**WEEK-4**

4.Explore the features of ES6 like arrow functions, call backs, promises, async/await. 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: 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

DESCRIPTION:HTML, CSS, and JavaScript are essential components for creating a functional responsive web application.

Project Structure:1. index.html - Main HTML file containing the structure of the web application.

2. main.css

3. main.js

4.bg.jpg

**Index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Weather App</title>

<link rel="stylesheet" href="main.css" />

</head>

<body>

<div class="app-wrap">

<header>

<input type="text" autocomplete="off" class="search-box" placeholder="Search for a city..." />

</header>

<main>

<section class="location">

<div class="city">New York, US</div>

<div class="date">Wednesday 22 July 2020</div>

</section>

<div class="current">

<div class="temp">15<span>°c</span></div>

<div class="weather">Sunny</div>

<div class="hi-low">13°c / 16°c</div>

</div>

</main>

</div>

<script src="main.js"></script>

</body>

</html>

**main.css**

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'montserrat', sans-serif;

background-image: url('bg.jpg');

background-size: cover;

background-position: top center;

}

.app-wrap {

display: flex;

flex-direction: column;

min-height: 100vh;

background-image: linear-gradient(to bottom, rgba(0, 0, 0, 0.3), rgba(0, 0, 0, 0.3));

}

header {

display: flex;

justify-content: center;

align-items: center;

padding: 50px 15px 15px;

}

header input {

width: 100%;

max-width: 280px;

padding: 10px 15px;

border: none;

outline: none;

background-color: rgba(255, 255, 255, 0.3);

border-radius: 0px 16px 0px 16px;

border-bottom: 3px solid gray;

color: #313131;

font-size: 20px;

font-weight: 300;

transition: 0.2s ease-out;

}

header input:focus {

background-color: rgba(255, 255, 255, 0.6);

}

main {

flex: 1 1 100%;

padding: 25px 25px 50px;

display: flex;

flex-direction: column;

align-items: center;

text-align: center;

}

.location .city {

color: #fff;

font-size: 32px;

font-weight: 500;

margin-bottom: 5px;

}

.location .date {

color: #fff;

font-size: 16px;

}

.current .temp {

color: #fff;

font-size: 102px;

font-weight: 900;

margin: 30px 0px;

text-shadow: 2px 10px rgba(0, 0, 0, 0.6);

}

.current .temp span {

font-weight: 500;

}

.current .weather {

color: #fff;

font-size: 32px;

font-weight: 700;

font-style: italic;

margin-bottom: 15px;

text-shadow: 0px 3px rgba(0, 0, 0, 0.4);

}

.current .hi-low {

color: #fff;

font-size: 24px;

font-weight: 500;

text-shadow: 0px 4px rgba(0, 0, 0, 0.4);

}

**main.js**

const api = {

key: "fcc8de7015bbb202209bbf0261babf4c",

base: "https://api.openweathermap.org/data/2.5/"

}

const searchbox = document.querySelector('.search-box');

searchbox.addEventListener('keypress', setQuery);

function setQuery(evt) {

if (evt.keyCode == 13) {

getResults(searchbox.value);

}

}

function getResults (query) {

fetch(`${api.base}weather?q=${query}&units=metric&APPID=${api.key}`)

.then(weather => {

return weather.json();

}).then(displayResults);

}

function displayResults (weather) {

let city = document.querySelector('.location .city');

city.innerText = `${weather.name}, ${weather.sys.country}`;

let now = new Date();

let date = document.querySelector('.location .date');

date.innerText = dateBuilder(now);

let temp = document.querySelector('.current .temp');

temp.innerHTML = `${Math.round(weather.main.temp)}<span>°c</span>`;

let weather\_el = document.querySelector('.current .weather');

weather\_el.innerText = weather.weather[0].main;

let hilow = document.querySelector('.hi-low');

hilow.innerText = `${Math.round(weather.main.temp\_min)}°c / ${Math.round(weather.main.temp\_max)}°c`;

}

function dateBuilder (d) {

let months = ["January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"];

let days = ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"];

let day = days[d.getDay()];

let date = d.getDate();

let month = months[d.getMonth()];

let year = d.getFullYear();

return `${day} ${date} ${month} ${year}`;

}