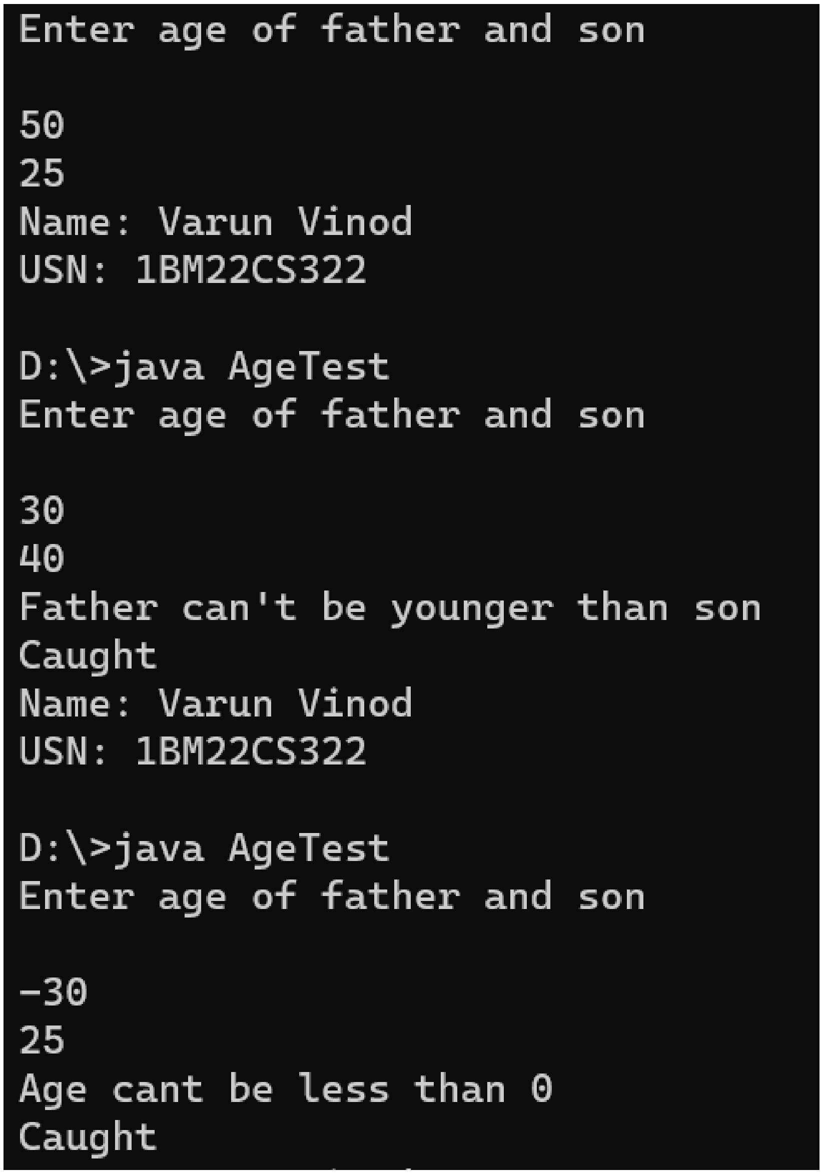
Program 7: Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >=father's age.

CODE:

```
import java.util.*;
class WrongAge extends Exception
{
        String message;
        public WrongAge(String msg)
        {
                this.message=msg;
                System.out.println(msg);
        }
}
class Father
{
        int f_age;
        Father(int f_age) throws WrongAge
        {
                if(f_age<0)
                {
                        throw new WrongAge("Age cant be less than 0");
                }
                this.f_age=f_age;
        }
}
class Son extends Father
{
        int s_age;
        Son(int f_age,int s_age) throws WrongAge
        {
                super(f_age);
                if(f_age<=s_age)
                {
                throw new WrongAge("Father can't be younger than son");
                }
                this.s_age=s_age;
```

```
    }
}
class AgeTest
{
    public static void main(String args[])
    {
        int f,s;
        Scanner input=new Scanner(System.in);
        System.out.println("Enter age of father and son\n");
        f=input.nextInt();
        s=input.nextInt();
        try{
            Father ob1=new Father(f);
            Son ob2=new Son(f,s);
        }
        catch(WrongAge e)
        {
            System.out.println("Caught");
        }
    }
}
```

OUTPUT:



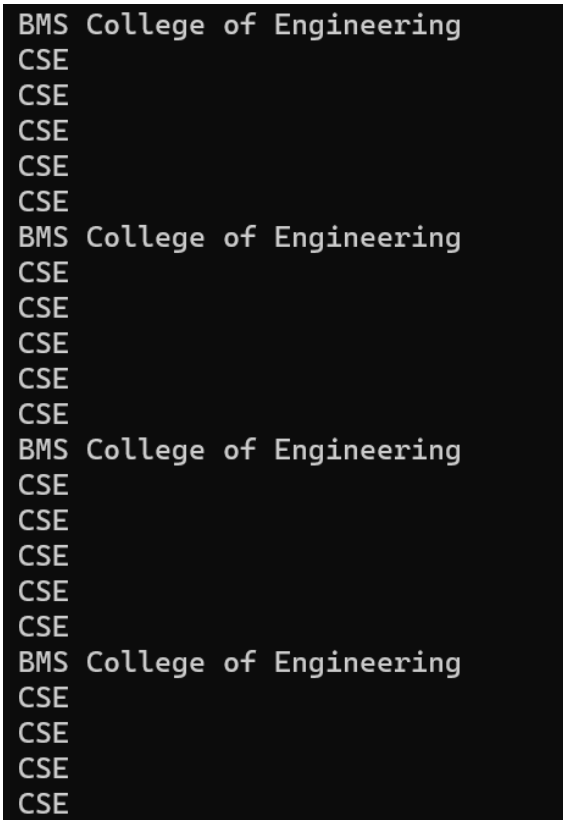
```
Enter age of father and son

50
25
Name: Varun Vinod
USN: 1BM22CS322

D:\>java AgeTest
Enter age of father and son

30
40
Father can't be younger than son
Caught
Name: Varun Vinod
USN: 1BM22CS322

D:\>java AgeTest
Enter age of father and son

-30
25
Age cant be less than 0
Caught
```

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 DisplayThread extends Thread {

    private String message;

    private int intervalMillis;

    public DisplayThread(String message, int intervalMillis) {

        this.message = message;

        this.intervalMillis = intervalMillis;

    }

    public void run() {

        while (true) {

            try {

                System.out.println(message);

                Thread.sleep(intervalMillis);

            } catch (InterruptedException e) {

                e.printStackTrace();

            }

        }

    }

}

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

        DisplayThread thread1 = new DisplayThread("BMS College of Engineering", 10000); // 10
seconds

        DisplayThread thread2 = new DisplayThread("CSE", 2000); // 2 seconds

        thread1.start();

        thread2.start();

    }

}
```

OUTPUT:



```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
```

WEEK 7

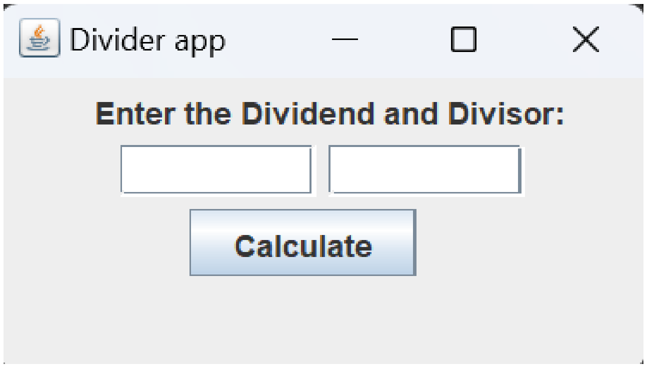
Program 9: Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

CODE:

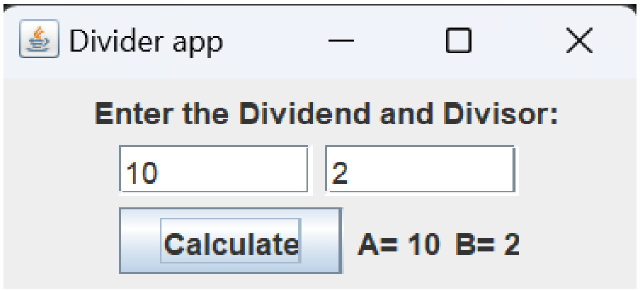
```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class SwingDemo
{
   SwingDemo()
   {
      JFrame jfrm= new JFrame("Divider app");
      jfrm.setSize(265,150);
      jfrm.setLayout(new FlowLayout());
      jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      JLabel jlab=new JLabel("Enter the divider and divident: ");
      JTextField ajtf=new JTextField(8);
      JTextField bjtf=new JTextField(8);

      JButton button = new JButton("Calculate");

      JLabel err=new JLabel();
      JLabel alab=new JLabel();
      JLabel blab= new JLabel();
      JLabel anslab=new JLabel();

      //add in order
      jfrm.add(err);//to display error
      jfrm.add(jlab);
      jfrm.add(ajtf);
      jfrm.add(bjtf);
      jfrm.add(button);
      jfrm.add(alab);
jfrm.add(blab);
jfrm.add(anslab);

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

button.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent evt)
    {
        if (err.getText()!=null)
            err.setText("");
    try
    {
        int a =Integer.parseInt(ajtf.getText());
        int b =Integer.parseInt(bjtf.getText());
        int ans= a/b;

        alab.setText("\nA= "+a);
        blab.setText("\nB= "+b);
        anslab.setText("\nAns= "+ans);
    }
    catch(NumberFormatException e)
    {
        alab.setText("");
        blab.setText("");
        err.setText("Enter only Integers!");
    }
    catch(ArithmeticException e)
    {
        alab.setText("");
        blab.setText("");
```

```
                err.setText("B should be NON zero!");
            }
        }
        });

        //display frame
        jfrm.setVisible(true);
    }

    public static void main(String args[])
    {
        //create frame on event dispatching thread
        SwingUtilities.invokeLater(new Runnable()
        {
            public void run()
            {
                new SwingDemo();
            }
        });
    }
}
```

OUTPUT:



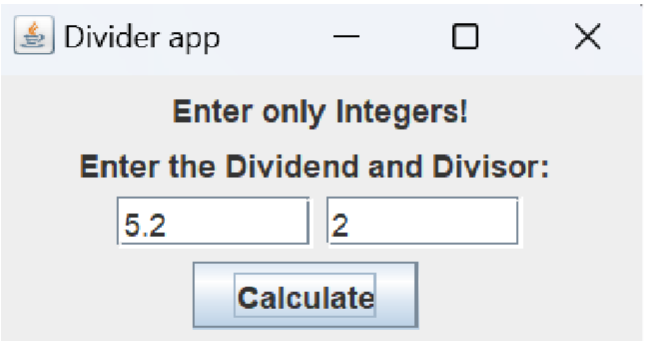
Enter the Dividend and Divisor:



Enter the Dividend and Divisor:

10 2

A= 10 B= 2



Enter only Integers!

Enter the Dividend and Divisor:

5.2 2