VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

OBJECT ORIENTED JAVA PROGRAMMING

Submitted by

Shamil Ahamed (1BM21CS248)

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
Oct 2022-Feb 2023

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**" carried out by **Y. Shamil Ahamed(1BM21CS248),** who is bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022-23. The Lab report has been approved as it satisfies the academic requirements in respect of Java Lab **- (22CS3PCOOJ)** work prescribed for the said degree.

Name of the Lab-Incharge
Designation
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

`

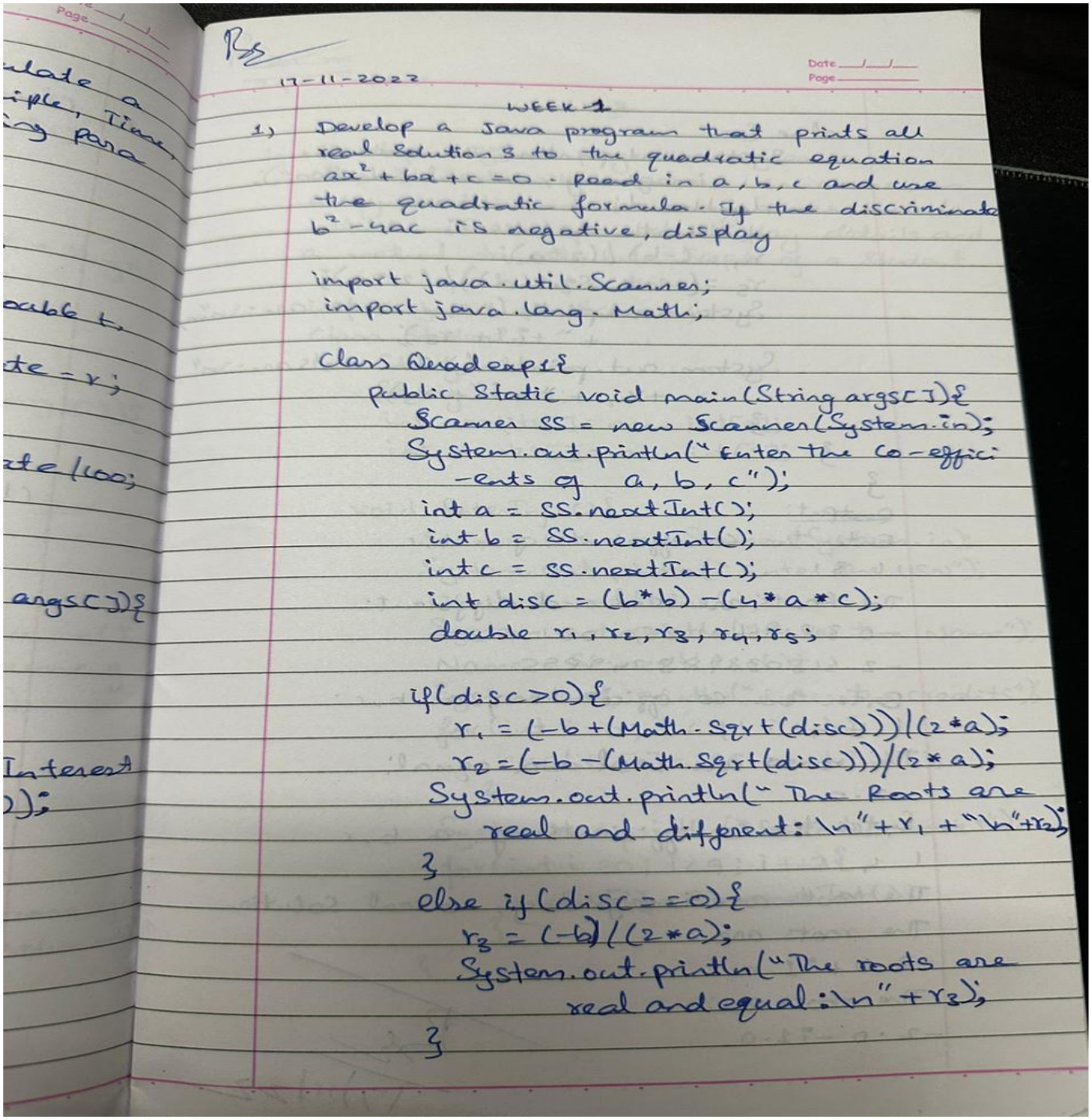
Index Sheet

Course Outcome

CO1	Apply the knowledge of java concepts to find the solution for a given solution
CO2	Analyze the given java application for correctness
CO3	Develop Java programs for a given requirement
CO4	Conduct practical experiments for demonstrating features of java

LAB PROGRAM 1:

Q: 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.



17-11-2023

WEEK-1

1) Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a, b, c and use the quadratic formula. If the discriminate $b^2 - 4ac$ is negative, display

```
import java.util.Scanner;
import java.lang.Math;

Class Quadexp1{
    public Static void main(String args[]){
        Scanner SS = new Scanner(System.in);
        System.out.println("Enter the Co-effici
            -ents of a, b, c");
        int a = SS.nextInt();
        int b = SS.nextInt();
        int c = SS.nextInt();
        int disc = (b*b) -(4*a*c);
        double r1, r2, r3, r4, r5;

    if(disc>0){
        r1 = (-b +(Math.Sqrt(disc)))/(2*a);
        r2 = (-b -(Math.Sqrt(disc)))/(2*a);
        System.out.println("The Roots are
            real and diffrent: \n" +r1, +"\n"+r2);
    }
    else if(disc==0){
        r3 = (-b)/(2*a);
        System.out.println("The roots are
            real and equal: \n" +r3);
    }
```

```
else {
    System.out.println("The roots are
        imaginary, no real solution");
    double d;
    d = Math.abs(disc);
    r4 = (-b) / (2*a);
    r5 = (Math.Sqrt(d)) / (2*a);
    System.out.println("The roots are:\n"
        + "+i" + r5);
    System.out.println("The roots are:\n"
        +r4 + "-i" + r5);
    }
  }
}
```

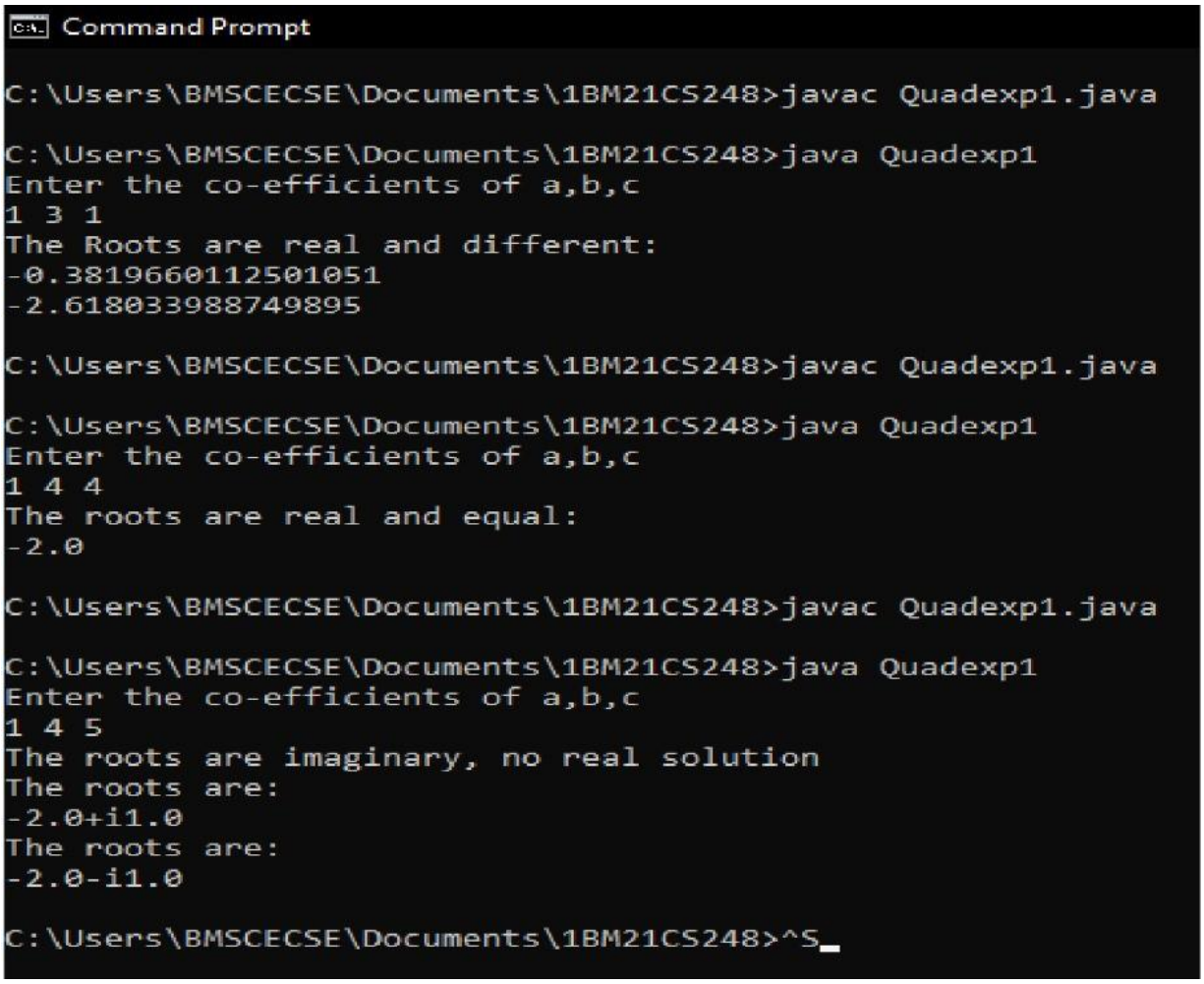
~~Output:~~

```
Enter the co-efficients of a, b, c
1 3 1
The Roots are real and different:
-0.3819660112501051
-2.618033988749895
Enter the co-efficients of a, b, c
1 4 4
The roots are real and equal:
-2.0
Enter the co-efficients of a, b, c
1 4 5
The roots are imaginary, no real solution
The roots are:
-2.0+i1.0
The roots are:
-2.0-i1.0
```

24-11-202

1) Develo
class
array
metho
a m
imp
cla

Output:


Command Prompt
C:\Users\BMSCECSE\Documents\1BM21CS248>javac Quadexp1.java
C:\Users\BMSCECSE\Documents\1BM21CS248>java Quadexp1
Enter the co-efficients of a,b,c
1 3 1
The Roots are real and different:
-0.3819660112501051
-2.618033988749895
C:\Users\BMSCECSE\Documents\1BM21CS248>javac Quadexp1.java
C:\Users\BMSCECSE\Documents\1BM21CS248>java Quadexp1
Enter the co-efficients of a,b,c
1 4 4
The roots are real and equal:
-2.0
C:\Users\BMSCECSE\Documents\1BM21CS248>javac Quadexp1.java
C:\Users\BMSCECSE\Documents\1BM21CS248>java Quadexp1
Enter the co-efficients of a,b,c
1 4 5
The roots are imaginary, no real solution
The roots are:
-2.0+i1.0
The roots are:
-2.0-i1.0
C:\Users\BMSCECSE\Documents\1BM21CS248>^S

LAB PROGRAM 2:

Q: 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.

24-11-2022

WEEK-3

1) Develop a java program to create a class Student with members USN, Name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a Student.

```
import java.util.Scanner;
Class Student{
    String USN;
    String Name;
    int credits[] = new int[9];
    int marks[] = new int[9];

  void EnterInfo(){
     Scanner ss = new Scanner(System.in);
     System.out.println("Enter the USN");
     USN = ss.nextLine();
     System.out.println("Enter the Name");
     Name = ss.nextLine();
     System.out.println("Enter the credits");
     for(int i=0; i<9; i++){
        credits[i] = ss.nextInt();
      }
     System.out.println("Enter the marks");
     for(int i=0; i<9; i++){
        marks[i] = ss.nextInt();
      }
  }
```

```
void DisplayInfo(){
    System.out.println("\n PRINTING
          STUDENT INFO: \n");
    System.out.println("USN:" + USN);
    System.out.println("NAME:" + Name);
    System.out.print("CREDITS: ");
    for(int i=0; i<9; i++){
        System.out.print(credits[i]+" ");
    }
    System.out.println();
    System.out.print("MARKS:" + " ");
    for (int i=0; i<9; i++){
        System.out.println(marks[i]+" ");
    }
}
float CalculateSGPA(){
    float Sgpa;
    float totalcredits=0;
    for (int i=0; i<9; i++){
        totalcredits += credits[i];
    }
    float Sumgp=0;
    float gp=0;
    for(int i=0; i<9; i++){
        gp = (((marks[i])/10)+1);
        Sumgp += credits[i] * gp;
    }
    Sgpa = Sumgp/totalcredits;
    return Sgpa;
}
}
```

```
Class SGPA{
    public static void main(String args[]){
        Student S1 = new Student();
        S1.EnterInfo();
        S1.DisplayInfo();
        float Sgpa = S1.CalculateSGPA();
        System.out.println("\n SGPA: " +sgpa);
    }
}
```

Output:

```
Enter the USN
1BM21CS249
Enter the Name
Zoro
Enter the credits
3 4 1 3 1 3 1 3 1
Enter the marks
66 49 82 43 65 33 46 44 75

PRINTING STUDENTS INFO:

USN: 1BM21CS249
NAME: Zoro
CREDITS: 3 4 1 3 1 3 1 3 1
MARKS: 66 49 82 43 65 33 46 44 75
SGPA: 5.6
```

24/11/22

Output:

```
Enter the USN
1BM21CS249
Enter the Name
Zoro
Enter the credits
3 4 1 3 1 3 1 3 1
Enter the marks
66 49 82 43 65 33 46 44 75

PRINTING STUDENTS INFO:

USN: 1BM21CS249
NAME: Zoro
CREDITS: 3 4 1 3 1 3 1 3 1
MARKS: 66 49 82 43 65 33 46 44 75
SGPA: 5.6
```

LAB PROGRAM 3:

Q: 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.

Bs

01-12-2022

WEEK 3

```
Import java.util.Scanner;

Class Book{
    String Name;
    String author;
    int price;
    int num_pages;
    Book(String N, String Au, int P, int Pa){
        Name = N;
        author = Au;
        price = P;
        num_pages = Pa;
    }
    Book(){ }
    void setdetails(){
        Scanner SS = new Scanner(System.in);
        System.out.println("Enter the book Name:");
        Name = SS.nextLine();
        System.out.println("Enter the Author Name:");
        author = SS.nextLine();
        System.out.println("Enter the price:");
        price = SS.nextInt();
        System.out.println("Enter the No.of Pages:");
        num_pages = SS.nextInt();
        System.out.println("Details SET
                            SUCCESFULLY");
    }
```

```
void getdetails(){
    System.out.println("NAME: " + Name);
    System.out.println("AUTHOR:" + author);
    System.out.println("PRICE:" + price);
    System.out.println("NO. of PAGES:"+num_pages);
  }
public String toString(){
    return("NAME: " + Name + "\n AUTHOR:"
      + author + "\n PRICE: " + price + "\n NO. of
                  Pages: " + num_pages);
  }
}
class Bookdemo{
    public static void main(String args[]){
    int n;
    Scanner ss = new Scanner(System.in);
    System.out.println("Enter the no. of Books");
    n=ss.nextInt();
    Book b1[] = new Book[n];
    for(int i=0; i<n; i++){
      System.out.println("Enter the Details: ");
      System.out.println("1:Set using methods
          \n 2: Set using constructor");
      int ch= ss.nextInt();
      if(ch==1){
        b1[i] = new Book();
        b1[i].setdetails();
      }
      else{
        b1[i] = new Book("Kings", "Kingsley",
            1000, 1500);
      }
}
```

```
for (int i=0; i<n; i++){
    System.out.println("\n PRINTING THE
                        DETAILS");
    System.out.println("1: Display using
    methods \n 2: Display using toString");
    int ch= ss.nextInt();
    if(ch==1){
        b1[i].getdetails();
        System.out.println();
    }
    else{
        String details = b1[i].toString();
        System.out.println(details);
    }
  }
 }
}
```

OUTPUT:

```
Enter the no of Books:
2
ENTER DETAILS:
1:Set using methods
2: Set using constructor
1
Enter the book Name:
haha
Enter the Author Name:
hahaha
Enter the price:
100
```

Enter the no. of pages:
200
DETAILS SET SUCCESFULLY
ENTER DETAILS:
1: Set using methods
2: Set using constructor
2
PRINTING
~~Printing~~ BOOK DETAILS
1: Display using methods
2: Display using toString
1
~~NAME: kohen~~ NAME: haha
~~AUTHOR: kohetha~~ AUTHOR: hahaha
~~PRICE: 100~~ PRICE: 100
~~NO. of PAGES: 200~~ NO. OF PAGES: 200

PRINTING BOOK DETAILS
1: Display using methods
2: Display using to String
2
~~NAME~~: Kings
AUTHOR: Kingsley
PRICE: 1000
NO. of pages: 1500.

01/12/22

Output:

```
C:\Users\Admin\Documents>javac Bookdemo.java

C:\Users\Admin\Documents>java Bookdemo
Enter the no of Books:
2
ENTER DETAILS:
1:set using methods
2:set using constructor
1
Enter the book Name:
haha
Enter the Author Name:
hahaha
Enter the Price:
100
Enter the No.Of Pages:
200
DETAILS SET SUCCESFULLY
ENTER DETAILS:
1:set using methods
2:set using constructor
2

 PRINRING BOOK DETAILS
1:Display using methods
2:Display using toString
1
NAME: haha
AUTHOR: hahaha
PRICE: 100
NO.OF PAGES: 200


 PRINRING BOOK DETAILS
1:Display using methods
2:Display using toString
2
NAME: Kings
AUTHOR: Kingsley
PRICE: 1000
No.Of Pages: 1500
```

LAB PROGRAM 4:

Q: 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.

08-12-2022

WEEK-5

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

```
abstract class Shape{
    int a, b;
    Shape(int a, int y){
        a=x;
        b=y;
    }
    Shape(int x){
        a=x;
    }
    void printArea(){
    }
}
class rectangle extends Shape{
    rectangle(int x, int y){
        super(a,y);
    }
    void printArea(){
        System.out.println("area of rectangle
                is: "+(a*b));
    }
}
```

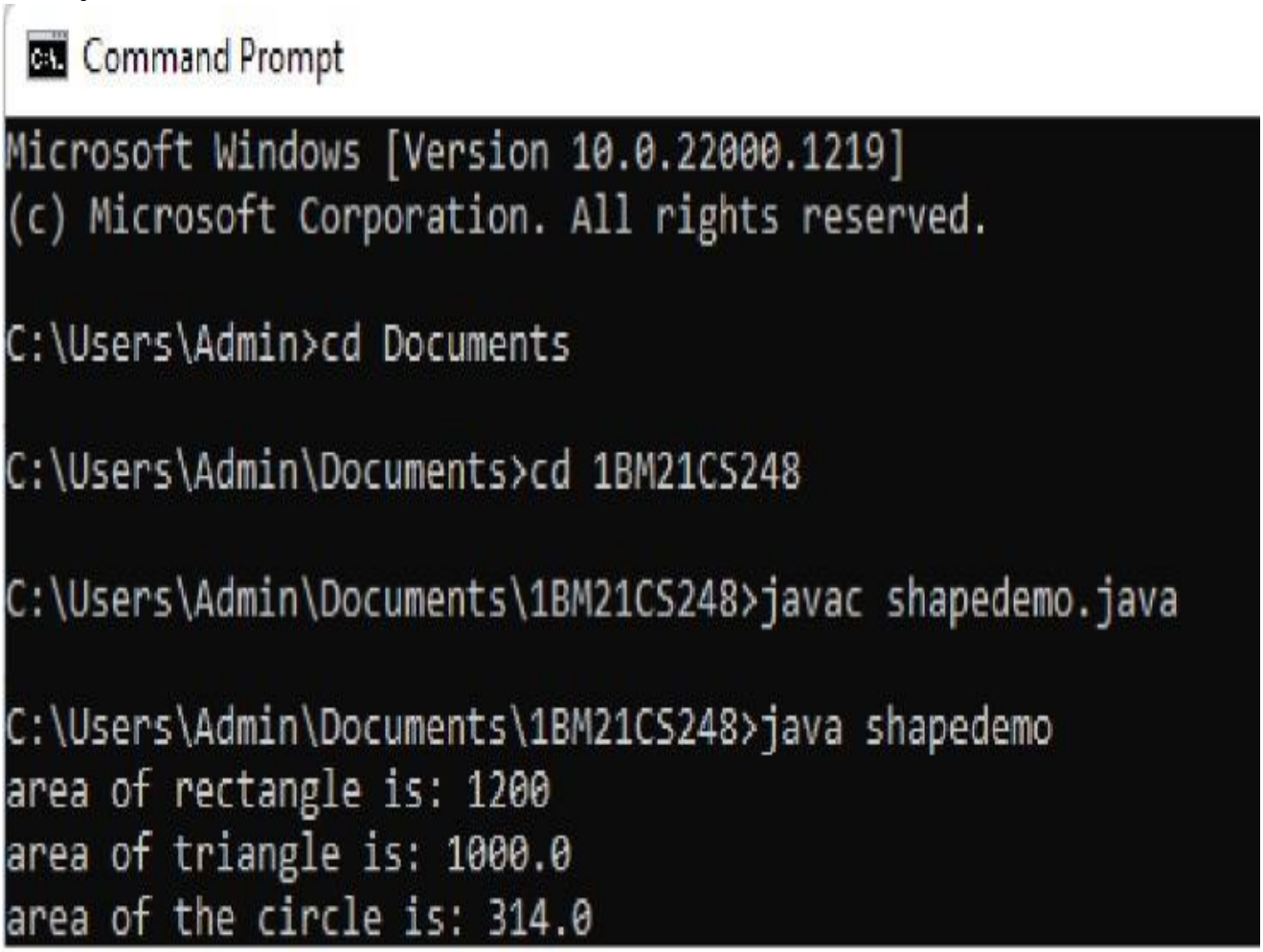
```
class triangle extends Shape{
    triangle(int a, int y){
        super(x, y);
    }
    void printArea(){
        System.out.println("area of
            triangle is: "(0.5 * a * b));
    }
}
class circle extends Shape{
    circle(int x){
        super(x);
    }
    void printArea(){
        System.out.println("area of circle
            is: " + (3.14 * a * a));
    }
}
class Shapedemo{
    public static void main(String args[]){
        rectangle r1 = new rectangle(30,40);
        triangle t1 = new triangle(50,40);
        Circle c1 = new Circle(10);
        r1.printArea();
        t1.printArea();
        c1.printArea();
    }
}
```

output:

Area of rectangle is: 1200.00
area of triangle is: 1000.00
area of circle is: 314.00

08/12/22

Output:


Command Prompt
Microsoft Windows [Version 10.0.22000.1219]
(c) Microsoft Corporation. All rights reserved.
C:\Users\Admin>cd Documents
C:\Users\Admin\Documents>cd 1BM21CS248
C:\Users\Admin\Documents\1BM21CS248>javac shapedemo.java
C:\Users\Admin\Documents\1BM21CS248>java shapedemo
area of rectangle is: 1200
area of triangle is: 1000.0
area of the circle is: 314.0

LAB PROGRAM 5:

Q: 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.

```
import java.util.Scanner;

class account{
    String Name;
    int Acc_num;
    String Acc_type;
}
class Sav_acc extends account{
    double balance;
    Sav_acc (String N, int An, String At, double B){
        Name = N;
        Acc_num = An;
        Acc_type = At;
        balance = B;
    }
    Scanner ss = new Scanner(System.in);
    void deposit(int val){
        balance += val;
    }
    void display_bal(){
        System.out.println("YOUR BALANCE IS: "
                + balance);

    }
    void deposit_interest(){
        double interest_rate = 0.05;
        double time = 0;
        System.out.println("ENTER THE TIME
                PERIOD");

        time = ss.nextDouble();
        double amount;
        amount = balance * Math.pow((1+
            interest_rate), time);
```

```
        balance = amount;
    }
    void withdraw(int val){
        if (val > balance){
            System.out.println("out of funds
                withdraw lesser amount");
        }
        else{
            balance -= val;
            System.out.println("withdrawal
                                successful");
            System.out.println("new balance"
                                + balance);
        }
    }
    void check_min(){
        double min_bal = 2000.0;
        double penalty = 100.00;
        if(balance < min_bal){
            System.out.println("Balance less
                than Minimum balance,
                penalty imposed");
            balance -= penalty;
        }
        else{
            System.out.println("Balance
                higher than minimum balan
                no penalty");
        }
    }
}
```

```
class cur_acc extends account{
    double balance;
    cur_acc(String N, int An, String At,
                    Double B){
        name = N;
        Acc_num = An;
        Acc_type = At;
        balance = B;
    }
void deposit(int val){
    balance += val;
  }
void display_bal(){
    System.out.println("Your Balance is :"+balance);
  }
void deposit_interest(){
    System.out.println("No deposit provided for
                        current Account");
  }
void withdraw(){
    System.out.println("No withdrawal provided
                        for current Account");
}
void cheque_withdrawal(int val){
    balance -= val;
    System.out.println("withdrawal successful");
    System.out.println("new balance: "+balance);
}
void check_min(){
    double min_bal = 2000.00;
    double penalty = 100.00;
```

```
        if(balance<min_bal){
          System.out.println("Balance lower than
                  minimum balance. penalty imposed");
          balance -= penalty;
        }
        else{
          System.out.println("Balance higher than
                  minimum balance. No penalty");
        }
    }
}
Class Bank{
    public static void main(String args[]){
      Scanner SS = new Scanner(System.in);
      System.out.println("Enter your name,
            account number, account type (Savings/
              current), balance ");
      String Name = SS.nextLine();
      int Acc_num = SS.nextInt();
      String Acc_type = SS.nextLine();
      double balance = SS.nextDouble();
      if(Acc_type.equal("Savings")){
        Sav_acc a1 = new Sav_acc(Name,
                Acc_num, Acc_type, balance);
        int choice = 0;
        while(choice != 6){
          System.out.println(" 1.Deposit\n
            2. Display Balance \n 3. Compute
            and deposit interest \n 4. withdraw
            \n 5. check for minimum balance\n
            6. Exit ");
          Choice = SS.nextInt();
                                                }
```

```
switch(choice){
    case 1:
        System.out.println("Enter the value
          of to deposit");
        int val = ss.nextInt();
        a1.deposit(val);
        break;
    case 2:
        a1.display_bal();
        break;
    case 3:
        a1.deposit_interest();
        break;
    case 4:
        System.out.println("Enter the value
                    to withdraw");
        int withd = ss.nextInt();
        a1.withdraw(withd);
        break;
    case 5:
        a1.check.min();
        break;
    case 6:
        System.out.println("Exited");
        break;
    default:
        System.out.println("Enter a valid
                    Choice");
        break;
    }
}
}
```

```
else {
    curr_acc a1 = new curr_acc(Name, Acc_num,
                Acc_type, balance);
    int choice = 0;
    while(choice != 7){
        System.out.println("1. Deposit\n
        2. Display balance\n 3. compute and
        deposit Interest\n 4. withdraw\n
        5. cheque withdrawal\n 6. check
        minimum balance\n 7. Exit");
        choice=SS.nextInt();
        Switch(choice){
            case 1:
                System.out.println("Enter the
                    value to deposit");
                int val = ss.nextInt();
                a1.deposit(val);
                break;
            Case 2:
                a1.display_bal();
                break;
            case 3:
                a1.deposit_interest();
                break;
            case 4:
                a1.withdraw();
                break;
            case 5:
                System.out.println("Enter the value
                    to withdraw");
                int val1 = ss.nextInt();
                a1.Cheque_withdrawal(val1);
```

```
            case 6:
                a1.check_min();
                break;
            case 7:
                System.out.println("Exited");
                break;
            default:
                System.out.println("Enter a valid
                                    choice");
                break;
        }
    }
}
}
```

output:

Savings account:

Enter your name, account number, account type(savings/current), balance

Zoro

12345000

Savings

2000

1. Deposit

2. Display Balance

3. compute and deposit Interest

4. withdraw

5. Check for minimum balance

6. Exit

1

Enter the value to deposit

500

1. Deposit
2. Display Balance
3. compute and deposit interest
4. withdraw
5. check for minimum balance
6. Exit.
2
your Balance is : 2500.0
3
Enter the time period
2
2
your balance is: 2756.25
4
Enter the value to withdraw
3000
out of funds, withdraw lesser amount
4
Enter the value to withdraw
500
withdrawal Successful.
new balance : 2256.25
5
Balance higher than minimum balance.
6
Exited.

Output

current Account

Enter your name, account number, account type (savings/current), balance.

ITACHI

123005000

current

2000

1. Deposit

2. Display Balance

3. compute and deposit interest

4. withdraw

5. cheque withdrawal

6. check for minimum balance

7. Exit

1

Enter value to deposit

500

2

Your balance is: 2500.0

3

No interest provided for current Account

4

No withdrawal provided for current Account

5

Enter value to withdraw

3000

withdrawal successful

new balance: -500.0

Balance lesser than minimum balance, penalty imposed.

1
Enter value to deposit
5000
2
your Balance is: 2400.0
6
Balance higher than minimum balance, no fee
7
Exited.

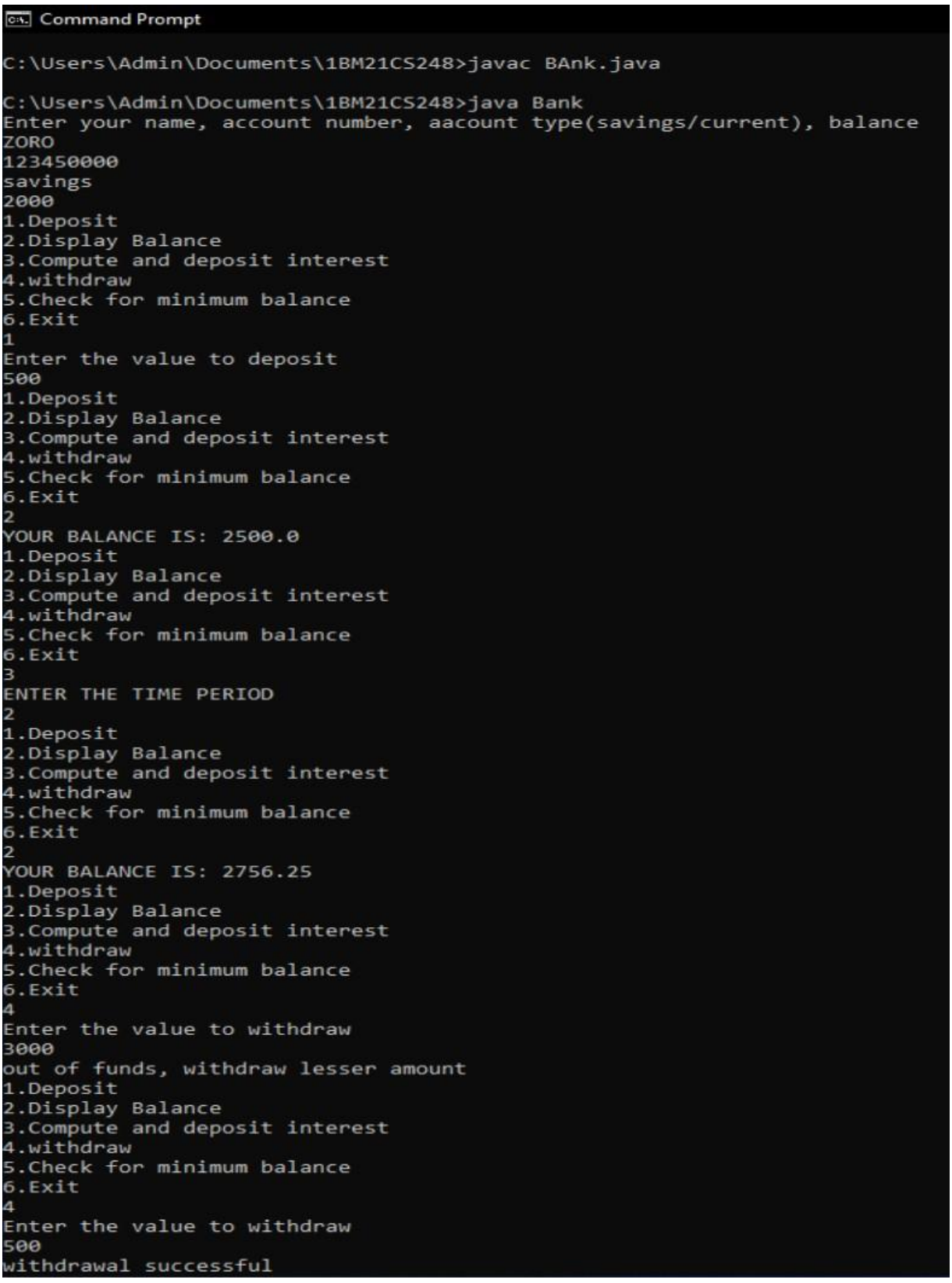
22/01/23

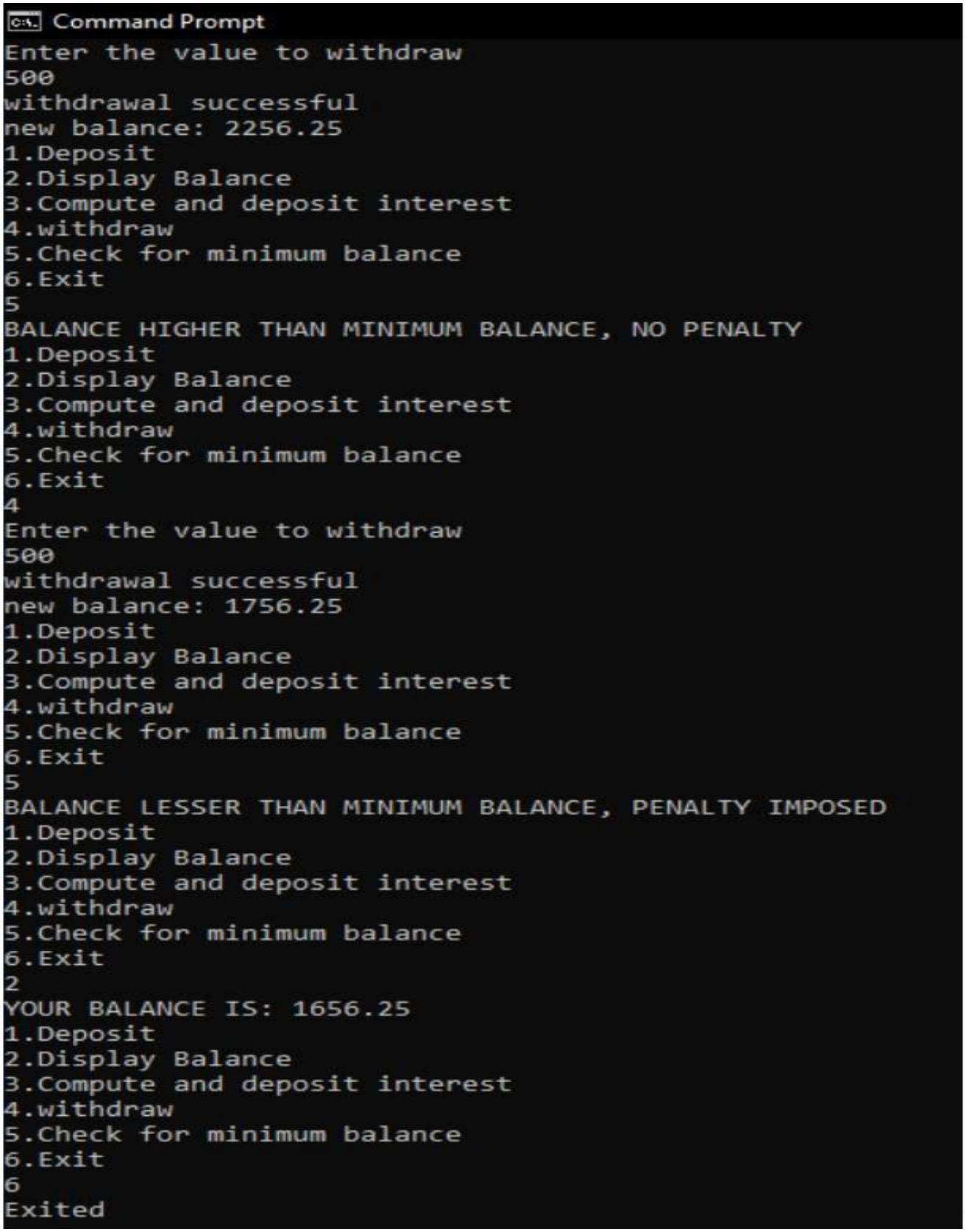
05 WAP th
in in
called
"Son"
father
takes
wrong

import
class

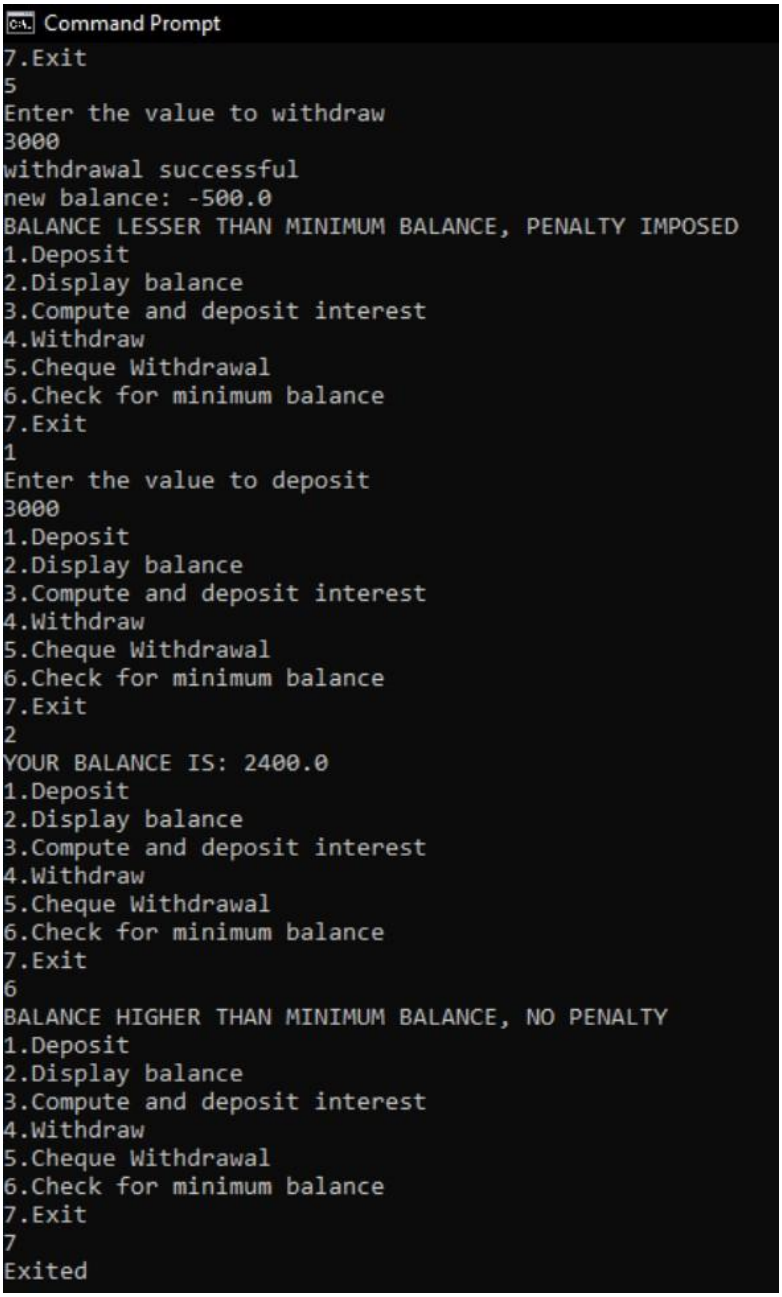
3
cla

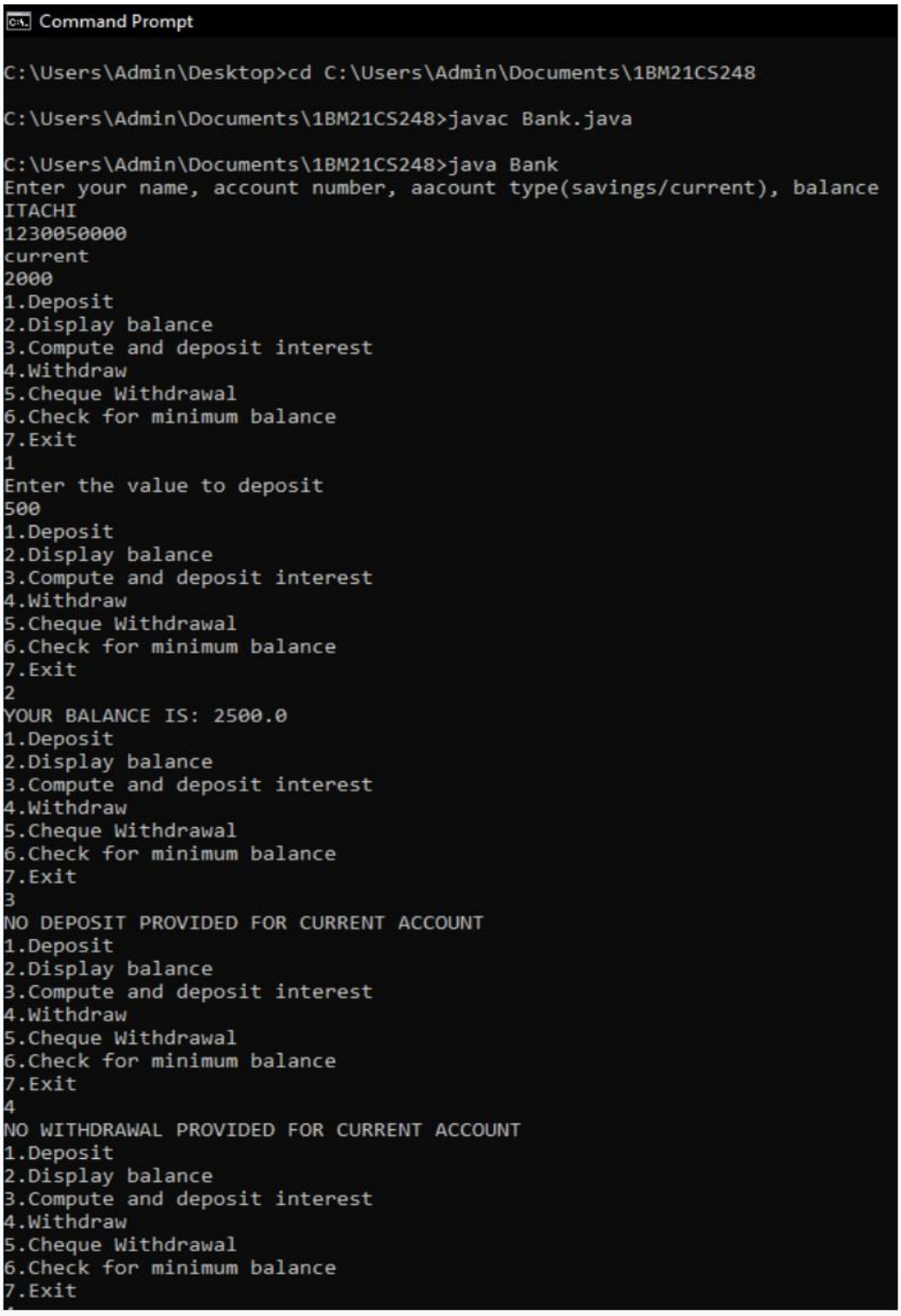
Output for savings:




Command Prompt
Enter the value to withdraw
500
withdrawal successful
new balance: 2256.25
1.Deposit
2.Display Balance
3.Compute and deposit interest
4.withdraw
5.Check for minimum balance
6.Exit
5
BALANCE HIGHER THAN MINIMUM BALANCE, NO PENALTY
1.Deposit
2.Display Balance
3.Compute and deposit interest
4.withdraw
5.Check for minimum balance
6.Exit
4
Enter the value to withdraw
500
withdrawal successful
new balance: 1756.25
1.Deposit
2.Display Balance
3.Compute and deposit interest
4.withdraw
5.Check for minimum balance
6.Exit
5
BALANCE LESSER THAN MINIMUM BALANCE, PENALTY IMPOSED
1.Deposit
2.Display Balance
3.Compute and deposit interest
4.withdraw
5.Check for minimum balance
6.Exit
2
YOUR BALANCE IS: 1656.25
1.Deposit
2.Display Balance
3.Compute and deposit interest
4.withdraw
5.Check for minimum balance
6.Exit
6
Exited

Output for current:




Command Prompt
C:\Users\Admin\Desktop>cd C:\Users\Admin\Documents\1BM21CS248
C:\Users\Admin\Documents\1BM21CS248>javac Bank.java
C:\Users\Admin\Documents\1BM21CS248>java Bank
Enter your name, account number, aacount type(savings/current), balance
ITACHI
1230050000
current
2000
1.Deposit
2.Display balance
3.Compute and deposit interest
4.Withdraw
5.Cheque Withdrawal
6.Check for minimum balance
7.Exit
1
Enter the value to deposit
500
1.Deposit
2.Display balance
3.Compute and deposit interest
4.Withdraw
5.Cheque Withdrawal
6.Check for minimum balance
7.Exit
2
YOUR BALANCE IS: 2500.0
1.Deposit
2.Display balance
3.Compute and deposit interest
4.Withdraw
5.Cheque Withdrawal
6.Check for minimum balance
7.Exit
3
NO DEPOSIT PROVIDED FOR CURRENT ACCOUNT
1.Deposit
2.Display balance
3.Compute and deposit interest
4.Withdraw
5.Cheque Withdrawal
6.Check for minimum balance
7.Exit
4
NO WITHDRAWAL PROVIDED FOR CURRENT ACCOUNT
1.Deposit
2.Display balance
3.Compute and deposit interest
4.Withdraw
5.Cheque Withdrawal
6.Check for minimum balance
7.Exit

LAB PROGRAM 7:

Q: 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.

Week-7

Q) WAP 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.

```
import java.util.Scanner;
class father{
    int father_age;
    public father(int fa){
        try{
            father_age = fa;
            if(father_age < 0){
                throw new Exception("Error! Age is
                        less than 0");
            }
            else{
                father_age = fa;
            }
        }catch(Exception e){
            System.out.println("caught: "+e);
        }
    }
}
class son extends father{
    int son_age;
    public son(int fa, int sa){
        super(fa);
```

```
try{
    Son_age = sa;
    if(Son_age < 0){
        throw new Exception("Error! Son's age
            less than 0");
    }
    else if(Son_age >= Father_age){
        throw new Exception("Error! Son's
            age cannot be more than
            father's age");
    }
    else{
        Son_age = sa;
    }
}catch(Exception e){
    System.out.println("caught: "+e);
}
}
void display(){
    System.out.println("father's age = "
                        + father_age);
    System.out.println("Son's age = "+Son_age);
}
}
class InheritanceTree extends Exception{
    public static void main(String args[]){
        Scanner ss = new Scanner(System.in);
        int a,b;
        System.out.println("Enter the father's
                            age");
        a = ss.nextInt();
```

```
        System.out.println("Enter the son's
                                    age");
        b= ss.nextInt();
        Son ob1= new Son(a,b);
        ob1.display();
    }
}
```

output:

```
Enter father's age
49
Enter the Son's age
19
Father's age = 49
Son's age = 19

Enter the father's age
49
Enter the Son's age
-2
caught: java.lang.Exception: Error! Son's age is
Father's age = 49                    less than 0
Son's age = -2

Enter the father's age
49
Enter the Son's age
50
caught: java.lang.Exception: Error! Son's age cannot
father's age = 49                    be more than the father's age.
Son's age = 50
```

Output:

```
Command Prompt

C:\Users\Admin\Documents\1BM21CS248>javac InheritanceTree.java

C:\Users\Admin\Documents\1BM21CS248>java InheritanceTree
Enter the father's age
49
Enter the son's age
19
Father's age = 49
Son's age = 19

C:\Users\Admin\Documents\1BM21CS248>javac InheritanceTree.java

C:\Users\Admin\Documents\1BM21CS248>java InheritanceTree
Enter the father's age
49
Enter the son's age
-2
Caught : java.lang.Exception: Error! Son's age is less than 0
Father's age = 49
Son's age = -2

C:\Users\Admin\Documents\1BM21CS248>javac InheritanceTree.java

C:\Users\Admin\Documents\1BM21CS248>java InheritanceTree
Enter the father's age
49
Enter the son's age
50
Caught : java.lang.Exception: Error! Son's age cannot be more than the Father's age
Father's age = 49
Son's age = 50

C:\Users\Admin\Documents\1BM21CS248>javac InheritanceTree.java

C:\Users\Admin\Documents\1BM21CS248>java InheritanceTree
Enter the father's age
-20
Enter the son's age
40
Caught : java.lang.Exception: Error! Age is less than 0
Caught : java.lang.Exception: Error! Son's age cannot be more than the Father's age
Father's age = -20
Son's age = 40
```

LAB PROGRAM 8:

Q: 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.

Experiment - 9

Q) WAP which creates two threads, one thread displaying "Bms College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

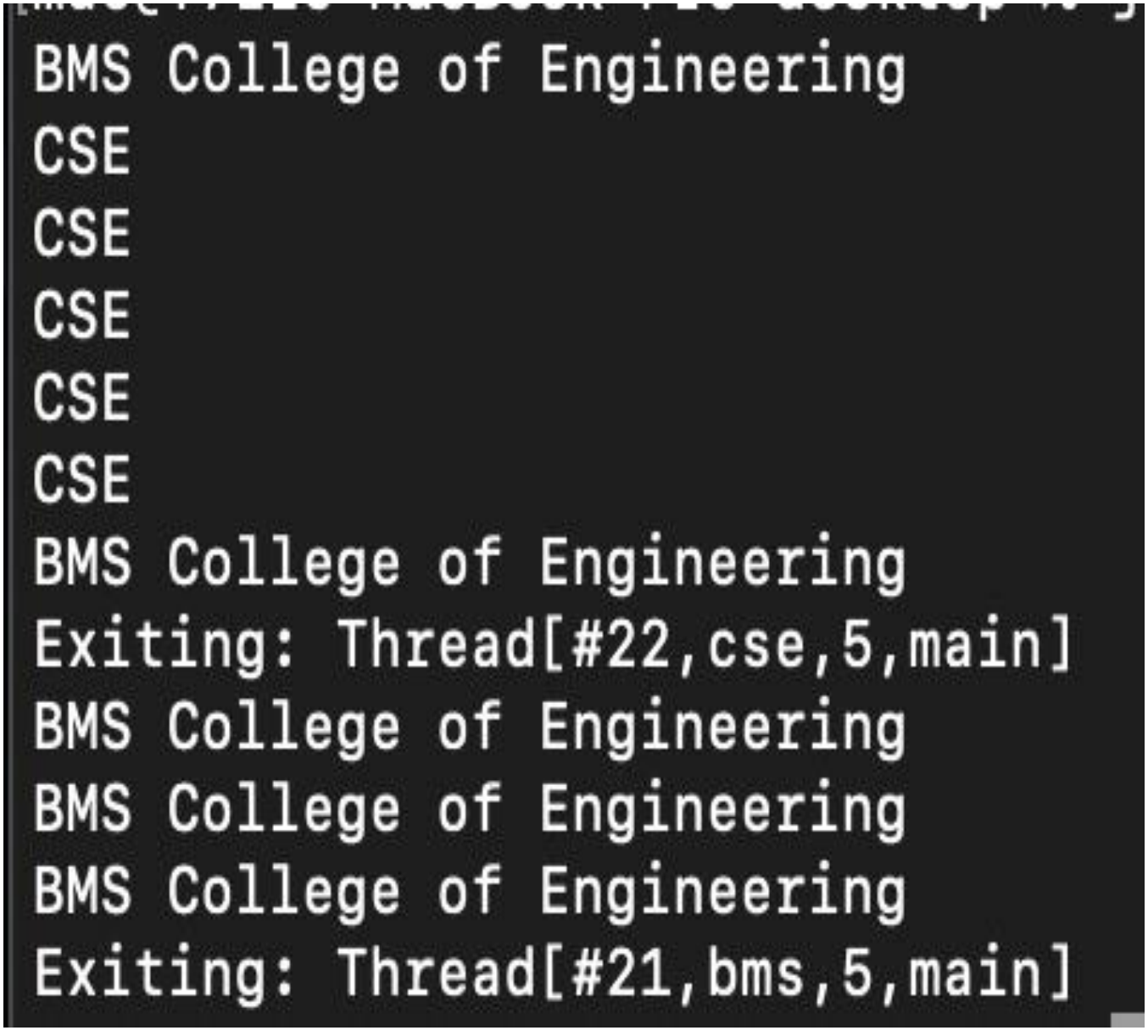
```
class bms implements Runnable{
    thread t1;
    bms(){
        t1 = new Thread(this, "bms");
    }
    public void run(){
        try{
            for(int i=1; i>0; i--){
                System.out.println("Bms College
                    of Engineering");
            }thread.sleep(10000);
        }catch(Exception e){};
        System.out.println("Exiting " +t1);
    }
}
class cse implements Runnable{
    thread t2;
    cse(){
        t2 = new Thread(this, "cse");
    }
    Public void run(){
        try{
            for(int i=0; i>0; i--){
                System.out.println("CSE");
                thread.sleep(2000);
            }
```

```
        } catch(Exception e){ };
          System.out.println("Exiting:"+t);
        }
    }
class threadprg{
    public static void main(String args[]){
        bms obj1 = new bms();
        CSe obj2 = new CSe();
        obj1.t1.start();
        obj2.t2.start();
    }
}
```

BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting: Thread(#22, CSe, 5, main)
BMS college of Engineering
BMS college of Engineering
BMS college of Engineering
Exiting: Thread(#21, bms, 5, main)

02/02/23

Output:


BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting: Thread[#22,cse,5,main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting: Thread[#21,bms,5,main]

LAB PROGRAM 9:

Q: 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.

Lab program - 9

Open ended Question

Q) 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 students. Import the two packages in a file that declares the final marks of n students in all five courses.

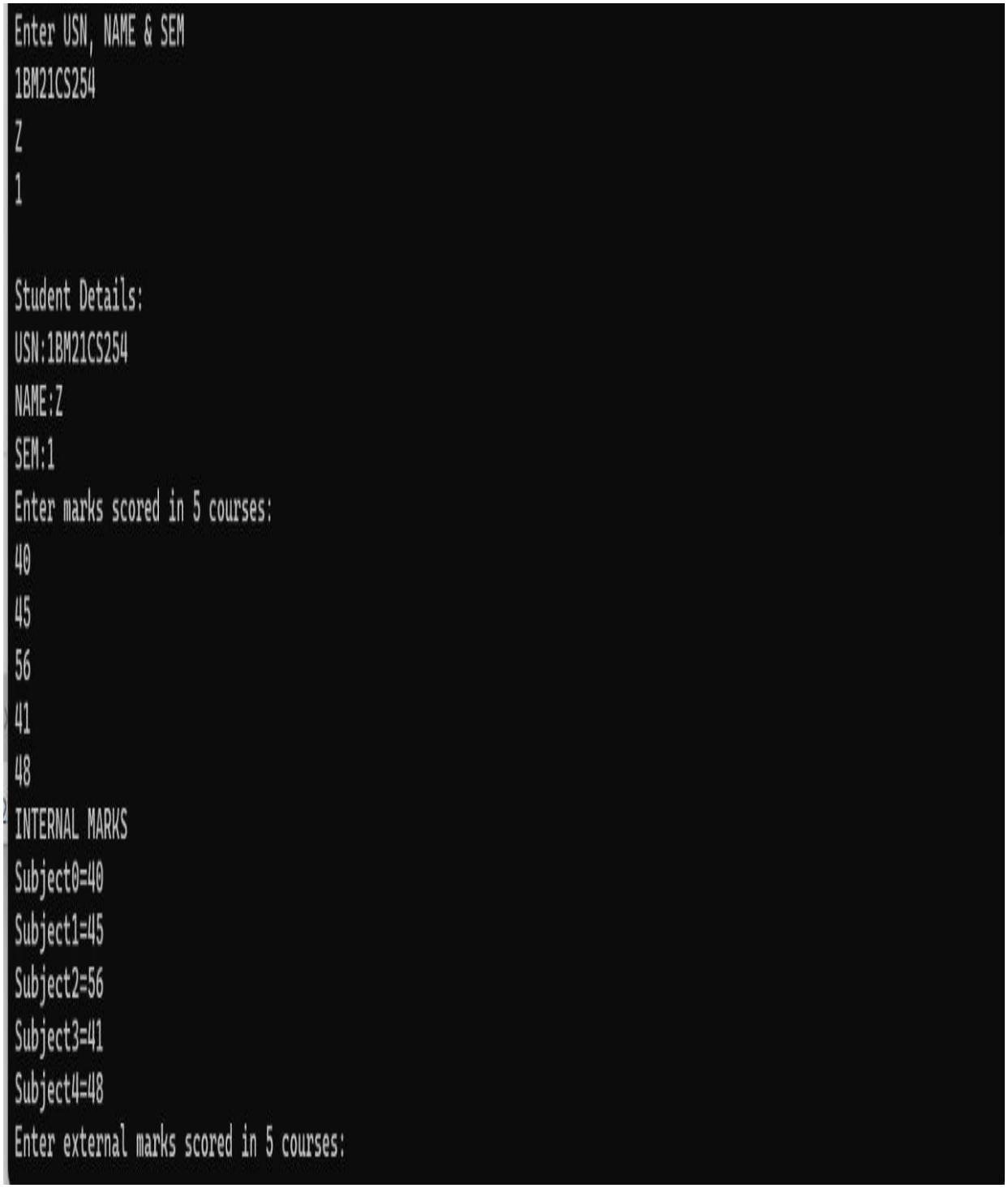
```
package CIE;
import java.util.Scanner;
public class Internals {
    public int Imarks[] = new int[5];
    public void getmi(){
        Scanner SS=new Scanner(System.in);
        System.out.println("enter marks scored
                            in 5 courses");
        for(int i=0; i<=4; i++){
            Imarks[i] = SS.nextInt();}
}
public void display(){
public void dispmi(){
    System.out.println("Internal marks");
    for(int i=0; i<=4; i++){
        System.out.println("Subject"+i+"="
                +Imarks[i]);}
```

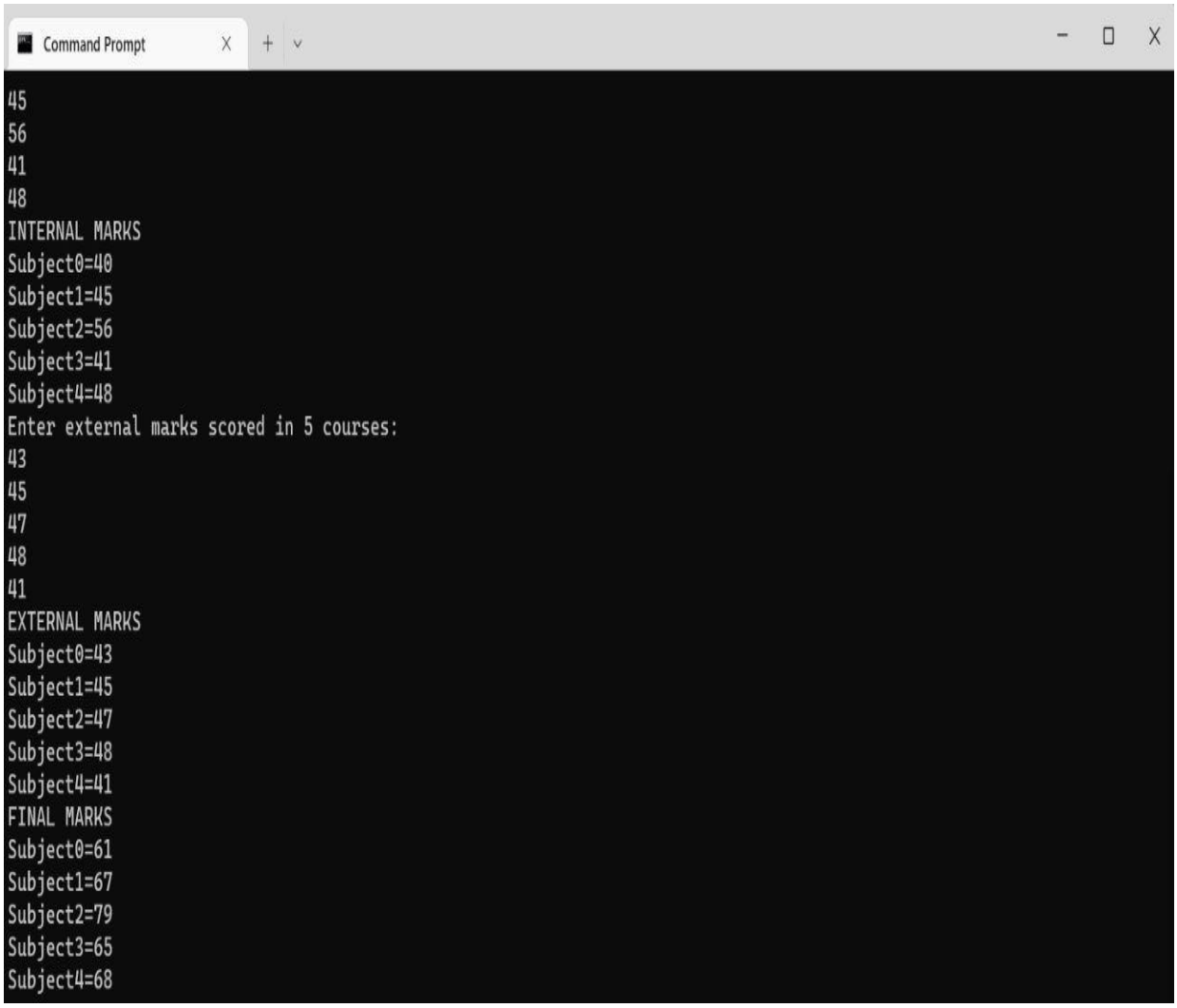
```
package SEE;
import java.util.Scanner;
import CIE.*;
public class External extends CIE.Internals{
        int Smarks[]= new int[5];
        public void getm(){
            Scanner ss= new Scanner(System.in);
            System.out.println("enter eat
                    marks scored is 5 course");
            for(int i=0; i<4; i++){
                Smarks[i] = ss.nextInt();
            }
    }
public void dispm(){
    System.out.println("external marks");
    for(int i=0; i<=4; i++){
        System.out.println("subject"+ i+"="
                + Smarks[i];
        }
}
public void finalcal(){
    int final[]= new int[5];
    for(int j=0; j<=4; j++){
        final[j]= Imarks[j]+ (Smarks[j])(2);
    }
    System.out.println("final marks");
    for(int i=0; i<=4; i++){
        System.out.println("subject"+i+":"+
                final[i]);
    }
}
```

```
import CIE.*;
import SEE.*;

class pmain{
    public static void main(String args[]){
        Student S1 = new Student();
        S1.getd();
        S1.disp();
        External e1 = new External();
        e1.getmi();
        e1.dispmi();
        e1.getm();
        e1.dispm();
        e1.finalcal();
    }
}
package CIE;
public class Student{
    public String usn, name;
    public int sem;
    public void getd(){
        System.out.println("enter usn, name,
                sem");
        usn = ss.nextline(); name = ss.nextline();
        sem = ss.nextInt();
    }
```

Output:


Enter USN, NAME & SEM
1BM21CS254
Z
1
Student Details:
USN:1BM21CS254
NAME:Z
SEM:1
Enter marks scored in 5 courses:
40
45
56
41
48
INTERNAL MARKS
Subject0=40
Subject1=45
Subject2=56
Subject3=41
Subject4=48
Enter external marks scored in 5 courses:


Command Prompt
45
56
41
48
INTERNAL MARKS
Subject0=40
Subject1=45
Subject2=56
Subject3=41
Subject4=48
Enter external marks scored in 5 courses:
43
45
47
48
41
EXTERNAL MARKS
Subject0=43
Subject1=45
Subject2=47
Subject3=48
Subject4=41
FINAL MARKS
Subject0=61
Subject1=67
Subject2=79
Subject3=65
Subject4=68