TEST-5

1.Create a student attendance system to record and manage student attendance. Implement methods to mark attendance, generate attendance reports, and calculate attendance percentages.

Methods:

• markAttendance(int studentId, String date, boolean isPresent)

• generateAttendanceReport(int studentId)

• calculateAttendancePercentage(int studentId)

**PROGRAM**

import java.util.Scanner;

import java.util.ArrayList;

class attendance

{

public static class Student

{

int total=0;

int n,Reg\_No;

Scanner scanner=new Scanner(System.in);

ArrayList<Integer> arr=new ArrayList<>();

ArrayList<Integer> arr1=new ArrayList<>();

public Student()

{

System.out.print("Enter The Total Number Of Students:-");

n=scanner.nextInt();

scanner.nextLine();

for(int i=0;i<n;i++)

{

arr1.add(0);

}

for(int i=0;i<n;i++)

{

System.out.print("Enter The Student RegNo:-");

Reg\_No=scanner.nextInt();

scanner.nextLine();

arr.add(Reg\_No);

}

}

public void markattendance(int reg\_no,boolean ispresent)

{

for(int i=0;i<arr.size();i++)

{

if(arr.get(i)==reg\_no)

{

total+=1;

if(ispresent==true)

{

arr1.set(i,arr1.get(i)+1);

}

}

}

}

public void getpercentage(int reg\_no)

{

for(int i=0;i<arr.size();i++)

{

if(reg\_no==arr.get(i))

{

int percentage=(arr1.get(i)/total)\*100;

System.out.println("The Attendance Percentage For The Student IS:-"+percentage+"%");

}

}

}

}

public static void main(String[] args)

{

int no;

Scanner scanner=new Scanner(System.in);

Student std=new Student();

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

std.markattendance(192311133,true);

System.out.print("Enter The Student Register Number To Get The Attendance Report:-");

no=scanner.nextInt();

std.getpercentage(no);

}

}

2. Develop a weather forecast application that fetches and displays weather information. Implement methods to get current weather, forecast for the week, and display weather details.

Methods:

• getCurrentWeather(String location)

• getWeeklyForecast(String location)

• displayWeatherDetails(String location)

**PROGRAM**

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.net.HttpURLConnection;

import java.net.URL;

public class WeatherApp {

private static final String API\_KEY = "YOUR\_API\_KEY";

private static final String BASE\_URL = "(link unavailable)";

public static void main(String[] args) {

WeatherApp app = new WeatherApp();

app.displayWeatherDetails("London");

}

public String getCurrentWeather(String location) {

try {

URL url = new URL(BASE\_URL + "weather?q=" + location + "&appid=" + API\_KEY);

HttpURLConnection connection = (HttpURLConnection) url.openConnection();

connection.setRequestMethod("GET");

int responseCode = connection.getResponseCode();

if (responseCode == 200) {

return readResponse(connection.getInputStream());

} else {

return "Failed to retrieve data";

}

} catch (Exception e) {

return "Error: " + e.getMessage();

}

}

public String getWeeklyForecast(String location) {

try {

URL url = new URL(BASE\_URL + "forecast?q=" + location + "&appid=" + API\_KEY);

HttpURLConnection connection = (HttpURLConnection) url.openConnection();

connection.setRequestMethod("GET");

int responseCode = connection.getResponseCode();

if (responseCode == 200) {

return readResponse(connection.getInputStream());

} else {

return "Failed to retrieve data";

}

} catch (Exception e) {

return "Error: " + e.getMessage();

}

}

private String readResponse(InputStream inputStream) {

try (BufferedReader reader = new BufferedReader(new InputStreamReader(inputStream))) {

String inputLine;

StringBuffer response = new StringBuffer();

while ((inputLine = reader.readLine()) != null) {

response.append(inputLine);

}

return response.toString();

} catch (Exception e) {

return "Error: " + e.getMessage();

}

}

public void displayWeatherDetails(String location) {

String currentWeather = getCurrentWeather(location);

String weeklyForecast = getWeeklyForecast(location);

JSONObject jsonObject = new JSONObject(currentWeather);

System.out.println("Current Weather in " + location);

System.out.println("Temperature: " + jsonObject.getJSONObject("main").getDouble("temp"));

System.out.println("Humidity: " + jsonObject.getJSONObject("main").getDouble("humidity"));

System.out.println("Weather: " + jsonObject.getJSONArray("weather").getJSONObject(0).getString("description"));

System.out.println("Weekly Forecast in " + location);

System.out.println(weeklyForecast);

}

}