B.M.S. College of Engineering

(Autonomous Institution affiliated to VTU, Belagavi)

Department of Computer Science and Engineering

LAB

OBJECT ORIENTED JAVA PROGRAMMING REPORT

23CS3PCOOJ

(December 2023-March 2024)

B.M.S. College of Engineering
Department of Computer Science and Engineering

Laboratory Certificate

This is to certify that BONALA NIKHITH has satisfactorily completed the course of Experiments in Practical OBJECT-ORIENTED JAVA PROGRAMMING prescribed by the Department during the odd semester 2023-24.

Name of the Candidate: BONALA NIKHITH

USN No.: **1BM22CS072** Semester: **III** Section: **B**

Marks	
Max. Marks	Obtained
10	
Marks in Words	

Signature of the staff in-charge **Head of the Department**

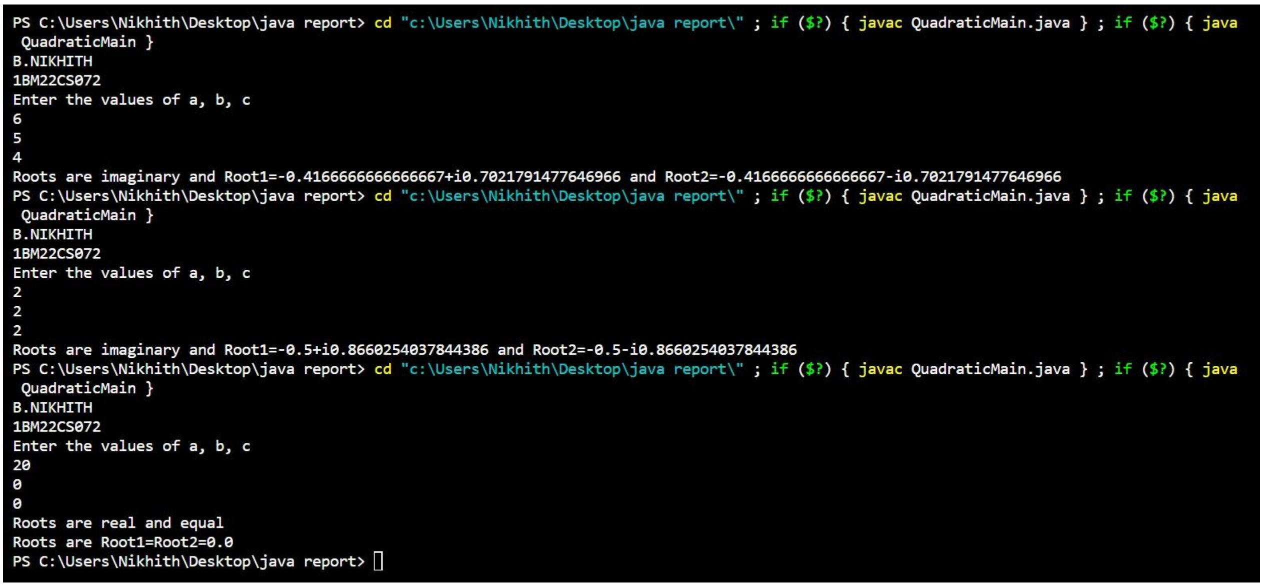
Date:

1.WRITE TO PROGRAM TO FIND QUADRATIC EQUATION OF GIVEN ROOTS.

Ans:

```
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 values 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 of a");
            Scanner s = new Scanner(System.in);
            a = s.nextInt();
        }

        d = (b * b) - (4 * a * c);

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

class QuadraticMain {
    public static void main(String[] args) {
        System.out.println("B.NIKHITH");
        System.out.println("1BM22CS072");
        quadratic q = new quadratic();
        q.getd();
        q.compute();
    }
}
```

Output:



```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac QuadraticMain.java } ; if ($?) { java
 QuadraticMain }
B.NIKHITH
1BM22CS072
Enter the values of a, b, c
6
5
4
Roots are imaginary and Root1=-0.4166666666666667+i0.7021791477646966 and Root2=-0.4166666666666667-i0.7021791477646966
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac QuadraticMain.java } ; if ($?) { java
 QuadraticMain }
B.NIKHITH
1BM22CS072
Enter the values of a, b, c
2
2
2
Roots are imaginary and Root1=-0.5+i0.8660254037844386 and Root2=-0.5-i0.8660254037844386
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac QuadraticMain.java } ; if ($?) { java
 QuadraticMain }
B.NIKHITH
1BM22CS072
Enter the values of a, b, c
20
0
0
Roots are real and equal
Roots are Root1=Root2=0.0
PS C:\Users\Nikhith\Desktop\java report> 
```

2.Create a class Book that contains four members: name, author, price, and 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.

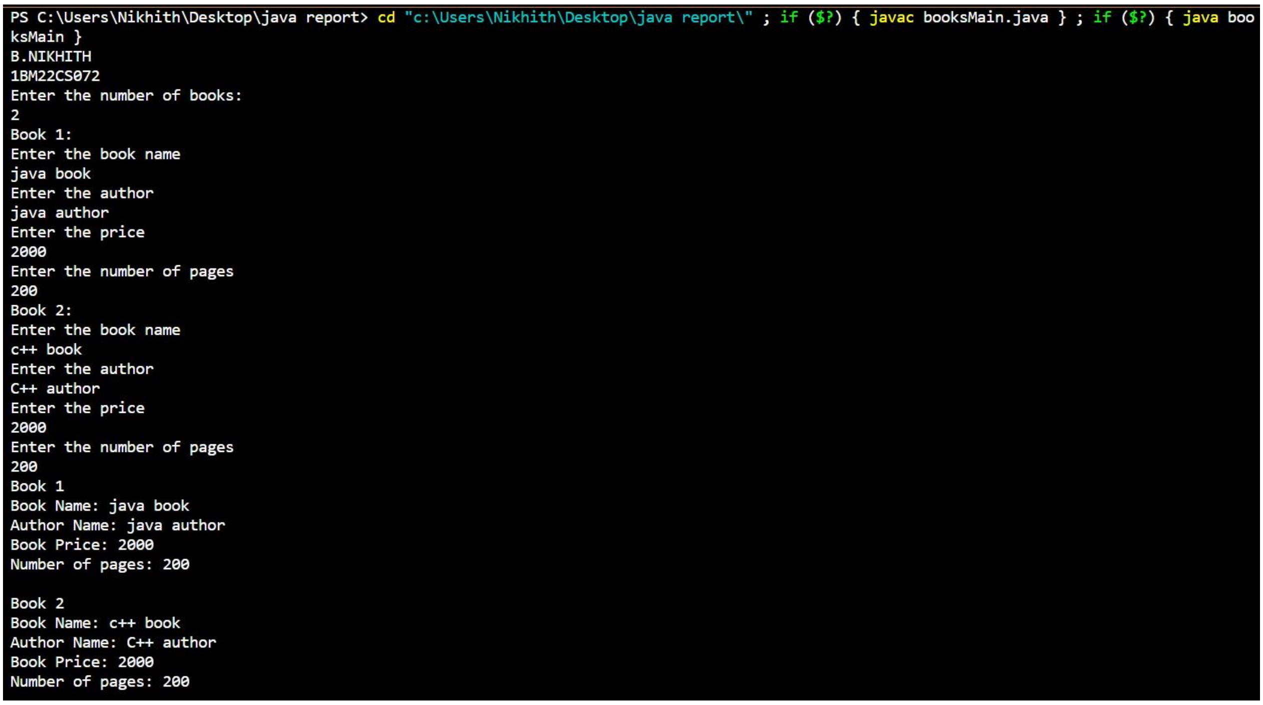
Ans:

```
import java.util.Scanner;

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

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

    public String toString() {
        return "Book Name: " + this.name + "\n" +
                "Author Name: " + this.author + "\n" +
                "Book Price: " + this.price + "\n" +
                "Number of pages: " + this.numPages + "\n";
    }
}

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

```
    System.out.println("B.NIKHITH");
    System.out.println("1BM22CS072");
    Scanner s = new Scanner(System.in);
    int n;
    String name;
    String author;
    int price;
    int numPages;

    System.out.println("Enter the number of books:");
    n = s.nextInt();

    books[] b;
    b = new books[n];

    for (int i = 0; i < n; i++) {
        System.out.println("Book " + (i + 1) + ":");
        System.out.println("Enter the book name");
        s.nextLine();
        name = s.nextLine();
        System.out.println("Enter the author");
        author = s.nextLine();
        System.out.println("Enter the price");
        price = s.nextInt();
        System.out.println("Enter the number of pages");
        numPages = s.nextInt();

        b[i] = new books(name, author, price, numPages);
    }

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

```
}
```

Output:



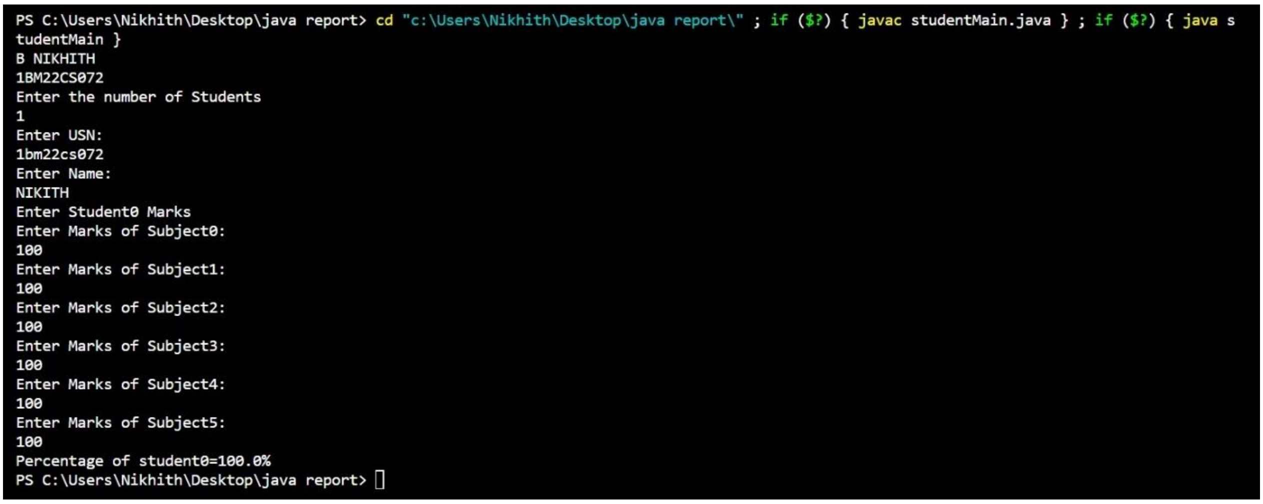
```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac booksMain.java } ; if ($?) { java boo
ksMain }
B.NIKHITH
1BM22CS072
Enter the number of books:
2
Book 1:
Enter the book name
java book
Enter the author
java author
Enter the price
2000
Enter the number of pages
200
Book 2:
Enter the book name
c++ book
Enter the author
C++ author
Enter the price
2000
Enter the number of pages
200
Book 1
Book Name: java book
Author Name: java author
Book Price: 2000
Number of pages: 200

Book 2
Book Name: c++ book
Author Name: C++ author
Book Price: 2000
Number of pages: 200
```

3.Write a Java program to create a class Student with members USN, name, marks(6 subjects). Include methods to accept student details and marks, Also include a method to calculate the percentage and display appropriate details. (Array of student object to be created).

Ans:

```
import java.util.Scanner;

class  student  {
    String  USN;
    String name;
    int marks[] = new int[6];
    float percentage = 0;

    void getd(int i) {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter USN: ");
        USN = s.next();
        System.out.println("Enter Name:");
        name = s.next();
        System.out.println("Enter Student" + i + " Marks");
        for (int j = 0; j < 6; j++) {
            System.out.println("Enter Marks of Subject" + j + ":");
            marks[j] = s.nextInt();
            percentage += marks[j];
        }
    }

    void calculatePercentage(int i) {
        percentage = (percentage / 6);
        System.out.println("Percentage of student" + i + "=" +
percentage + "%");
    }
}

class studentMain {
    public static void main(String[] args) {
    System.out.println("B NIKHITH");
        System.out.println("1BM22CS072");
        System.out.println("Enter the number of Students");
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        student s[] = new student[n];
        for (int i = 0; i < n; i++) {
            s[i] = new student();
            s[i].getd(i);
        }
        for (int i = 0; i < n; i++) {
            s[i].calculatePercentage(i);
        }
    }
}
```

<u>Output:</u>



```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac studentMain.java } ; if ($?) { java s
tudentMain }
B NIKHITH
1BM22CS072
Enter the number of Students
1
Enter USN:
1bm22cs072
Enter Name:
NIKITH
Enter Student0 Marks
Enter Marks of Subject0:
100
Enter Marks of Subject1:
100
Enter Marks of Subject2:
100
Enter Marks of Subject3:
100
Enter Marks of Subject4:
100
Enter Marks of Subject5:
100
Percentage of student0=100.0%
PS C:\Users\Nikhith\Desktop\java report> 
```

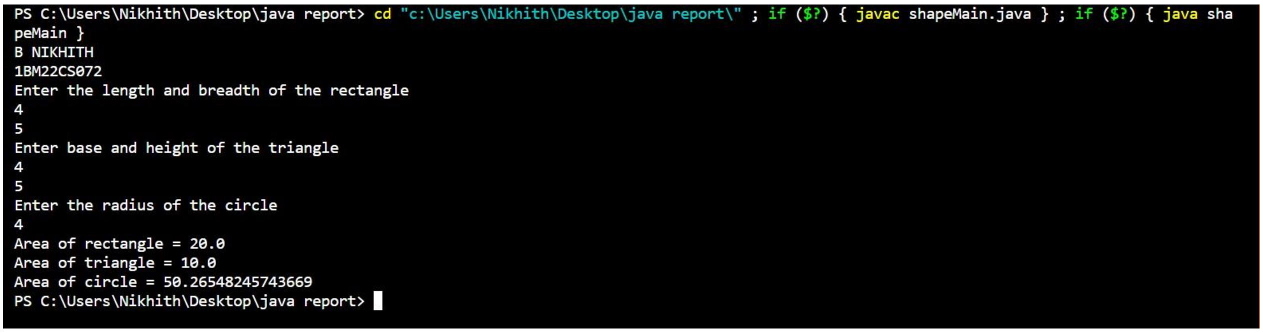
4.Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain the method printArea(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

Ans:

```
import java.util.Scanner;

abstract class shape {
    int dim1;
    int dim2;

    shape(int dim1, int dim2) {
        this.dim1 = dim1;
        this.dim2 = dim2;
    }

    abstract void printArea();
}

class rectangle extends shape {
    rectangle(int length, int breadth) {
        super(length, breadth);
    }

    void printArea() {
        double area = dim1 * dim2;
        System.out.println("Area of rectangle = " + area);
    }
}
```

```
class triangle extends shape {
    triangle(int height, int base) {
        super(height, base);
    }

    void printArea() {
        double area = 0.5 * dim1 * dim2;
        System.out.println("Area of triangle = " + area);
    }
}

class circle extends shape {
    circle(int radius) {
        super(radius, 0);
    }

    void printArea() {
        double area = Math.PI * dim1 * dim1;
        System.out.println("Area of circle = " + area);
    }
}

public class AbstractMain {
    public static void main(String[] args) {
        System.out.println("B NIKHITH");
        System.out.println("1BM22CS072");
        Scanner s = new Scanner(System.in);

        System.out.println("Enter the length and breadth of the
rectangle");
        int l = s.nextInt();
        int b = s.nextInt();
        System.out.println("Enter base and height of the triangle");
        int ba = s.nextInt();
        int h = s.nextInt();

        System.out.println("Enter the radius of the circle");
        int r = s.nextInt();

        rectangle re = new rectangle(l, b);
        triangle t = new triangle(h, ba);
        circle c = new circle(r);

        re.printArea();
        t.printArea();
        c.printArea();
    }
}
```

Output:



```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac shapeMain.java } ; if ($?) { java sha
peMain }
B NIKHITH
1BM22CS072
Enter the length and breadth of the rectangle
4
5
Enter base and height of the triangle
4
5
Enter the radius of the circle
4
Area of rectangle = 20.0
Area of triangle = 10.0
Area of circle = 50.26548245743669
PS C:\Users\Nikhith\Desktop\java report> █
```

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:

- **Accept deposit from customer and update the balance.**
- **Display the balance.**
- **Compute and deposit interest**
- **Permit withdrawal and update the balance**

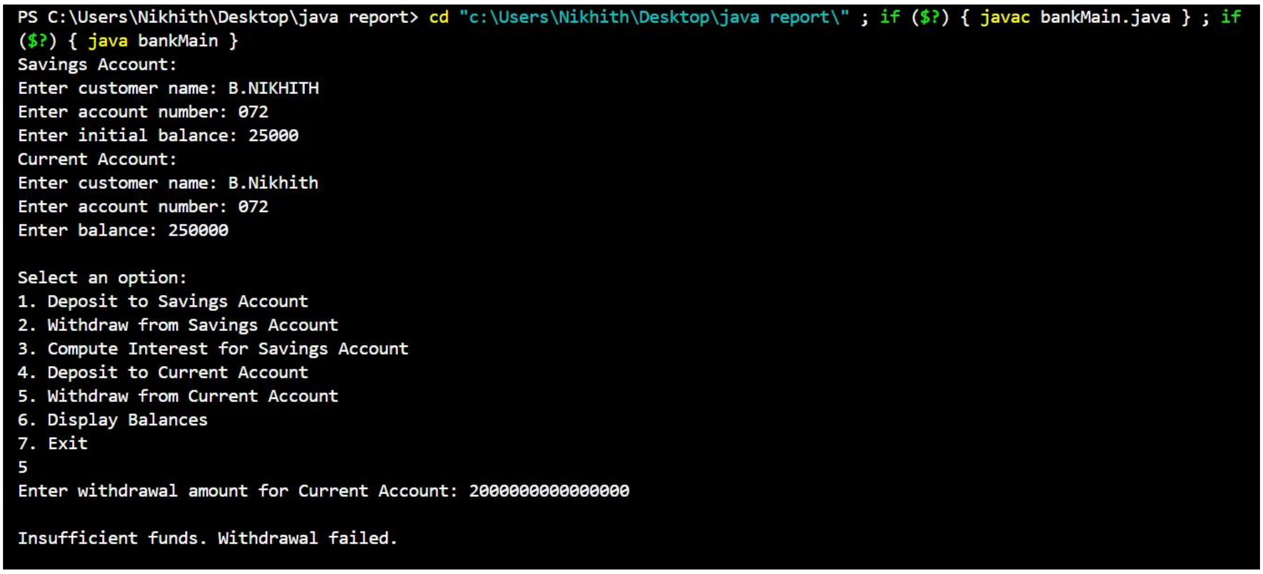
Check for the minimum balance, impose penalty if necessary and update the balance.

Ans:

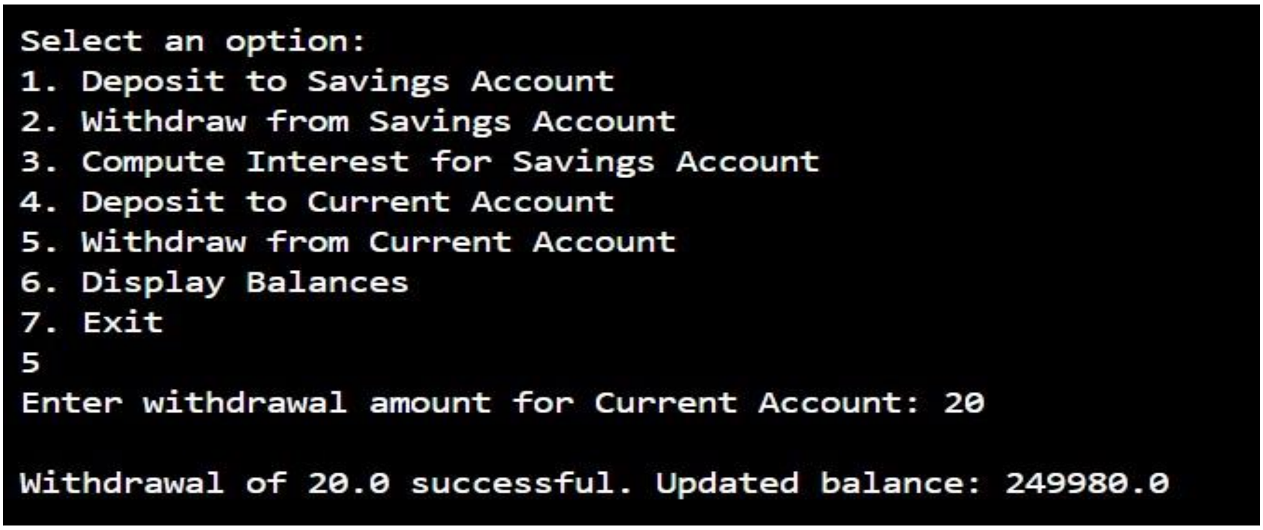
```
import java.util.Scanner;

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.");
                }
            }

        }
}
```

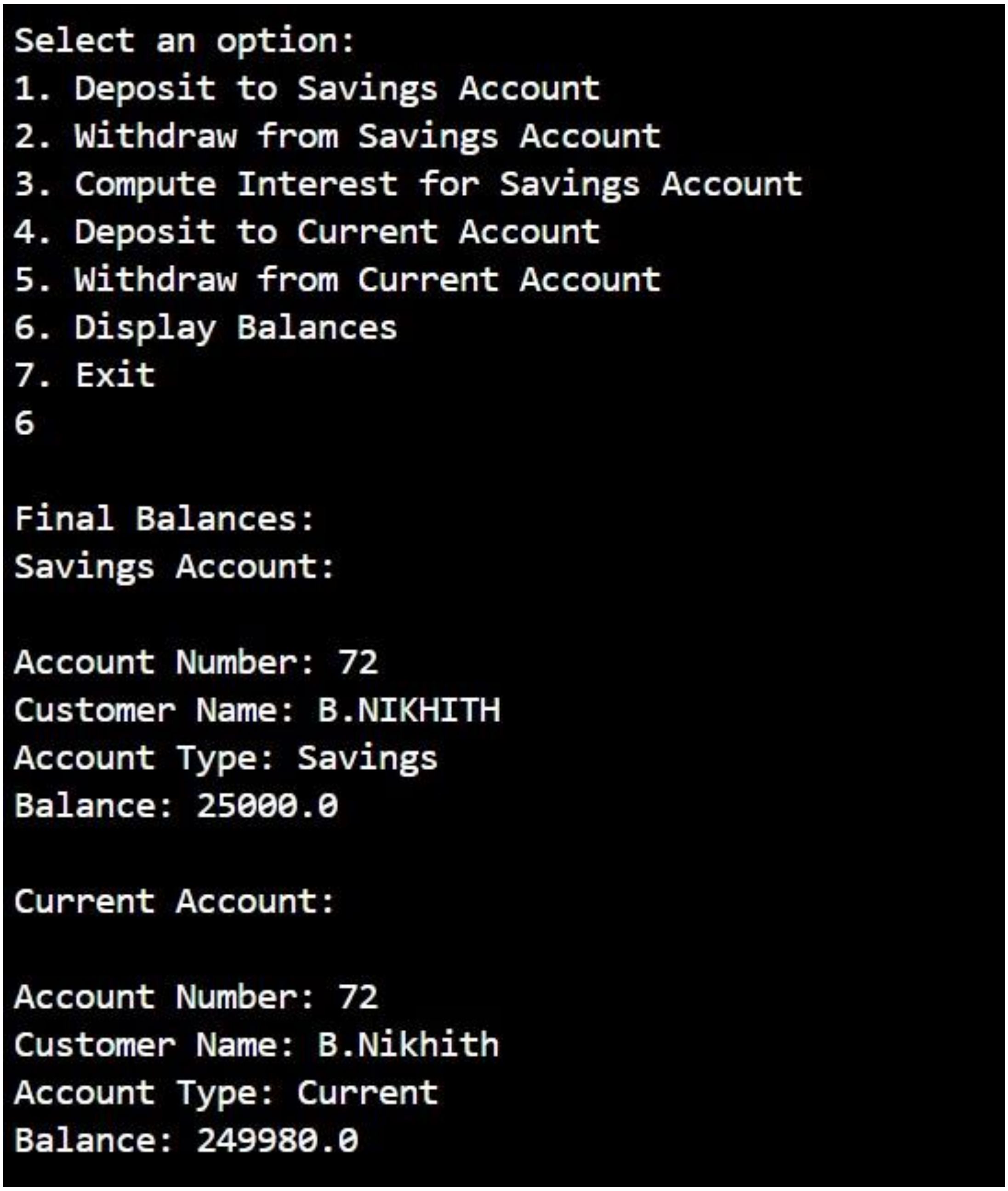
Output:



```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac bankMain.java } ; if
($?) { java bankMain }
Savings Account:
Enter customer name: B.NIKHITH
Enter account number: 072
Enter initial balance: 25000
Current Account:
Enter customer name: B.Nikhith
Enter account number: 072
Enter balance: 250000

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
5
Enter withdrawal amount for Current Account: 2000000000000000

Insufficient funds. Withdrawal failed.
```



```
Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
5
Enter withdrawal amount for Current Account: 20

Withdrawal of 20.0 successful. Updated balance: 249980.0
```



```
Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
6

Final Balances:
Savings Account:

Account Number: 72
Customer Name: B.NIKHITH
Account Type: Savings
Balance: 25000.0

Current Account:

Account Number: 72
Customer Name: B.Nikhith
Account Type: Current
Balance: 249980.0
```

6.Create a package CIE which has two classes- Student and Internals. The class Student has members like usn, name, sem. The class internals derived from student 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.

- **Create a folder CIE and save the programs Student.java and Internals.java within it.**
- **Create a folder SEE and save the program External.java within it.**
- **Save the Main program outside these two folders.**
- **Compile Main.java and Execute the**

Main.class ***Ans:***

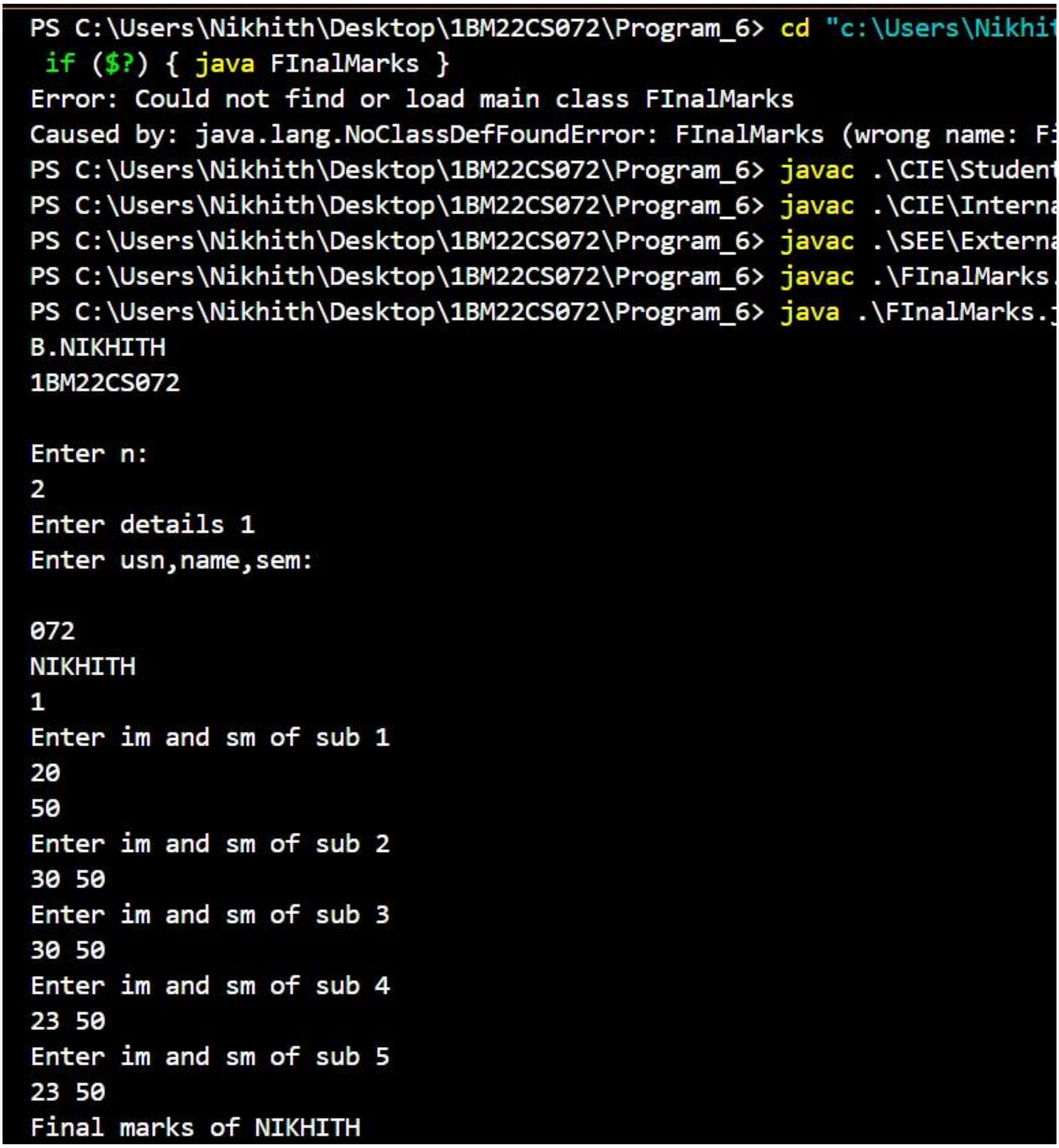
```
//CIE PACKAGE
//Student.java
package CIE;
public class Student { public String usn, name; public int sem;
public Student(String usn, String name, int sem) { this.usn = usn;
this.name = name; this.sem = sem;
}
}

//Internals.java
package CIE;
public class Internals extends Student {
public double[] internalMarks = new double[5];
public Internals(String usn, String name, int sem, double[]
internalMarks) { super(usn, name, sem);
this.internalMarks = internalMarks;
}
}
```

```
//SEE PACKAGE

//External.java
package SEE;
import CIE.Student;
public class External extends Student { public double[] seeMarks =
new double[5];
public External(String usn, String name, int sem, double[] seeMarks)
{ super(usn, name, sem);
this.seeMarks = seeMarks;
}
}

//FinalMarks.java
import CIE.Internals;
import SEE.External;
import java.util.Scanner;
public class FinalMarks {
public static void main(String[] args) {
System.out.println("B.NIKHITH");
System.out.println("1BM22CS072"); Scanner input = new
Scanner(System.in);

System.out.println("Enter the number of students: ");
int n = input.nextInt();
input.nextLine();



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




for (int i = 0; i < n; i++) {
 System.out.println("Enter details of Student " + (i + 1));
System.out.println("Enter USN: ");
String usn = input.nextLine();
```

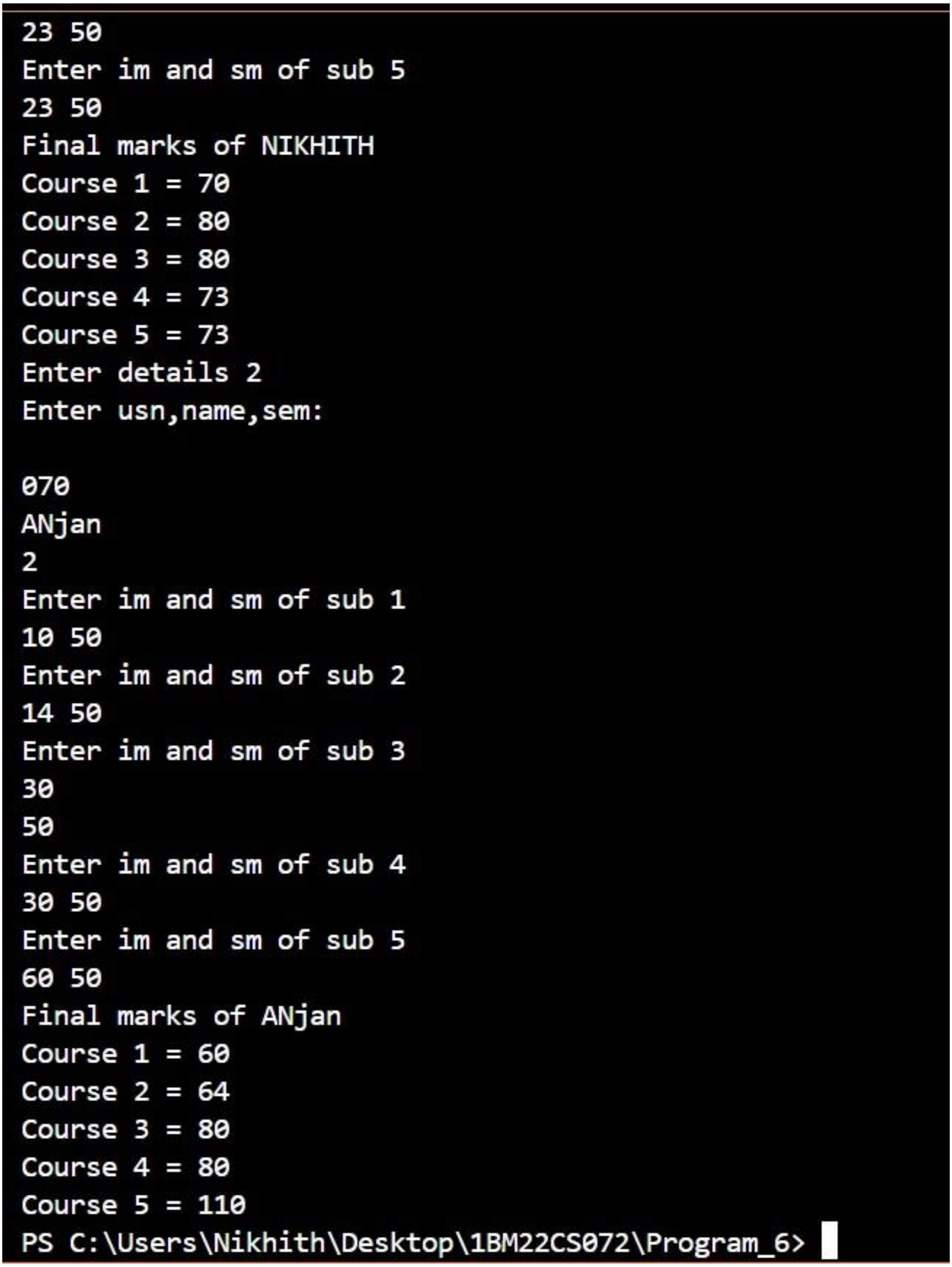
```
System.out.println("Enter Name: ");
String name = input.nextLine();
System.out.println("Enter Semester: ");
int sem = input.nextInt();
input.nextLine();
double[] internalMarks = new double[5];
System.out.println("Enter Internal Marks for 5 courses: ");
for (int j = 0; j < 5; j++) {
System.out.println("Enter Internal Marks for Course " + (j + 1) + ":
");
internalMarks[j] = input.nextDouble();
}
input.nextLine();


internals[i] = new Internals(usn, name, sem, internalMarks);
}
for (int i = 0; i < n; i++) {
System.out.println("Enter SEE Marks for Student " + (i + 1));
double[] seeMarks = new double[5];
for (int j = 0; j < 5; j++) {
System.out.println("Enter SEE Marks for Course " + (j + 1) + ": ");
seeMarks[j] = input.nextDouble();
}
input.nextLine();


externals[i] = new External(internals[i].usn, internals[i].name,
internals[i].sem, seeMarks);
}
System.out.println("\nFinal Marks of Students:");

for (int i = 0; i < n; i++) {
System.out.println("Student " + (i + 1) + " : USN: " + internals[i].usn
+ "\nName: " + internals[i].name + "\nSemester: " +
internals[i].sem);
```

```
for (int j = 0; j < 5; j++) { System.out.println("Subject " + (j + 1) + ":
" +
((internals[i].internalMarks[j]) + (externals[i].seeMarks[j] / 2)) +
"\n");
}
System.out.println();
}
}
}
```

Output:



```
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6> cd "c:\Users\Nikhit
 if ($?) { java FInalMarks }
Error: Could not find or load main class FInalMarks
Caused by: java.lang.NoClassDefFoundError: FInalMarks (wrong name: Fi
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6> javac .\CIE\Student
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6> javac .\CIE\Interna
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6> javac .\SEE\Externa
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6> javac .\FInalMarks.
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6> java .\FInalMarks.j
B.NIKHITH
1BM22CS072

Enter n:
2
Enter details 1
Enter usn,name,sem:

072
NIKHITH
1
Enter im and sm of sub 1
20
50
Enter im and sm of sub 2
30 50
Enter im and sm of sub 3
30 50
Enter im and sm of sub 4
23 50
Enter im and sm of sub 5
23 50
Final marks of NIKHITH
```



```
23 50
Enter im and sm of sub 5
23 50
Final marks of NIKHITH
Course 1 = 70
Course 2 = 80
Course 3 = 80
Course 4 = 73
Course 5 = 73
Enter details 2
Enter usn,name,sem:

070
ANjan
2
Enter im and sm of sub 1
10 50
Enter im and sm of sub 2
14 50
Enter im and sm of sub 3
30
50
Enter im and sm of sub 4
30 50
Enter im and sm of sub 5
60 50
Final marks of ANjan
Course 1 = 60
Course 2 = 64
Course 3 = 80
Course 4 = 80
Course 5 = 110
PS C:\Users\Nikhith\Desktop\1BM22CS072\Program_6>
```

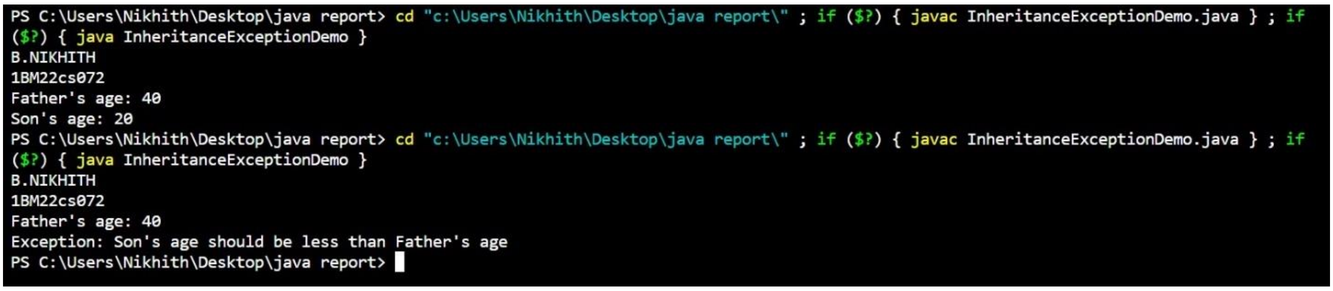
7.Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son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

Ans:

```
class WrongAgeException extends Exception {
    WrongAgeException(String message) {
        super(message);
    }
}

class Father {
    private int age;

    public Father(int age) throws WrongAgeException {
        if (age < 0) {
            throw new WrongAgeException("Father's age cannot be
negative");
        }
        this.age = age;
    }

    public int getAge() {
        return age;
    }
}
```

```
class Son extends Father {
    private int sonAge;

    public Son(int fatherAge, int sonAge) throws
WrongAgeException {
        super(fatherAge);

        if (sonAge >= fatherAge) {
            throw new WrongAgeException("Son's age should be less
than Father's age");
        }

        this.sonAge = sonAge;
    }

    public int getSonAge() {
        return sonAge;
    }
}

public class InheritanceExceptionDemo {
    public static void main(String[] args) {
System.out.println(“B .NIKHITH”);
System.out.println(“1BM22CS072”);
        try {
            Father father = new Father(40);
            System.out.println("Father's age: " + father.getAge());

            Son son = new Son(40, 20); // This will throw an
exception due to son's age being >= father's age
            System.out.println("Son's age: " + son.getSonAge());
        } catch (WrongAgeException e) {
            System.out.println("Exception: " + e.getMessage());
        }
    }
}
```

Output:



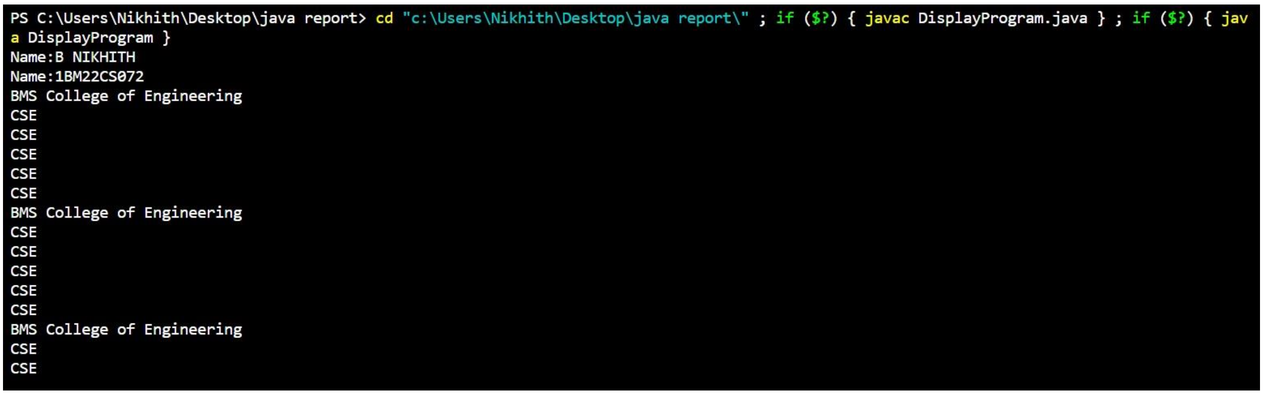
```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac InheritanceExceptionDemo.java } ; if
($?) { java InheritanceExceptionDemo }
B.NIKHITH
1BM22cs072
Father's age: 40
Son's age: 20
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac InheritanceExceptionDemo.java } ; if
($?) { java InheritanceExceptionDemo }
B.NIKHITH
1BM22cs072
Father's age: 40
Exception: Son's age should be less than Father's age
PS C:\Users\Nikhith\Desktop\java report>
```

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.

Ans:

```
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 DisplayProgram {
     public static void main(String[] args) {
     System.out.println(“Name:B .NIKHITH”);
     System.out.println(“Name:1BM22CS072”);
         DisplayThread thread1 = new DisplayThread("BMS College
 of Engineering", 10000); // 10 seconds
         DisplayThread thread2 = new DisplayThread("CSE",
 2000); // 2 seconds

         thread1.start();
         thread2.start();
     }
 }
```

Output:



```
PS C:\Users\Nikhith\Desktop\java report> cd "c:\Users\Nikhith\Desktop\java report\" ; if ($?) { javac DisplayProgram.java } ; if ($?) { jav
a DisplayProgram }
Name:B NIKHITH
Name:1BM22CS072
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
```