WD-MAJOR-SEPTEMBER

HTML CODE:

```
<!DOCTYPE html>
<html>

<head>
    <meta charset="utf-8">
    <title>GoT A PLAN</title>
      <link rel="shortcut icon" href="images/weatherlogo.jpeg">
    <link rel="stylesheet" type="text/css" href="src/reset.css" />
    <link rel="stylesheet" type="text/css" href="STYLE[1].css" />
    <link href="https://fonts.googleapis.com/css?family=Montserrat"
rel="stylesheet">
    <link
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro|Work+Sans"
rel="stylesheet">
  <link
href="https://fonts.googleapis.com/css2?family=Varela+Round&display=swap"rel="
stylesheet">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></scrip
t>
</head>

<body>

    <!-- Main Section -->
    <main class="main">
      <h1 class="q"></h1>
      <h1 id="">GOT A PLAN</h1>
      <form autocomplete="off">
            <a href="https://owm-inc.github.io/VANE-intro/apps/leaflet-
owm.html"><img src="images/weather.png" id="s"></a>
      <input type="text" id="city">
        <button id="button" type="submit">Submit</button>
      </form>
      <h1 class="q1"></h1>
    </main>
    <div class="container">
        <div id="destination">
        </div>
        <h2 class="e">Pollution</h2>
  <form onsubmit="event.preventDefault(); handleSubmit();">
```

```
  <!--   <label for="cityInput">City:</label>
  <input type="text" id="cityInput">     -->
  <section  id="cityInput"></section>
  <button type="submit" id="button">show weather and pollution</button>

</form>
<div id="result"></div>

        <div class="sectiontitle">
        <h2>CURRENT WEATHER</h2>
        </div>

        <section id="weather">
          <div class="weather" id="weather1">
          </div>
        </section>
        <div class="sectiontitle">
          <h2>5 DAY FORECAST</h2>
        </div>
      <section id="weather">
        <div class="weather" id="weather2">
        </div>
        <div class="weather" id="weather3">
        </div>
        <div class="weather" id="weather4">
        </div>
      </section>
      <section id="weather">
        <div class="weather" id="weather5">
        </div>
      <!--  <section id="weatherk"> -->
        <div class="weather" id="weather6">
        </div>
      <!-- </section> -->
      </section>
    <!--  <section id="venues">
          <div class="venue" id="venue1">
          </div>
       <div class="venue" id="venue2">
          </div>
       <div class="venue" id="venue3">
        </div>
      </section>  -->
      <!-- Footer -->
      <footer>
        <p align="center">© copyright by got a plan team(2023)
      </footer>
```

```
    </div>

    <!-- JS and jQuery source links -->
    <script src="https://code.jquery.com/jquery-3.2.1.min.js"
        integrity="sha256-hwg4gsxgFZhOsEEamdOYGBf13FyQuiTwlAQgxVSNgt4="
crossorigin="anonymous"></script>

    <script src="src/JS/helper.js"></script>
    <script src="tt.js"></script>
    <script src="t1.js"></script>
</body>

</html>
```

CSS CODE:

```
/* Reset default styles */
body, h1, h2, form {
    margin: 0;
    padding: 0;
}

/* Set font family */
body {
    font-family: 'Montserrat', sans-serif;
    background: rgb(174,238,236);
background: radial-gradient(circle, rgba(174,238,236,1) 13%,
rgba(148,187,233,1) 100%);
}
.main{
    background-image: url('https://images.unsplash.com/photo-1464618663641-
bbdd760ae84a?ixlib=rb-
4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w
=870&q=80');
    margin: 0px auto 0px auto;
    text-align: center;
    background-size: cover;
}
/* Main section */
main {
    background-color: #f5f5f5;
    padding: 30px;
    font-size: 36px;
    color: blue;
    margin-bottom: 600px;
    text-align: center;
```

```css
}
.q {
    font-size: 36px;
    color: #333;
    margin-bottom: 300px;
    text-align: center;

}
#w{
  width: 50px;
}
.q1{
  margin-bottom: 200px;
}
.e{
  text-align: center;
}
#s{
  width: 50px;
  cursor: pointer;
}
#n1{
  color: blue;
}
form {
    margin-top: 20px;
    text-align: center;
}

input[type="text"] {
    padding: 10px;
    width: 300px;
    border: none;
    border-radius: 5px;
    margin-right: 10px;
}

/*#button {
    padding: 10px 20px;
    background-color: #007bff;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
}*/
#city {
  padding: 10px;
  font-size: 16px;
```

```
  border: none;
  border-bottom: 2px solid #ccc;
  background-color: #f2f2f2;
  color: #333;
  transition: background-color 0.3s ease;
}
#city:focus {
  background-color: #fff;
}
#button {
  padding: 10px 30px;
  font-size: 16px;
  background: rgb(2,0,36);
background: linear-gradient(90deg, rgba(2,0,36,1) 0%, rgba(0,168,255,1) 40%);
  border: none;
  border-radius: 25px;
  box-shadow: 0 2px 5px rgba(0,0,0,0.2);
  cursor: pointer;
  transition: background-color 0.3s ease;
}
#button:hover {
  background-color: #ff9800;
  color: #fff;
}
.container {
    max-width: 1200px;
    margin: 0 auto;
    padding: 30px;
}

.sectiontitle {
    margin-top: 50px;
    margin-bottom: 30px;
}

h2 {
    font-size: 24px;
    color: #007bff;
}

/*.weather {
    background-color: #f8f9fa;
    padding: 20px;
    margin-bottom: 20px;
    border-radius: 5px;
}

#weather1 {
```

```
    background-color: #f8d7da;
}

#weather2 {
    background-color: #d4edda;
}

#weather3 {
    background-color: #c3e6cb;
}

#weather4 {
    background-color: #bee5eb;
}

#weather5 {
    background-color: #f8d7da;
}

#weather6 {
    background-color: #d4edda;
}*/

#weather {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-wrap: wrap;
  gap: 20px;
  margin-top: 40px;
}

/* Set up forecast cards */
.weather {
  background-color: #bf4947;
  border-radius: 10px;
  box-shadow: 0px 2px 6px rgba(0, 0, 0, 0.1);
  display: flex;
  flex-direction: column;
  justify-content: space-around;
  align-items: center;
  background: rgb(238,174,202);
background: radial-gradient(circle, rgba(238,174,202,1) 71%,
rgba(148,187,233,1) 100%);
  width: 350px;
  height: 400px;
}
```

```
/* Add unique styling for each forecast card */
#weather2 {
  background-image: url("https://example.com/forecast2.jpg");
  background-position: center;
  background-size: cover;
  background: rgb(238,174,202);
background: radial-gradient(circle, rgba(238,174,202,1) 71%,
rgba(148,187,233,1) 100%);
}

#weather3 {
  background-image: url("https://example.com/forecast3.jpg");
  background-position: center;
  background-size: cover;
  background: rgb(238,174,202);
background: radial-gradient(circle, rgba(238,174,202,1) 71%,
rgba(148,187,233,1) 100%);
}

#weather4 {
  background-image: url("https://example.com/forecast4.jpg");
  background-position: center;
  background-size: cover;
  background: rgb(238,174,202);
background: radial-gradient(circle, rgba(238,174,202,1) 71%,
rgba(148,187,233,1) 100%);
}

#weather5 {
  background-image: url("https://example.com/forecast5.jpg");
  background-position: center;
  background-size: cover;
  background: rgb(238,174,202);
background: radial-gradient(circle, rgba(238,174,202,1) 71%,
rgba(148,187,233,1) 100%);
}

#weather6 {
  background-image: url("https://example.com/forecast6.jpg");
  background-position: center;
  background-size: cover;
  background: rgb(238,174,202);
background: radial-gradient(circle, rgba(238,174,202,1) 71%,
rgba(148,187,233,1) 100%);
length:20px;
}

/* Add a section title */
```

```css
.sectiontitle {
  text-align: center;
  margin-bottom: 10px;
}

.sectiontitle h2 {
  font-size: 28px;
  font-weight: bold;
  text-transform: uppercase;
  letter-spacing: 1px;
  color: #333;
}

#venues {
    display: flex;
    justify-content: center;
    gap: 20px;
    flex-wrap: wrap;
}

.venue {
    flex-basis: calc(33.33% - 20px);
    background-color: #FF69B4 ;
    padding: 20px;
    margin-bottom: 20px;
    border-radius: 5px;
}
footer{
  padding-top: 20px;
}
.feels,.min,.max,.temp,.cond,.pressure,.sea,.ground{
    padding-top: 1px;
    padding-bottom: 10px

}
#z{
  padding-left: 200px;
}
#s{

  width: 30px;
  padding-right: 10px;
}
```

JAVA CODE:

```
const createWeatherHTML = (currentDay) => {
  console.log(currentDay);
  return `<h2>day:${weekDays[new Date().getDay()]}
    <h2>Condition: ${currentDay.weather[0].description}</h2>
        <h2>Temperature:
${kelvinToFahrenheit(currentDay.main.temp)}&deg;C</h2>
        <img
src="https://openweathermap.org/img/wn/${currentDay.weather[0].icon}@2x.png">`
;
};
const createWeatherHTML1 = (currentDay1) => {
  console.log(currentDay1);
  return `<h2 class="feels"><img src="images/hot.png"
id="s">temp_max:${kelvinToFahrenheit(currentDay1.list[8].main.feels_like)}&deg
;C</h2>
                    <h2 class="min"><img src="images/cold.png"
id="s">temp_min:${kelvinToFahrenheit(currentDay1.list[8].main.temp_min)}&deg;C
</h2>
        <h2 class="temp"><img src="images/temperatures.png"
id="s">Temperature:${kelvinToFahrenheit(currentDay1.list[8].main.temp)}&deg;C<
/h2>
            <h2 class="cond"><img src="images/sun.png"
id="s">Condition:${currentDay1.list[8].weather[0].main}</h2>
         <h2 class="pressure"><img id="s"
src="images/pressure.png">pressure:${currentDay1.list[8].main.pressure}</h2>
             <h2 class="sea"><img src="images/sea-level.png"
id="s">sea_level:${currentDay1.list[8].main.sea_level}</h2>
                <h2 class="ground"><img src="images/ground-level.png"
id="s">ground_level:${currentDay1.list[8].main.grnd_level}</h2> `;
};
const createWeatherHTML2 = (currentDay2) => {
  console.log(currentDay2);
  return `<h2 class="feels"><img src="images/hot.png"
id="s">temp_max:${kelvinToFahrenheit(currentDay2.list[16].main.feels_like)}&de
g;C</h2>
                    <h2 class="min"><img src="images/cold.png"
id="s">temp_min:${kelvinToFahrenheit(currentDay2.list[16].main.temp_min)}&deg;
C</h2>
        <h2 class="temp"><img src="images/temperatures.png"
id="s">Temperature:${kelvinToFahrenheit(currentDay2.list[8].main.temp)}&deg;C<
/h2>
            <h2 class="cond"><img src="images/sun.png" id="s">Condition:
${currentDay2.list[16].weather[0].main}</h2>
         <h2 class="pressure"><img id="s"
src="images/pressure.png">pressure:${currentDay2.list[16].main.pressure}</h2>
            <h2 class="sea"><img src="images/sea-level.png"
id="s">sea_level:${currentDay2.list[16].main.sea_level}</h2>
```

```
                <h2 class="ground"><img src="images/ground-level.png"
id="s">ground_level:${currentDay2.list[16].main.grnd_level}</h2> `;
};
const createWeatherHTML3 = (currentDay3) => {
  console.log(currentDay3);
  return `<h2 class="feels"><img src="images/hot.png"
id="s">temp_max:${kelvinToFahrenheit(currentDay3.list[24].main.feels_like)}&de
g;C</h2>
                    <h2 class="min"><img src="images/cold.png"
id="s">temp_min:${kelvinToFahrenheit(currentDay3.list[24].main.temp_min)}&deg;
C</h2>
        <h2 class="temp"><img src="images/temperatures.png"
id="s">Temperature:${kelvinToFahrenheit(currentDay3.list[8].main.temp)}&deg;C<
/h2>
        <h2 class="cond"><img src="images/sun.png"
id="s">Condition:${currentDay3.list[24].weather[0].main}</h2>
         <h2 class="pressure"><img id="s"
src="images/pressure.png">pressure:${currentDay3.list[24].main.pressure}</h2>
             <h2 class="sea"><img src="images/sea-level.png"
id="s">sea_level:${currentDay3.list[24].main.sea_level}</h2>
                <h2 class="ground"><img src="images/ground-level.png"
id="s">ground_level:${currentDay3.list[24].main.grnd_level}</h2> `;
};
const createWeatherHTML4 = (currentDay4) => {
  console.log(currentDay4);
  return `<h2 class="feels"><img src="images/hot.png"
id="s">temp_max:${kelvinToFahrenheit(currentDay4.list[32].main.feels_like)}&de
g;C</h2>
                    <h2 class="min"><img src="images/cold.png" id="s">
temp_min:${kelvinToFahrenheit(currentDay4.list[32].main.temp_min)}&deg;C</h2>
            <h2 class="temp"><img src="images/temperatures.png" id="s">
Temperature: ${kelvinToFahrenheit(currentDay4.list[8].main.temp)}&deg;C</h2>
            <h2 class="cond"><img src="images/sun.png" id="s"> Condition:
${currentDay4.list[32].weather[0].main}</h2>
              <h2 class="pressure"><img id="s" src="images/pressure.png" >
pressure:${currentDay4.list[32].main.pressure}</h2>
                <h2 class="sea"><img src="images/sea-level.png" id="s">
sea_level:${currentDay4.list[32].main.sea_level}</h2>
                  <h2 class="ground"><img src="images/ground-level.png"
id="s"> ground_level:${currentDay4.list[32].main.grnd_level}</h2> `;
};
const createWeatherHTML5 = (currentDay5) => {
  console.log(currentDay5);
  return `<h2 class="feels"><img src="images/hot.png"
id="s">temp_max:${kelvinToFahrenheit(currentDay5.list[39].main.feels_like)}&de
g;C</h2>
```

```
                    <h2 class="min"><img src="images/cold.png"
id="s">temp_min:${kelvinToFahrenheit(currentDay5.list[39].main.temp_min)}&deg;
C</h2>
                    <h2 class="temp"><img src="images/temperatures.png"
id="s">temp_max:${kelvinToFahrenheit(currentDay5.list[39].main.temp_max)}&deg;
C</h2>
        <h2 class="cond"><img src="images/sun.png" id="s">Condition:
${currentDay5.list[39].weather[0].main}</h2>
         <h2 class="pressure"><img id="s"
src="images/pressure.png">pressure:${currentDay5.list[39].main.pressure}</h2>
             <h2 class="sea"><img src="images/sea-level.png"
id="s">sea_level:${currentDay5.list[39].main.sea_level}</h2>
                <h2 class="ground"><img src="images/ground-level.png"
id="s">ground_level:${currentDay5.list[39].main.grnd_level}</h2> `;
};
/*const createWeatherHTML6 = (currentDay6) => {
    console.log(currentDay6);
    return
`<h2>feels_like:${kelvinToFahrenheit(currentDay6.list[0].main.feels_like)}&deg
;C</h2>
                    <h2>temp_min:${kelvinToFahrenheit(currentDay6.list[0].main
.temp_min)}&deg;C</h2>
                    <h2>temp_max:${kelvinToFahrenheit(currentDay6.list[0].main
.temp_max)}&deg;C</h2>
        <h2>Temperature:
${kelvinToFahrenheit(currentDay6.list[0].main.temp)}&deg;C</h2>
        <h2>Condition: ${currentDay6.list[0].weather[0].main}</h2>
         <h2>pressure:${currentDay6.list[0].main.pressure}</h2>
             <h2>sea_level:${currentDay6.list[0].main.sea_level}</h2>
                <h2>ground_level:${currentDay6.list[0].main.grnd_level}</h2>
`;
};
/*const createWeatherHTML7 = (currentDay7) => {
    console.log(currentDay7);
    return ` <h2>Condition: ${currentDay7.list[39].weather[0].main}</h2>
    <h2>day:${weekDays[new Date().getDay()+7]}
    <h2>Condition: ${currentDay7.list[39].weather[0].description}</h2>
        <h2>Temperature:
${kelvinToFahrenheit(currentDay7.list[39].main.temp)}&deg;F</h2>
    <img
src="https://openweathermap.org/img/wn/${currentDay7.list[39].weather[0].icon}
@2x.png">`;
};
const createWeatherHTML8 = (currentDay8) => {
    console.log(currentDay8);
    return ` <h2>Condition: ${currentDay8.list[39].weather[0].main}</h2>
    <h2>day:${weekDays[new Date().getDay()+8]}
    <h2>Condition: ${currentDay8.list[39].weather[0].description}</h2>
```

```
        <h2>Temperature:
${kelvinToFahrenheit(currentDay8.list[39].main.temp)}&deg;F</h2>
    <img
src="https://openweathermap.org/img/wn/${currentDay8.list[39].weather[0].icon}
@2x.png">`;
};*/

const createWeatherHTML12 = (currentDay12) => {
  //var inputValue=currentDay3.[0].lat
  console.log(currentDay12);
  return `<h2>co:${currentDay12.list[0].components.co} &µg/m3</h2>
    <h2>no:${currentDay12.list[0].components.no} µg/m3</h2>
    <h2>no2:${currentDay12.list[0].components.no2} µg/m3</h2>
    <h2>o3:${currentDay12.list[0].components.o3} µg/m3</h2>
    <h2>so2:${currentDay12.list[0].components.so2} µg/m3</h2>`;
};
/*const createWeatherHTML3 = (currentDay3) => {
    return `<img src=${currentDay3.hits[0].webformatURL}>`

};*/


const kelvinToFahrenheit = (k) => (k - 273.15).toFixed(0);
```

JAVA(1) CODE:

```
const clientId = 'OZATO4USDMG5ZYJMKY54YTK4Y2TPDWK34PSUG3GIV0RULAXP';
const clientSecret = 'OSXQJ50AFX20J2HNOHKFGPWKVVNYNSP13IOGOB5YETXLPLR4';
const url = 'https://api.foursquare.com/v2/venues/explore?near=';

// OpenWeather Info
const openWeatherKey = '7d20a5895a38a13dcdbb647529a3d4ca';
const weatherUrl = 'https://api.openweathermap.org/data/2.5/weather';
//5 day forecast
const openWeatherKey1 = '8c8a69668bf6a4341f61a3bf5e833bbd';
const weatherUrl1 = 'https://api.openweathermap.org/data/2.5/forecast';
//background
const
weatherUrl2='http://api.openweathermap.org/data/2.5/air_pollution/forecast'
// Page Elements
const $input = $('#city');
const $submit = $('#button');
const $destination = $('#destination');
const $container = $('.container');
const $venueDivs = [ $('#venue1'), $('#venue2'), $('#venue3'),
$('#venue4'),$('#venue5'),$('#venue6'),$('#venue7'),$('#venue8')];
const $weatherDiv = $('#weather1');
```

```
const $weatherDiv1= $('#weather2');
const $weatherDiv2= $('#weather3');
const $weatherDiv3= $('#weather4');
const $weatherDiv4= $('#weather5');

const $weatherDiv5= $('#weather6');
/*const $weatherDiv6= $('#venue1');
const $weatherDiv7= $('#venue2');
const $weatherDiv8= $('#venue3');*/
const $weatherDiv12= $('#weather25');
const $myElement2=$("#myElement");
const $myElement3=$("#myElement1");
const $back1=$('#back');
const $click1=$('#click2');
const weekDays = [ 'Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday',
'Friday', 'Saturday' ];
const monthNames = ["January", "February", "March", "April", "May",
"June","July", "August", "September", "October", "November", "December"];
    // Function to fetch weather data from OpenWeatherMap API
    async function fetchWeatherData(city) {
     // const apiKey = ; // Replace with your OpenWeatherMap API key
      const url =
`https://api.openweathermap.org/data/2.5/weather?q=${$input.val()}&appid=${ope
nWeatherKey1}`;

      try {
        const response = await fetch(url);
        const data = await response.json();

        if (response.ok) {
          // Extract longitude and latitude from the weather data
          const longitude = data.coord.lon;
          const latitude = data.coord.lat;

          // Fetch pollution data using the latitude and longitude
          const pollutionData = await fetchPollutionData(latitude, longitude);

          return { longitude, latitude, pollutionData };
        } else {
          throw new Error(`Failed to fetch weather data: ${data.message}`);
        }
      } catch (error) {
        console.error(error);
      }
    }

    // Function to fetch pollution data from OpenWeatherMap API
    async function fetchPollutionData(latitude, longitude) {
```

```
      const apiKey = 'YOUR_API_KEY'; // Replace with your OpenWeatherMap API
key
      const url =
`https://api.openweathermap.org/data/2.5/air_pollution?lat=${latitude}&lon=${l
ongitude}&appid=${openWeatherKey1}`;

      try {
        const response = await fetch(url);
        const data = await response.json();

        if (response.ok) {
          // Extract pollution data
          const aqi = data.list[0].main.aqi;
          const pm2_5 = data.list[0].components.pm2_5;
          const pm10 = data.list[0].components.pm10;
          const so2 = data.list[0].components.so2;
          const no2 = data.list[0].components.no2;
          const co = data.list[0].components.co;
          const o3 = data.list[0].components.o3;

          return { aqi, pm2_5, pm10, so2, no2, co, o3 };
        } else {
          throw new Error(`Failed to fetch pollution data: ${data.message}`);
        }
      } catch (error) {
        console.error(error);
      }
    }

    // Function to handle form submission
    function handleSubmit() {
      const city = document.getElementById('cityInput').value;
      fetchWeatherData(city)
        .then(({ longitude, latitude, pollutionData }) => {
          document.getElementById('result').innerHTML = ` <P
align="center">Longitude: ${longitude}<br>
          <br>
          Latitude: ${latitude}<br>
      <br>
      <br>
      <br>
          AQI: ${pollutionData.aqi}<br>
      <br>
          <img src="images/pm2.5-icon.png" id="w">PM2.5
:${pollutionData.pm2_5} µg/m3<br>
      <br>
          <img src="images/pm10-icon.png" id="w">PM10: ${pollutionData.pm10}
µg/m3<br>
```

```
      <br>
      <br>
          <p align="center"><img src="images/so2.png"
id="w">SO2:${pollutionData.so2} µg/m3<br></p>
      <br>
           <p align="center"><img src="images/no2.png" id="w">NO2:
${pollutionData.no2} µg/m3<br></p>
      <br>
          <p align="center"><img src="images/CO.png" id="w">CO:
${pollutionData.co} µg/m3<br></p>
      <br>
          <p align="center"><img src="images/O3.png"id="w">O3:
${pollutionData.o3} µg/m3</p>
      <img src="images/conc.png" id="z">
      `;
        })
        .catch(error => {
          console.error(error);
          document.getElementById('result').innerHTML = 'Error: ' +
error.message;
        });
    }


const getpollution = async () => {
    const urlToFetch =
`${weatherUrl2}?lat=${$myElement2.val()}&lon=${$myElement3.val()}&appid=${open
WeatherKey1}`;
    try {
        const response = await fetch(urlToFetch);
        if (response.ok) {
            const jsonResponse = await response.json();
            return jsonResponse;
        }
    } catch (error) {
        console.log(error);
    }
};
// get Data from OpenWeather
const getForecast = async () => {
    const urlToFetch =
`${weatherUrl}?&q=${$input.val()}&APPID=${openWeatherKey}`;
    try {
        const response = await fetch(urlToFetch);
        if (response.ok) {
            const jsonResponse = await response.json();
            return jsonResponse;
        }
    } catch (error) {
```

```
        console.log(error);
    }
};
const get5dayForecast = async () => {
    const urlToFetch =
`${weatherUrl1}?&q=${$input.val()}&appid=${openWeatherKey1}`;
    try {
        const response = await fetch(urlToFetch);
        if (response.ok) {
            const jsonResponse = await response.json();
            return jsonResponse;
        }
    } catch (error) {
        console.log(error);
    }
};

// Render functions
/*const renderbackground = (backg) => {
    const backgroundContent = createWeatherHTML3(backg);
    $back1.append( backgroundContent);
};*/
const renderForecast12= (day1) => {
    const weatherContent12 = createWeatherHTML12(day1);
    $weatherDiv12.append(weatherContent12);
};
const renderForecast = (day) => {
    const weatherContent = createWeatherHTML(day);
    $weatherDiv.append(weatherContent);
};
const renderForecast1 = (day1) => {
    const weatherContent1 = createWeatherHTML1(day1);
    $weatherDiv1.append(weatherContent1);
};
const renderForecast2 = (day2) => {
    const weatherContent2 = createWeatherHTML2(day2);
    $weatherDiv2.append(weatherContent2);
};
const renderForecast3 = (day3) => {
    const weatherContent3 = createWeatherHTML3(day3);
    $weatherDiv3.append(weatherContent3);
};
const renderForecast4 = (day4) => {
    const weatherContent4 = createWeatherHTML4(day4);
    $weatherDiv4.append(weatherContent4);
};
const renderForecast5 = (day5) => {
    const weatherContent5 = createWeatherHTML5(day5);
```

```
    $weatherDiv5.append(weatherContent5);
};
/*const renderForecast6 = (day6) => {
    const weatherContent6 = createWeatherHTML6(day6);
    $weatherDiv6.append(weatherContent6);
};
const renderForecast7 = (day7) => {
    const weatherContent7 = createWeatherHTML7(day7);
    $weatherDiv7.append(weatherContent7);
};
const renderForecast8 = (day8) => {
    const weatherContent8 = createWeatherHTML8(day8);
    $weatherDiv8.append(weatherContent8);
};*/

const executeSearch = () => {
    $weatherDiv.empty();
    $weatherDiv1.empty();
    $weatherDiv2.empty();
        $weatherDiv3.empty();
        $weatherDiv4.empty();
            $weatherDiv5.empty();
    $destination.empty();
    $back1.empty();
    $container.css('visibility', 'visible');
    getpollution().then((forecast12) => renderForecast12(forecast12));
    getForecast().then((forecast) => renderForecast(forecast));
    get5dayForecast().then((weather) => renderForecast1(weather));
    get5dayForecast().then((weather1) => renderForecast2(weather1));
    get5dayForecast().then((weather2) => renderForecast3(weather2));
    get5dayForecast().then((weather3) => renderForecast4(weather3));
    get5dayForecast().then((weather4) => renderForecast5(weather4));
/*  get5dayForecast().then((weather6) => renderForecast6(weather6));
    get5dayForecast().then((weather7) => renderForecast7(weather7));
    get5dayForecast().then((weather8) => renderForecast8(weather8));*/

    //getbackground().then((back1g)=>renderbackground(back1g));
/*$(document).ready(function(){
    var originalHref = $("#click2").attr("href");
    $("#click2").click(function(){
        var userInput = $input.val();
        var newHref = originalHref.replace("3/", "3/" + userInput);
        $(this).attr("href", newHref);
      });
    });*/
    return false;
};
```

```
$submit.click(executeSearch);
```

OUTPUT