Name: **Vedant Tiwari**
Section: **CSE-A**
Roll no: **68**
Subject: **JAVA OOPS**

TA 1

Q1) Create the hierarchy as given below.
Provide the following functionality:
a) Calculate Age: Calculate the age using the predefined LocalDate Class
b) Calculate minimum and maximum age of the teacher
c) Calculate average age of the student
d) Calculate the salary of teacher as per the data given below

Qualification: PG – Base Salary: 60,000
PHD – Base Salary: 80,000
Salary:- Base Salary + experience*2000
Example:- Qualification: PG, Exp : 10 years
Salary = 60,000 + 10*2000= 80,000

e) Calculate Departmental Average – It is Average of CGPA for all semesters.
f) Calculate Semester Average: Average of CGPA for a given semester(Semester
is provided as parameter)

Person Class:

```java
public class Person {
   String name;
   String dob;
   public Person(String name, String dob) {
      this.name = name;
      this.dob = dob;
   }
}
```

Student Class:

```java
import java.time.LocalDate;
import java.time.Period;
public class Student extends Person{
    int rollNo;
    int semester;
    double CGPA;
    public Student(String name, String dob, int rollNo, int semester,
   double CGPA) {
        super(name, dob);
        this.rollNo = rollNo;
        this.semester = semester;
        this.CGPA = CGPA;
    }
    public int calculateAge(){
        LocalDate dateOfBirth = LocalDate.parse(dob);
        LocalDate currentDate = LocalDate.now();
        return Period.between(dateOfBirth,currentDate).getYears();
    }
   }
```

Department Class:

```
public class Department {
    String name;
    Student[] s;
    Teacher[] t;
    public Department(Student[] student, Teacher [] teacher, String name){
        this.name = name;
        s = student;
        t = teacher;
    }
    public double calculateAverageAge(){
        int sum = 0;
        double average;
        int l = s.length;
        for (Student student : s)
            sum += student.calculateAge();
        average = (double) sum /l;
        return average;
    }
    public int[] minMaxAge(){
        int[] ans = new int[2];
        ans[0] = t[0].calculateAge();
        ans[1] = t[0].calculateAge();
        for (Teacher teacher : t) {
            if (teacher.calculateAge() >= ans[1]) ans[1]
=teacher.calculateAge();
            if (teacher.calculateAge() <= ans[0]) ans[0]
=teacher.calculateAge();
        }
        return ans;
    }
    public double departmentalAverage(){
        double sum = 0;
        for(Student value : s)
            sum+=value.CGPA;
        return (sum/s.length);
    }
    public double semesterAverage(int n){
        double sum=0;
```

```
        int count = 0;
        for(Student s : s){
            if(s.semester==n) {
                sum+=s.CGPA;
                count++;
            }
        }
        return (sum/count);
    }
}
```

Teacher Class:

```
import java.time.LocalDate;
import java.time.Period;
public class Teacher extends Person {
 double salary;
 String qualifications;
 int experience;
 public Teacher(String name, String dob, String qualifications, int
experience) {
    super(name, dob);
    this.qualifications = qualifications;
    this.experience = experience;
 }
 public double calculateSalary() {
    double baseSalary = 0;
    if (qualifications.equals("PG")) baseSalary = 60000;
    else if (qualifications.equals("PHD")) baseSalary = 80000;
    salary = baseSalary + (experience * 2000L);
    return salary;
 }
 public int calculateAge() {
    LocalDate dateOfBirth = LocalDate.parse(dob);
    LocalDate currentDate = LocalDate.now();
    return Period.between(dateOfBirth, currentDate).getYears();
 }
}
```

Main Class:

```
public class Main {
    public static void main(String[] args) {
        Student[] students = new Student[3];
        Teacher[] teachers = new Teacher[3];
        students[0] = new Student("Vedant","2003-09-06",81,6,8.8);
        students[1] = new Student("Prathmesh","2003-01-16",57,23,9.8);
        students[2] = new Student("Arpit","2001-07-13",39,5,9.5);
        teachers[0] = new Teacher("KanakMam","1985-05-21","PG",7);
        teachers[1] = new Teacher("HirkaniMam","1998-01-25","PHD",12);
        teachers[2] = new Teacher("PratishrutiMam","1981-01-05","PG",7);

        System.out.println("Age of the students is: ");

        for(Student s : students){
            System.out.println(s.name+" - "+s.calculateAge());
        }

        System.out.println();

        Department Depart = new Department(students,teachers,"CSE");
        int[] a = Depart.minMaxAge();
        System.out.println("Minimum and maximum age of the teacher are
"+a[0]+" and "+a[1]+" respectively.");
        System.out.println();

        System.out.println("Average age of the student
:"+Depart.calculateAverageAge());

        System.out.println("Salaries of Teacher's are : ");
        for(Teacher t : teachers){
            System.out.println(t.name+" - "+t.calculateSalary());
        }
        System.out.println();

        System.out.printf("Departmental average :
%.3f",Depart.departmentalAverage());
        System.out.println();
```

```
        System.out.printf("Semester average :
%.3f",Depart.semesterAverage(3));
    }
   }
```

OUTPUT:

```
PS E:\RCOEM 4TH Semester\Proper java\TA\Question1>  & 'C:\Program Files\Java\jdk-20\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\051d02b8e6dcaee3a15070d4b489b6bb\redhat.java\jdt_ws\Question1_8727d491\bin' 'Main'
Age of the students is:
Vedant - 19
Prathmesh - 20
Arpit - 21

Minimum and maximum age of the teacher are 25 and 42 respectively.
Salaries of Teacher's are :
KanakMam - 78000.0
HirkaniMam - 108000.0
PratishrutiMam - 64000.0

Departmental average : 9.367
Semester average : 9.686
PS E:\RCOEM 4TH Semester\Proper java\TA\Question1>
```

Q2) The naive way of computing a value to the power n performs n multiplications.
A much better algorithm, involves repeated squaring:
Example : pow(a, 1)=a
pow(a, n) = pow(a*a, n/2) [n even]
= a * pow(a*a, (n-1)/2) [odd]
Even this is not always optimal.
For example, it computes a15 via the sequence a; a2; a3; a6; a7; a14; a15 in 6 steps
while it is possible to use a; a2; a4; a5; a10; a15 and manage in 5 steps.
Design a Java program that can be given any integer n and will find(by some formof exhaustive search) the smallest number of multiplications that could be used to
raise a value to the power n.

CODE:

Main.java code.java

Question2 > code.java > { } Question2

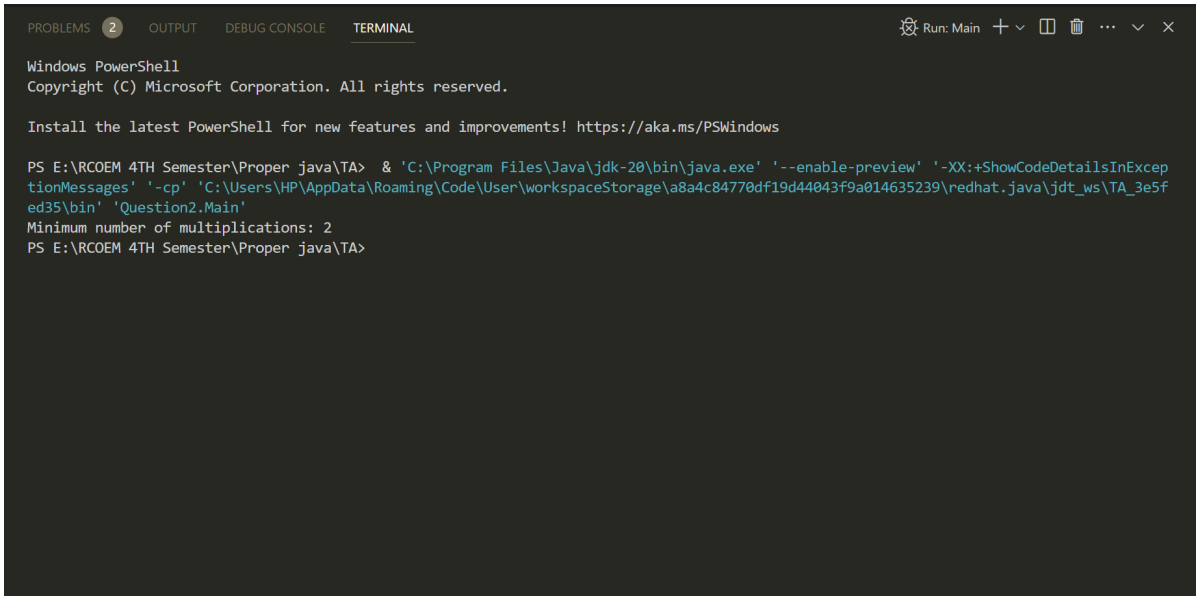
```java
package Question2;
class MinimumMultiplications {
    int findMinimumMultiplications(int a, int n) {
        if (n == 0) {
            return 0; // Base case: a^0 = 1 (no multiplications needed)
        }
        int minMultiplications = Integer.MAX_VALUE;
        for (int i = 1; i <= n; i++) {
            int multiplications;
            if (i % 2 == 0) {
                multiplications = 1 + findMinimumMultiplications(a * a, n / 2);
            }
            else {
                multiplications = 1 + findMinimumMultiplications(a * a, (n - 1) / 2);
            }
            if (multiplications < minMultiplications) {
                minMultiplications = multiplications;
            }
        }
        return minMultiplications;
    }
}

```

Main.java code.java

Question2 > Main.java > Main > main(String[])

```java
package Question2;

public class Main{

    public static void main(String[] args) {
        int a = 2;
        int n = 3;
        MinimumMultiplications x=new MinimumMultiplications();
        int minMultiplications = x.findMinimumMultiplications(a, n);
        System.out.println("Minimum number of multiplications: " + minMultiplications);
    }
}

```

OUTPUT:



Q3) The values in an array could be used to represent digits in the decimal representation of a number. For example, the number 1/7 has decimal representation 0.142857 and that

could be stored in an array whose elements started from {1,4,2,8,5,7}. For a number stored in this way we write code that multiplies the number by a given integer, returning the whole number part of the result and leaving the array updated to hold the fractional part of the product. Design proper class and return the result from the method.

C

CODE:

Main.java | Code.java

Question3 > Code.java > ...

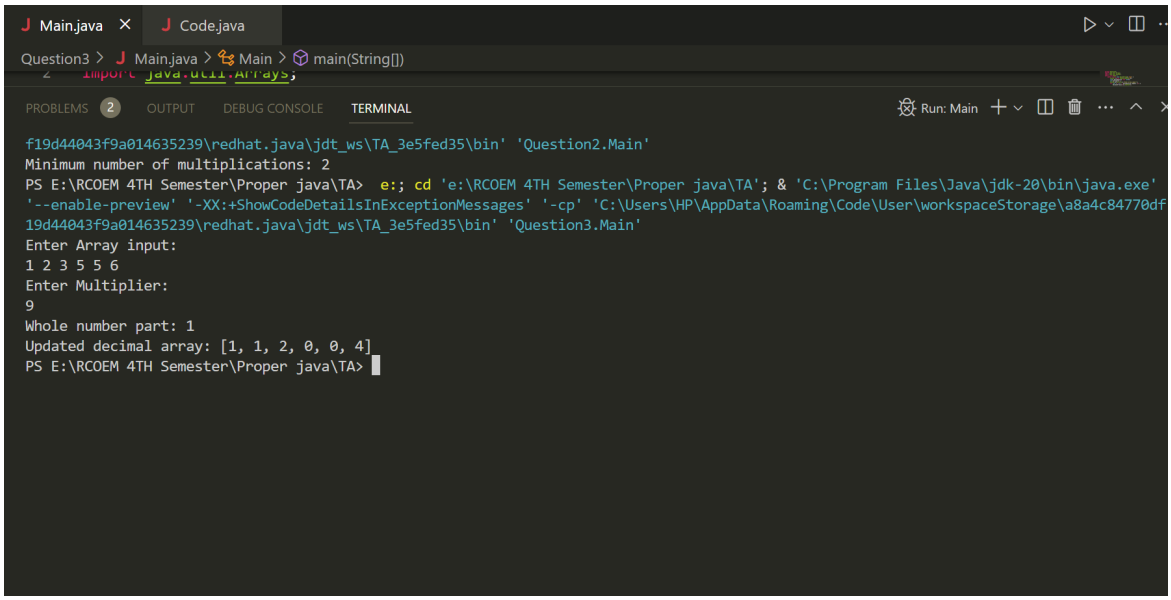
```java
package Question3;

class DecimalArrayMultiplier {
    public static int multiplyAndReturnWholeNumber(int[] decimalArray, int multiplier) {
        int carry = 0;

        for (int i = decimalArray.length - 1; i >= 0; i--) {
            int product = decimalArray[i] * multiplier + carry;
            decimalArray[i] = product % 10;
            carry = product / 10;
        }

        return carry; // The remaining carry represents the whole number part of the result
    }
}
```

Main.java | Code.java

Question3 > Main.java > ...

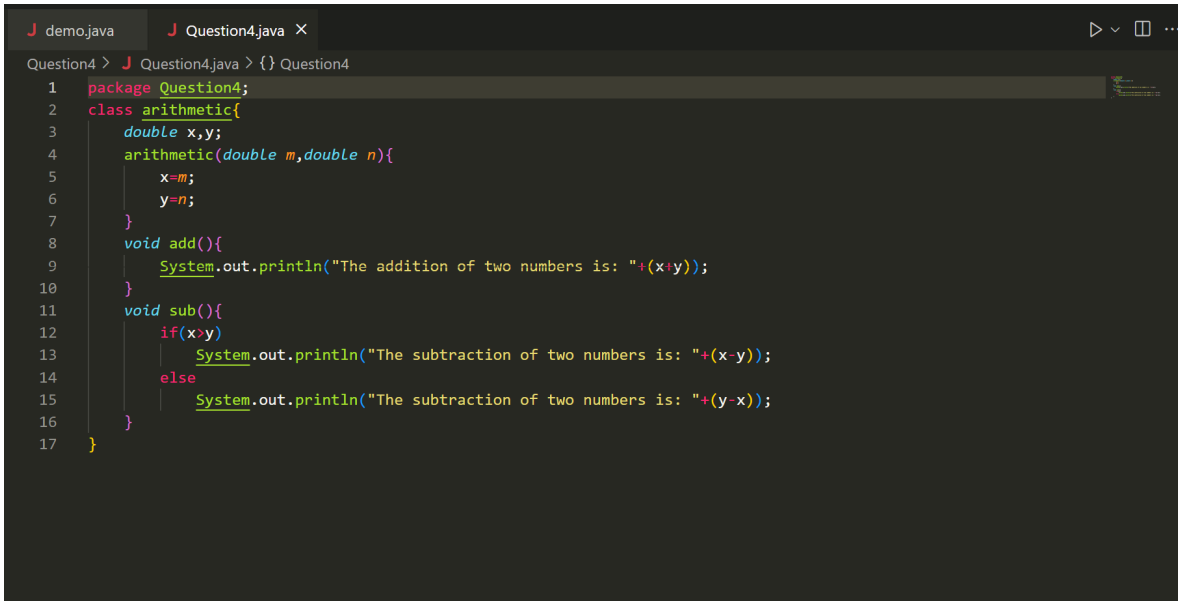
```java
package Question3;
import java.util.Arrays;
import java.util.Scanner;
public class Main {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        int[] decimalArray =new int[6];
        int multiplier=1;
        System.out.println(x:"Enter Array input:");
        for (int i = 0; i < decimalArray.length; i++) {
            decimalArray[i]=sc.nextInt();
        }
        System.out.println(x:"Enter Multiplier:");
        multiplier=sc.nextInt();
        int wholeNumberPart = DecimalArrayMultiplier.multiplyAndReturnWholeNumber(decimalArray, multiplier);

        System.out.println("Whole number part: " + wholeNumberPart);
        System.out.println("Updated decimal array: " + Arrays.toString(decimalArray));
        sc.close();
    }
}
```

OUTPUT:


Main.java
Code.java
Question3 > Main.java > Main > main(String[])
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Run: Main
f19d44043f9a014635239\redhat.java\jdt_ws\TA_3e5fed35\bin' 'Question2.Main'
Minimum number of multiplications: 2
PS E:\RCOEM 4TH Semester\Proper java\TA> e:; cd 'e:\RCOEM 4TH Semester\Proper java\TA'; & 'C:\Program Files\Java\jdk-20\bin\java.exe'
'--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\a8a4c84770df
19d44043f9a014635239\redhat.java\jdt_ws\TA_3e5fed35\bin' 'Question3.Main'
Enter Array input:
1 2 3 5 5 6
Enter Multiplier:
9
Whole number part: 1
Updated decimal array: [1, 1, 2, 0, 0, 4]
PS E:\RCOEM 4TH Semester\Proper java\TA>

Q4) Design a Package to contain class arithmetic with member function as add() and sub(). Create another class PackageDemo in another package which uses object of class arithmetic.

CODE:



demo.java | Question4.java

Question4 > Question4.java > { } Question4

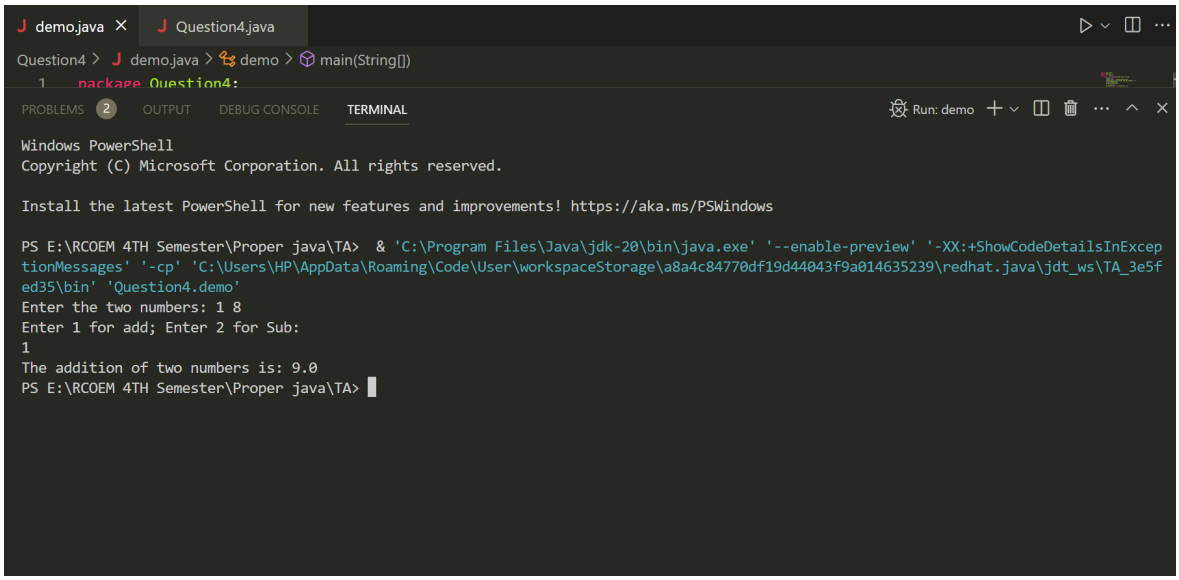
```
1  package Question4;
2  class arithmetic{
3      double x,y;
4      arithmetic(double m,double n){
5          x=m;
6          y=n;
7      }
8      void add(){
9          System.out.println("The addition of two numbers is: "+(x+y));
10     }
11     void sub(){
12         if(x>y)
13             System.out.println("The subtraction of two numbers is: "+(x-y));
14         else
15             System.out.println("The subtraction of two numbers is: "+(y-x));
16     }
17 }
```

demo.java | Question4.java

Question4 > demo.java > ...

```java
package Question4;
import java.util.*;
public class demo {
    Run | Debug
    public static void main(String[] args){
        double i,j;
        Scanner sc=new Scanner(System.in);
        System.out.print(s:"Enter the two numbers: ");
        i=sc.nextDouble();
        j=sc.nextDouble();
        // sc.close();
        arithmetic a=new arithmetic(i, j);
        int flag;
        System.out.println(x:"Enter 1 for add; Enter 2 for Sub: ");
        flag=sc.nextInt();
        sc.close();
        if(flag==1)
            a.add();
        else
            a.sub();
    }
}
```

OUTPUT:



Q5) Write a java program that accepts vehicle name and vehicle registration number
(ex MH-31 AZ-8207) and display the details on screen. Throw user defined
exception:
a. If no valid state is found(Valid states are MH, MP, AP, DL, GJ)
b. If registration number is not in the standard format (CC-99 CC-9999, C
stands for character and 9 stands for number)
CODE:

```
package Question5;
public class Registration_Number {
    String r;
    Registration_Number(String s){
        r=s;
    }
    void check(){
        String state=r.substring(0, 2);
        String code=r.substring(2, 5);
        String character=r.substring(6, 8);
        String number=r.substring(8);
        if(Check_states(state) &&
        Check_district(code) &&
        Check_character(character) &&
        Check_number(number)){
            System.out.println("Entered number is valid: "+r);
        }

    }

    boolean Check_states(String s){
        String[] states={"AP","DL","GJ","MH","MP" };
        for (int i = 0; i < states.length; i++) {
            if(states[i].equals(s)){
```

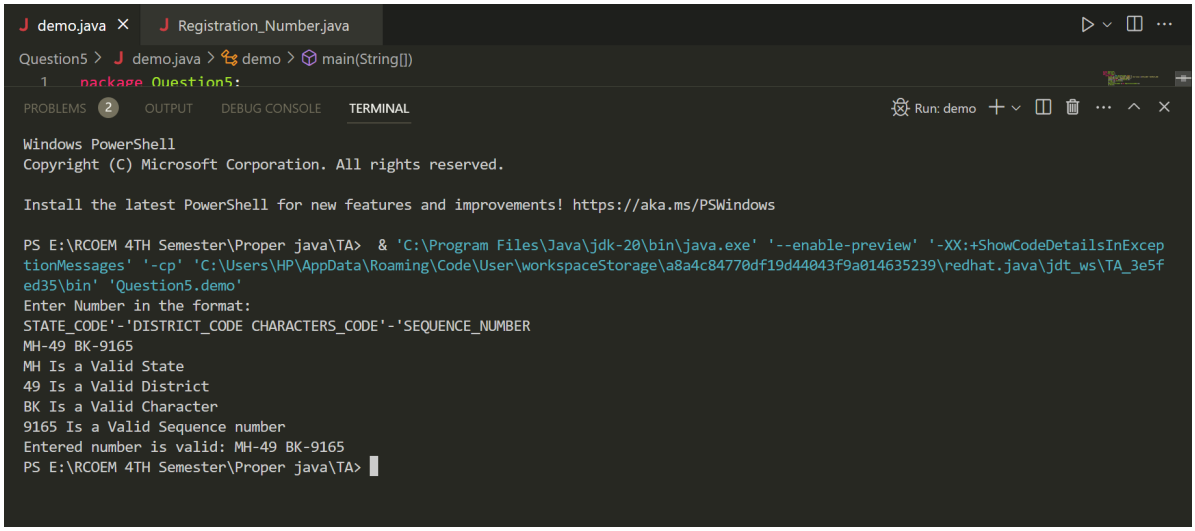
```
                System.out.println(r.substring(0, 2)+" Is a Valid State");
                return true;
            }
        }
        System.out.println("Invalid State code");
        return false;
    }
    boolean Check_district(String s){
        String[] codes={"-31","-49","-40","-09","-1" };
        for (int i = 0; i < codes.length; i++) {
            if(codes[i].equals(s)){
                System.out.println(r.substring(3, 5)+" Is a Valid
District");
                return true;
            }
        }
        System.out.println("Invalid District code");
        return false;
    }
    boolean Check_character(String s){
        char[] a=s.toCharArray();
        for (int i = 0; i < a.length; i++) {
            if(a[i]>=65 && a[i]<=90){
                System.out.println(r.substring(6, 8)+" Is a Valid
Character");
                return true;
            }
        }
        System.out.println("Invalid Character code");
        return false;
    }boolean Check_number(String s){
        char[] a=s.toCharArray();
        for (int i = 0; i < a.length; i++) {
             if(a[i]>=48 && a[i]<=57){
                System.out.println(r.substring(9)+" Is a Valid Sequence
number");
                return true;
            }
        }
        System.out.println("Invalid Sequence code");
```

```
        return false;
    }
}
```

MAIN:

```
package Question5;
import java.util.*;
public class demo {
    public static void main(String[] args) {
        System.out.println("Enter Number in the
format:\nSTATE_CODE'-'DISTRICT_CODE CHARACTERS_CODE'-'SEQUENCE_NUMBER");
        Scanner sc=new Scanner(System.in);
        String s=sc.nextLine();
        sc.close();
        Registration_Number a=new Registration_Number(s);
        a.check();
    }
}
```

OUTPUT:



Q6) Write a java program to implement multiple inheritance given below:

A. Create a class Time with the data members as hours and minutes.
Add Functionality to add and subtract 2 time objects.
Test the time class is main().
B. "GreatClock" (A scientific research company) wants advanced time objects which
will also provide functionalities of addition and subtraction of seconds and
milliseconds along with hours and minutes.
How will you add this feature without changing the Time Class?
C. Create a class TimeZone which will add functionality to convert the time from one
time zone to another time zone.
Note: Class Time zone uses the Time Object and uses the Add and Subtract methods
of Time Class.

CODE:

ADD & SUB:

```
package Question6;

public class timeTA {
    int h,m,s;
    int h1,m1,s1;
    timeTA(int h,int m,int s){
        this.h=h;
        this.m=m;
        this.s=s;
    }
    timeTA add(timeTA a,timeTA b){
```

```
        System.out.println("The addition of time is: ");
        timeTA c=new timeTA(0,0,0);
        c.h=a.h+b.h;
        if(c.h>=24){
            c.h=c.h%24;
        }
        c.m=a.m+b.m;
        if(c.m>=60){
            c.m=c.m%60;
            c.h++;
        }
        c.s=a.s+b.s;
        if(c.s>=60){
                c.s=c.s%60;
                c.m++;
        }
        display(c);
        return c;
    }
    timeTA sub(timeTA a,timeTA b){
        System.out.println("The Subtraction of time is: ");
        timeTA c=new timeTA(0,0,0);
        c.h=a.h-b.h;
        c.m=a.m-b.m;
        c.s=a.s-b.s;
        if(c.h<0){
            c.h=c.h*(-1);
        }
        if(c.m<0){
            c.m=c.m*(-1);
        }
        if(c.s<0){
            c.s=c.s*(-1);
        }
        if(c.h>=24){
            c.h=0;
        }
        if(c.m>=60){
            c.m=c.m%60;
            c.h++;
```

```
        }
        if(c.s>=60){
            c.s=c.s%60;
            c.m++;
        }
        display(c);
        return c;
    }
    void display(timeTA x){
        System.out.println("The Final Time is\n"+x.h+" : "+x.m+" : "+x.s);
    }
}
```

TIME CALCULATOR IN SECONDS AND MILLISECONDS:

```
package Question6;

public class timecalTA {
    int h,m,s,ms;
    int h1,m1,s1;
    timecalTA(int h,int m,int s,int ms){
        this.h=h;
        this.m=m;
        this.s=s;
        this.ms=ms;
    }
    timecalTA add(timecalTA a,timecalTA b){
        System.out.println("The addition of time calculator is: ");
        timecalTA c=new timecalTA(0,0,0,0);
        c.h=a.h+b.h;
        if(c.h>=24){
            c.h=c.h%24;
        }
        c.m=a.m+b.m;
        if(c.m>=60){
            c.m=c.m%60;
            c.h++;
        }
        c.s=a.s+b.s;
```

```
        if(c.s>=60){
                c.s=c.s%60;
                c.m++;
        }
        c.ms=a.ms+b.ms;
        if(c.ms>=60){
                c.ms=c.ms%60;
                c.s++;
        }
        display(c);
        return c;
    }
    timecalTA sub(timecalTA a,timecalTA b){
        System.out.println("The Subtraction of time calculator is: ");
        timecalTA c=new timecalTA(0,0,0,0);
        c.h=a.h-b.h;
        c.m=a.m-b.m;
        c.s=a.s-b.s;
        c.ms=a.ms-b.ms;

        if(c.h<0){
            c.h=c.h*(-1);
        }
        if(c.m<0){
            c.m=c.m*(-1);
        }
        if(c.s<0){
            c.s=c.s*(-1);
        }
        if(c.ms<0){
            c.ms=c.ms*(-1);
        }
        if(c.h>=24){
            c.h=0;
        }
        if(c.m>=60){
            c.m=c.m%60;
            c.h++;
        }
        if(c.s>=60){
```

```
            c.s=c.s%60;
            c.m++;
        }
        if(c.ms>=60){
            c.ms=c.ms%60;
            c.s++;
        }
        display(c);
        return c;
    }

    void display(timecalTA x){
        System.out.println("The Final Time calculator is\n"+x.h+" :
"+x.m+" : "+x.s+" : "+x.ms);
    }
}
```

TIMEZONE:

```
package Question6;

// import java.sql.TimeTA;
import java.util.*;

public class timezone {
    timeTA a;
    timezone(timeTA a){
        this.a=a;
    }
    void timeTAc(timeTA a){
        Scanner sc=new Scanner(System.in);
        System.out.print("Is time +/-: ");
        String p=sc.next();
        char q[]=p.toCharArray();
        if(q[0]=='+'){
            System.out.print("Enter the hours ahead from given time: ");
            int h=sc.nextInt();
            System.out.print("Enter the Minutes ahead from given time: ");
            int m=sc.nextInt();
```

```
            System.out.print("Enter the Seconds ahead from given time: ");
            int s=sc.nextInt();
            timeTA c=new timeTA(h, m, s);
            a.add(a, c);
        }
        else if(q[0]=='-'){
            System.out.print("Enter the hours behind from given time: ");
            int h=sc.nextInt();
            System.out.print("Enter the Minutes behind from given time:
");
            int m=sc.nextInt();
            System.out.print("Enter the Seconds behind from given time:
");
            int s=sc.nextInt();
            timeTA c=new timeTA(h, m, s);
            a.sub(a, c);
        }
        sc.close();
    }
}
```

MAIN:

```
package Question6;


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


        System.out.println("Part one of Question Giving simple addition
and substraction of HOURS MINUTES SECOND");
        timeTA x=new timeTA(12, 42, 55);
        timeTA y=new timeTA(2, 4, 5);
        x.add(x, y);
        y.sub(x, y);
        System.out.println("\n\n");
```

```
        System.out.println("Part two of Question Giving simple addition
and substraction For Scientific Calculator of HOURS MINUTES SECOND
MILLISECOND");
        timecalTA a=new timecalTA(10, 20, 12,11);
        timecalTA b=new timecalTA(2,4,12,32);
        timecalTA c=new timecalTA(0, 0, 0, 0);
        c.add(a, b);
        c.sub(a, b);
        System.out.println("\n\n");

        System.out.println("Part Three of Question Giving simple Timezone
of the countries: ");
        timezone v=new timezone(x);
        v.timeTAc(x);

    }
}
```

RESULT:

```
PS E:\RCOEM 4TH Semester\Proper java\TA>  & 'C:\Program Files\Java\jdk-20\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExcep
tionMessages' '-cp' 'C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\a8a4c84770df19d44043f9a014635239\redhat.java\jdt_ws\TA_3e5f
ed35\bin' 'Question6.Main'
Part one of Question Giving simple addition and substraction of HOURS MINUTES SECOND
The addition of time is:
10 : 38 : 50


Part two of Question Giving simple addition and substraction For Scientific Calculator of HOURS MINUTES SECOND MILLISECOND
The addition of time calculator is:
The Final Time calculator is
12 : 24 : 24 : 43
The Subtraction of time calculator is:
The Final Time calculator is
8 : 16 : 0 : 21


Part Three of Question Giving simple Timezone of the countries:
Is time +/-: +
Enter the hours ahead from given time: 1
Enter the Minutes ahead from given time: 4
Enter the Seconds ahead from given time: 9
The addition of time is:
The Final Time is
13 : 47 : 4
```

Q7) Write a java program to implement multiple inheritance given below:

DIRECTOR CLASS:

```
public class director {
    String name;
    director() {}
    protected director(String name) {
        this.name = name;
    }
    public void details(String name) {
        System.out.println("Name of Director : " + name);
    }
}
```

EMPLOYEE CLASS:

```
public class employee extends Manager implements tempemp, permanentemp {
    employee() {}
    public void details(String name) {
        System.out.println("Name of Employee : " + name);
    }
    public void clientDealt(int numClients) {
        System.out.println("total number of clients dealt : " +
numClients);
    }
    public void clientTillNow(int numClients) {
        System.out.println("total number of clients till now : " +
numClients);
    }
}
```

INTERFACES CLASS:

```
interface permanentemp extends permanentManager {
    public void clientTillNow(int numClients);
}
```

```
interface tempemp extends Consultant {
    public void salaryIncr(double salary);
    public void clientDealt(int numClients);
}
interface permanentManager {
    public void totalExp(int years);
}
interface Consultant extends ConsultantManager {
    public void clientDealt(int numOfClients);
}
interface ConsultantManager {
    public void salaryIncr(double salary);
}
```

MANAGER CLASS:

```
public class Manager extends director implements ConsultantManager,
permanentManager {
    Manager() {}
    public void details(String name) {
        System.out.println("Name of Manager : " + name);
    }
    public void salaryIncr(double salary) {
        salary += 0.5 * salary;
        System.out.println("Your incremented salary is " + salary);
    }
    public void totalExp(int years) {
        System.out.println("Experience : " + years);
    }
}
```

MAIN CLASS:

```
public class Main {
    public static void main(String[] args) {
        director d1 = new director();
        d1.details("Ram");
```

```java
        Manager m1 = new Manager();
        m1.details("Lakhan");
        m1.salaryIncr(200000);
        m1.totalExp(5);

        employee e1 = new employee();
        e1.details("Bharat");
        e1.clientDealt(50);
        e1.clientTillNow(200);
    }
}
```

OUTPUT :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS E:\RCOEM 4TH Semester\Proper java\TA>  & 'C:\Program Files\Java\jdk-20\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExcep
tionMessages' '-cp' 'C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\a8a4c84770df19d44043f9a014635239\redhat.java\jdt_ws\TA_3e5f
ed35\bin' 'App'
Name of Director : Ram
Name of Manager : Lakhan
Your incremented salary is 300000.0
Experience : 5
Name of Employee : Bharat
total number of clients dealt : 50
total number of clients till now : 200
PS E:\RCOEM 4TH Semester\Proper java\TA>
```

RESULT:

Successful execution of TA.