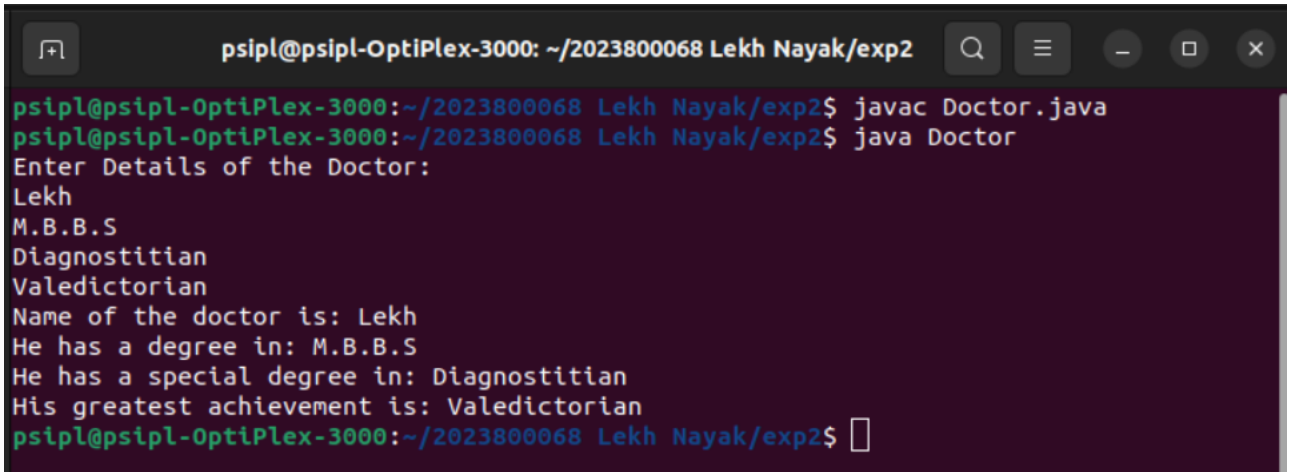
Name:	**Lekh Sanatan Nayak**
UID:	**2023800068**
Experiment No.	**2**

AIM:	***Write a program to demonstrate constructor***
Program 1	
PROBLEM STATEMENT :	Create a class doctor with name,degree,spdegree and acheivement. The class contains a display function to display the values. A parameterized constructor should be included in your program. Use user given input for the same.
PROGRAM:	import java.util.*; **// Creating a class named Doctor** class Doctor { **// Declaring instance variables to store doctor details** String name; **// Doctor's name** String degree; **// Degree obtained by the doctor** String spdegree; **// Special degree obtained by the doctor** String achievement; **// Doctor's greatest achievement** **// Parameterized constructor for Doctor class** Doctor(String name, String degree, String spdegree, String achievement) { **// Initializing instance variables with provided values** this.name = name; this.degree = degree; this.spdegree = spdegree; this.achievement = achievement; } **// Method to display doctor's details** void display() { System.out.println("Name of the doctor is: " + name); System.out.println("He has a degree in: " + degree);

```
        System.out.println("He has a special degree in: " + spdegree);
        System.out.println("His greatest achievement is: " + achievement);
    }

    // Main method
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Prompting the user to enter details of the doctor
        System.out.println("Enter Details of the Doctor:");
        String name = sc.next(); // Reading doctor's name
        String degree = sc.next(); // Reading doctor's degree
        String spdegree = sc.next(); // Reading doctor's special degree
        String achievement = sc.next(); // Reading doctor's greatest
achievement

        // Creating a Doctor object with the provided details
        Doctor doc = new Doctor(name, degree, spdegree, achievement);

        // Displaying the details of the doctor
        doc.display();

        sc.close(); // Closing the scanner object
    }
}
```

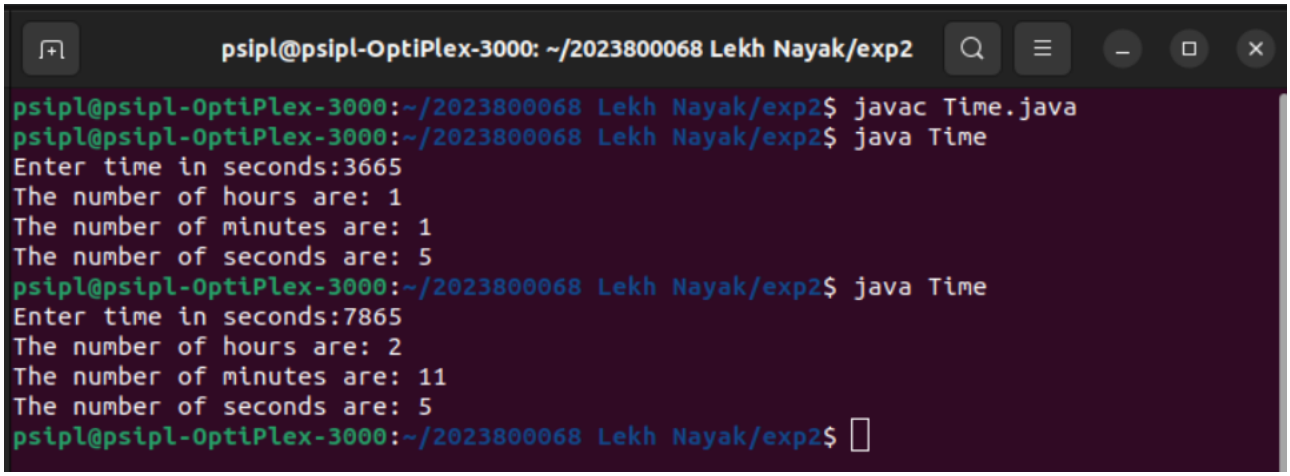
RESULT:



Program 2

PROBLEM STATEMENT :	**Time class** : Hours, Minutes and Seconds a)Write a program that asks the user to enter seconds as integer. The program should compute and display the number of hours, number of minutes and number of seconds in that seconds. For example if user enters 4205 seconds as a input called totSeconds. The answer should be Hours : 1 Minutes : 10 Seconds : 5 These calculations can be done in a method name: conversion1() which will also display the result as Hour,minutes and seconds. Similarly- if user input is Hour, Minute and Second then write another method conversion2() which will have these three parameters and the conversion2 method will compute and display the totSeconds. Note - Use any class (ie scanner /BufferedReader class) for input through console as hour/min/seconds/totSeconds
PROGRAM:	import java.util.*; // **Creating the class Time** class Time { int h, m, s; // **Variables to store hours, minutes, and seconds** // **Constructor for Time class with one parameter representing seconds** Time(int s) { this.s = s; // **Initialize seconds** } // **Method to convert seconds into hours, minutes, and remaining**

	seconds void conversion() { h = s / 3600; **// Calculate hours** m = (s % 3600) / 60; **// Calculate minutes** s = (s % 3600) % 60; **// Calculate remaining seconds** } **// Method to display the converted time** void display() { System.out.println("The number of hours are: " + h); System.out.println("The number of minutes are: " + m); System.out.println("The number of seconds are: " + s); } // Main method public static void main(String[] args) { Scanner sc = new Scanner(System.in); System.out.print("Enter time in seconds: "); **// Prompt the user to enter time in seconds** int seconds = sc.nextInt(); **// Read the input from the user** Time t = new Time(seconds); **// Create a Time object with the provided seconds** t.conversion(); **// Convert seconds to hours, minutes, and seconds** t.display(); **// Display the converted time** sc.close(); **// Close the scanner object** } }
RESULT:	

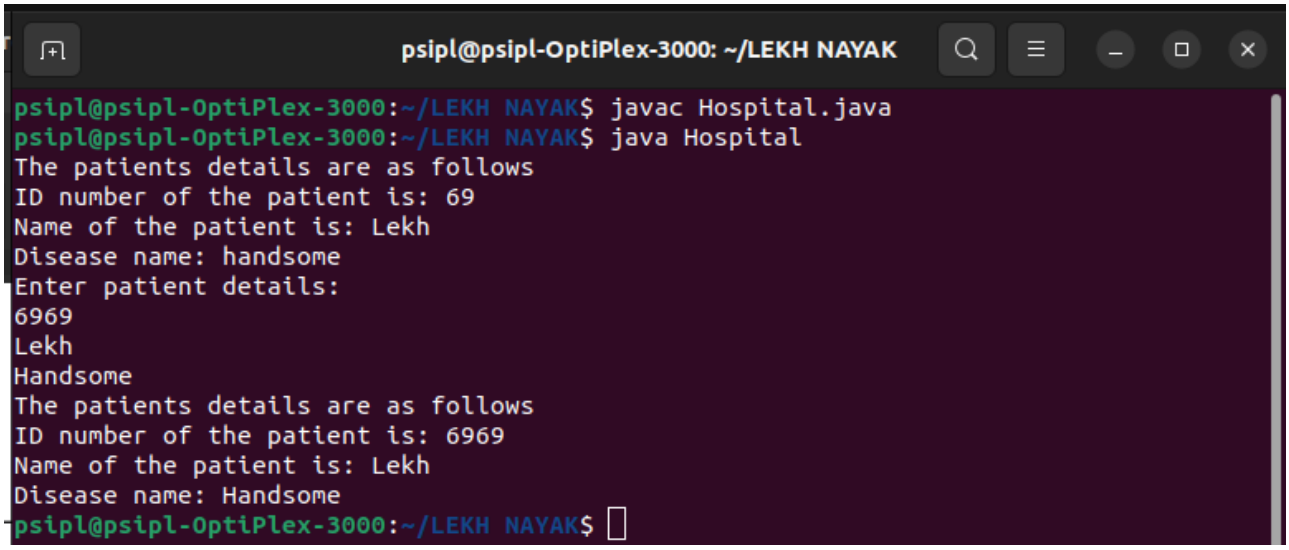


Program 3	
PROBLEM STATEMENT:	Create a class Hospital, assume appropriete data members. Create a default constructor and provide appropriete get and set methods for the data members.
PROGRAM:	import java.util.*; **// Defining a class named Hospital** public class Hospital { // Declaring instance variables to store patient details int pid; // **Patient ID** String pname; **// Patient name** String pdisease; **// Patient disease** **// Setter method for patient ID** public void setpid(int id) { pid = id; } **// Setter method for patient name** public void setpname(String name) { pname = name; } **// Setter method for patient disease** public void setpdisease(String disease) { pdisease = disease;

```
    }

    // Getter method for patient ID
    public int getpid() {
        return pid;
    }

    // Getter method for patient name
    public String getpname() {
        return pname;
    }

    // Getter method for patient disease
    public String getpdisease() {
        return pdisease;
    }

    // Default constructor for Hospital class
    Hospital() {
        // Initializing default values for patient details
        pid = 69;
        pname = "Lekh";
        pdisease = "handsome";
    }

    // Method to display patient details
    public void display() {
        System.out.println("The patients details are as follows");
        System.out.println("ID number of the patient is: " + pid);
        System.out.println("Name of the patient is: " + pname);
        System.out.println("Disease name: " + pdisease);
    }

    // Main method
    public static void main(String[] args) {
        // Creating a Hospital object with default values and displaying details
        Hospital obj1 = new Hospital();
        obj1.display();
```

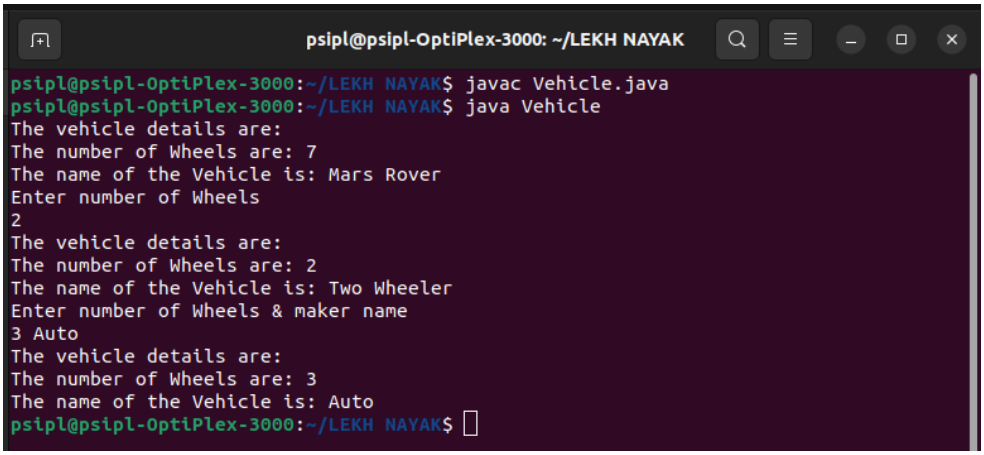
```
        // Creating a Scanner object to take user input
        Scanner sc = new Scanner(System.in);

        // Asking user to enter patient details
        System.out.println("Enter patient details:");

        // Taking patient ID input from user
        int id = sc.nextInt();
        obj1.setpid(id); // Setting patient ID
        obj1.getpid(); // Getting patient ID

        // Taking patient name input from user
        String name = sc.next();
        obj1.setpname(name); // Setting patient name
        obj1.getpname(); // Getting patient name

        // Taking patient disease input from user
        String disease = sc.next();
        obj1.setpdisease(disease); // Setting patient disease
        obj1.getpdisease(); // Getting patient disease

        // Displaying updated patient details
        obj1.display();

        sc.close(); // Closing the scanner object
    }
}
```

RESULT:



Program 4	
PROBLEM STATEMENT:	Create a class Vehicle. Assume appropriete data members. Demonstrate default constructor and parameterised constructor for the data members.
PROGRAM:	import java.util.*; **// Defining a class named Vehicle** class Vehicle { int noWheels**; //** **Variable to store the number of wheels** String make; **//** **Variable to store the make of the vehicle** **// Default constructor for Vehicle class** Vehicle() { noWheels = 7**; //** **Default number of wheels** make = "Mars Rover"; **//** **Default make** } **// Parameterized constructor with one parameter** Vehicle(int num) { noWheels = num; **// Set make based on the number of wheels** if (num == 2) { make = "Two Wheeler"; } else { make = "Not found"; **//** **Default make if number of wheels is not**

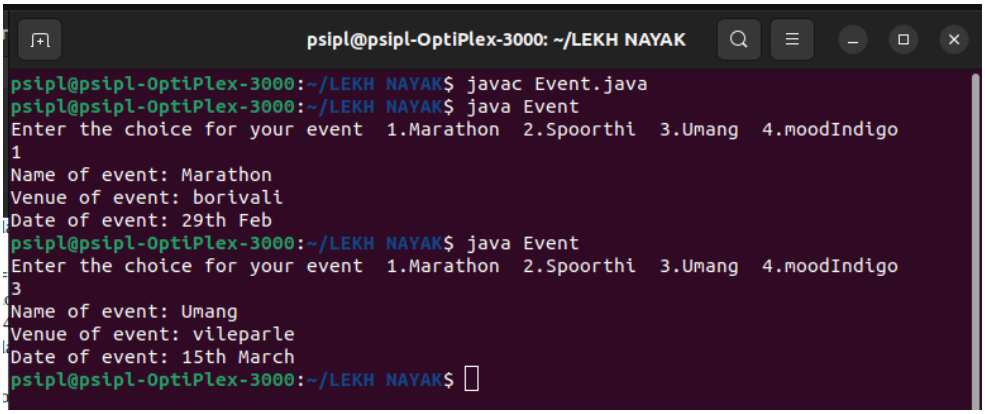
```
recognized
        }
    }

    // Parameterized constructor with two parameters
    Vehicle(int num, String name) {
        noWheels = num;
        // Set make based on the number of wheels
        if (num == 3) {
            make = name; // Custom make provided by the user
        } else if (num == 4) {
            name = "F1 Car"; // Default make for 4-wheeled vehicles
        } else {
            make = "Not found"; // Default make if number of wheels is not
recognized
        }
    }

    // Method to display vehicle details
    public void display() {
        System.out.println("The vehicle details are: ");
        System.out.println("The number of Wheels are: " + noWheels);
        System.out.println("The name of the Vehicle is: " + make);
    }

    // Main method
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Creating a default Vehicle object and displaying its details
        Vehicle obj1 = new Vehicle();
        obj1.display();

        // Asking user input for number of wheels and creating Vehicle
object accordingly
        System.out.println("Enter number of Wheels");
        int num1 = sc.nextInt();
        Vehicle obj2 = new Vehicle(num1);
        obj2.display();
```

	// **Asking user input for number of wheels and maker name, then creating Vehicle object** System.out.println("Enter number of Wheels & maker name "); int num2 = sc.nextInt(); String name = sc.next(); Vehicle obj3 = new Vehicle(num2, name); obj3.display(); sc.close(); // **Closing the scanner object** } }
RESULT:	
Program 5	
PROBLEM STATEMENT:	Create a class Event. Assume appropriete data members. Demonstrate menu driven program to invoke default constructor, parameterised constructor and copy constructor for the data members.
PROGRAM:	import java.util.*; // **Define a class named Event** class Event { String ename; // **Event name** String evenue; // **Event venue** String edate; // **Event date**

```
    // Default constructor initializes some default values
    Event() {
        ename = "marathon";
        evenue = "borivali";
        edate = "29th Feb";
    }

    // Constructor with a parameter to set event details based on the provided name
    Event(String name) {
        this.ename = name;
        // Set event details based on the provided name
        if (name.equals("Marathon")) {
            evenue = "borivali";
            edate = "29th Feb";
        } else if (name.equals("spoorthi")) {
            evenue = "andheri";
            edate = "15th Feb";
        } else if (name.equals("Umang")) {
            evenue = "vileparle";
            edate = "15th March";
        } else if (name.equals("moodIndigo")) {
            evenue = "powai";
            edate = "15th Dec";
        }
    }

    // Method to display event details
    void display() {
        System.out.println("Name of event: " + this.ename);
        System.out.println("Venue of event: " + evenue);
        System.out.println("Date of event: " + edate);
    }

    // Main method
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String name;
        System.out.println("Enter the choice for your event  1.Marathon 2.Spoorthi  3.Umang  4.moodIndigo");
```

```
        int choice;
        choice = sc.nextInt();
        // Check user's choice and create Event object accordingly
        if (choice == 1) {
            name = "Marathon";
            Event obj1 = new Event(name);
            obj1.display();
        } else if (choice == 2) {
            name = "spoorthi";
            Event obj2 = new Event(name);
            obj2.display();
        }
        if (choice == 3) {
            name = "Umang";
            Event obj3 = new Event(name);
            obj3.display();
        }
        if (choice == 4) {
            name = "moodIndigo";
            Event obj4 = new Event(name);
            obj4.display();
        } else if (choice > 4) {
            System.out.println("No information");
        }
    }
}
```

RESULT:



CONCLUSION:

In this experiment i learnt the use of defalut & parameterized constructor in in problem solving coding