Experiment 4

1. Explore the features of ES6 like arrow functions, callbacks, promises, async/await.
2. Implement an application for reading the weather information from openweathermap.org and display the information in the form of a graph on the web page.

**Aim:**

To explore the features of ES6, such as arrow functions, callbacks, promises, and to fetches weather data from openweathermap.org

**Objective:**

1. To understand ES6 features like arrow functions, callbacks, promises, and async/await.
2. To display the fetched weather information in the form of a graph.

**Procedure:**

Step 1:  
Create a folder and open in VS code.

Step 2:

Create a index.html and script.js files.

Step 3:

Link the chart.js file.

Step 4:

Write all the relevant codes and click on live server or open it in browser by copying path.

Step 5:

Open the console page by clicking on inspect for errors.

**Code Snippet:**

**index.html:**

<h1>Weather Report</h1>

    <form >

        <input type="text"  id="city"> <br>

        <button type="onsubmit" onclick="weatherForm(event)" >Get weather information</button>

    </form>

    <canvas id="myChart"></canvas>

    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

    <script src="script.js"></script>

**script.js:**

async function weatherForm(event) {

    event.preventDefault()

    let city = document.getElementById("city").value

    let data = await weatherData(city)

    console.log(data)

    let [labels,temps]=weatherParameters(data)

    renderChart(labels,temps)

}

async function weatherData(city) {

    let apiKey = "00b934cea6db6174aa19a3ff5f93eeb2"

    let apiCall = `https://api.openweathermap.org/data/2.5/forecast?q=${city}&appid=${apiKey}&units=metric`

    try {

        let result = await fetch(apiCall)

        let data = await result.json()

        if (data.cod=="200"){

            return data

        }

        else {

            alert("city not found")

            return null

        }

    }

    catch (error) {

        console.error("city not found",error)

        alert("error while fetching")

        return null

    }

}

function weatherParameters(data) {

    let labels = data.list.map((item)=> new Date(item.dt\_txt).toLocaleTimeString())

    let temps = data.list.map((item)=> item.main.temp)

    console.log(temps)

    console.log(labels)

    return [labels,temps]

}

function renderChart(labels,temps) {

    const ctx = document.getElementById('myChart');

  new Chart(ctx, {

    type: 'line',

    data: {

      labels: labels,

      datasets: [{

        label: 'temp',

        data: temps,

        borderWidth: 1

      }]

    },

    options: {

      scales: {

        y: {

          beginAtZero: false

        }

      }

    }

  });

}

**Conclusion:**

Successfully explored various ES6 features like arrow functions, callbacks, promises, and async/await to handle asynchronous tasks effectively and fetched weather data from openweathermap.org using chart.js.