**# 06 Server-Side APIs: Weather Dashboard**

**## Your Task**

Third-party APIs allow developers to access their data and functionality by making requests with specific parameters to a URL. Developers are often tasked with retrieving data from another application's API and using it in the context of their own. Your challenge is to build a weather dashboard that will run in the browser and feature dynamically updated HTML and CSS.

Use the [5 Day Weather Forecast](https://openweathermap.org/forecast5) to retrieve weather data for cities. The base URL should look like the following: `https://api.openweathermap.org/data/2.5/forecast?lat={lat}&lon={lon}&appid={API key}`. After registering for a new API key, you may need to wait up to 2 hours for that API key to activate.

**\*\*Hint\*\***: Using the 5 Day Weather Forecast API, you'll notice that you will need to pass in coordinates instead of just a city name. Using the OpenWeatherMap APIs, how could we retrieve geographical coordinates given a city name?

You will use `localStorage` to store any persistent data. For more information on how to work with the OpenWeather API, refer to the [Full-Stack Blog on how to use API keys](https://coding-boot-camp.github.io/full-stack/apis/how-to-use-api-keys).

**## User Story**

```

AS A traveler

I WANT to see the weather outlook for multiple cities

SO THAT I can plan a trip accordingly

```

**## Acceptance Criteria**

```

GIVEN a weather dashboard with form inputs

WHEN I search for a city

THEN I am presented with current and future conditions for that city and that city is added to the search history

WHEN I view current weather conditions for that city

THEN I am presented with the city name, the date, an icon representation of weather conditions, the temperature, the humidity, and the wind speed

WHEN I view future weather conditions for that city

THEN I am presented with a 5-day forecast that displays the date, an icon representation of weather conditions, the temperature, the wind speed, and the humidity

WHEN I click on a city in the search history

THEN I am again presented with current and future conditions for that city

```

**## Mock-Up**

The following image shows the web application's appearance and functionality:

![The weather app includes a search option, a list of cities, and a five-day forecast and current weather conditions for Atlanta.](./Assets/06-server-side-apis-homework-demo.png)

**## Grading Requirements**

> **\*\*Note\*\***: If a Challenge assignment submission is marked as “0”, it is considered incomplete and will not count towards your graduation requirements. Examples of incomplete submissions include the following:

>

> \* A repository that has no code

>

> \* A repository that includes a unique name but nothing else

>

> \* A repository that includes only a README file but nothing else

>

> \* A repository that only includes starter code

This Challenge is graded based on the following criteria:

**### Technical Acceptance Criteria: 40%**

\* Satisfies all of the above acceptance criteria plus the following:

    \* Uses the OpenWeather API to retrieve weather data.

    \* Uses `localStorage` to store persistent data.

**### Deployment: 32%**

\* Application deployed at live URL.

\* Application loads with no errors.

\* Application GitHub URL submitted.

\* GitHub repository that contains application code.

**### Application Quality: 15%**

\* Application user experience is intuitive and easy to navigate.

\* Application user interface style is clean and polished.

\* Application resembles the mock-up functionality provided in the Challenge instructions.

**### Repository Quality: 13%**

\* Repository has a unique name.

\* Repository follows best practices for file structure and naming conventions.

\* Repository follows best practices for class/id naming conventions, indentation, quality comments, etc.

\* Repository contains multiple descriptive commit messages.

\* Repository contains quality readme file with description, screenshot, and link to deployed application.

**## Review**

You are required to submit BOTH of the following for review:

\* The URL of the functional, deployed application.

\* The URL of the GitHub repository. Give the repository a unique name and include a readme describing the project.

- - -

© 2023 edX Boot Camps LLC. Confidential and Proprietary. All Rights Reserved.

It is what I have for now.

<!DOCTYPE html>

<html>

<head>

    <title>Weather Dashboard</title>

</head>

<body>

    <header>

        <h1>Weather Dashboard</h1>

    </header>

    <div>

        <div>

            <div>

                Search for a City:

            </div>

            <input id="search" placeholder="search...">

            <button id="search\_btn">search</button>

            <div id="cities">

                <!-- <ul id="cities">

                    <li id="city\_1" class="city"></li>

                    <li id="city\_2" class="city"></li>

                    <li id="city\_3" class="city"></li>

                    <li id="city\_4" class="city"></li>

                    <li id="city\_5" class="city"></li>

                    <li id="city\_6" class="city"></li>

                    <li id="city\_7" class="city"></li>

                    <li id="city\_8" class="city"></li>

                </ul> -->

            </div>

            <br>

            <br>

            <div id="something">

                dfdfdf

            </div>

        </div>

    </div>

    <script>

        var city = ["Atlanta", "Denver", "Seattle", "San Francisco", "Orlando", "New York", "Chicago", "Austin"]

        localStorage.setItem("City", JSON.stringify(city))

        var cities = JSON.parse(localStorage.getItem("City"))

        var something = document.getElementById("something");

        // for (i=0; i<(cities.length) ; i++){

        //     // console.log()

        //     document.querySelector(`#city\_${i+1}`).innerHTML = cities[i]

        // }

        for (let i = 0; i < cities.length; i++) {

            const city\_ = cities[i];

            var city\_div=document.createElement("div");

            city\_div.textContent=city\_;

            document.getElementById("cities").append(city\_div)

        }

        var searchButton = document.getElementById("search\_btn");

        var inputValue = document.getElementById("search");

        fetch("https://api.openweathermap.org/data/2.5/weather?q=Atlanta,usa&APPID=b3d8a2004c09344e9c1563543c626c92")

        .then(response => response.json())

        .then(data => console.log(data));

        // for (i = 0; i < weatherULEl.length; i++) {

        //         var tempDay = data.daily[i].temp.day;

        //         var windSpeed = data.daily[i].wind\_speed;

        //         var humid = data.daily[i].humidity;

        //         var uvIndex = data.daily[i].uvi;

        //         var iconID = data.daily[i].weather[0].icon;

        //         var utcDate = data.daily[i].dt;

        //         var date = new Date(utcDate \* 1000);

        //         var day = date.getDate();

        //         var month = date.getMonth() + 1;

        //         var year = date.getFullYear();

        //         var dateFormatted = month.toString() + '/' + day.toString() + '/' + year.toString();

        //         //  adding the formatting to display on website

        //         // current day formatting

        //         if (i === 0) {

        //             currentDataList.innerText = searchCity + "  (" + dateFormatted + ")";

        //             weatherULEl[i].innerHTML = "<li><img src='https://openweathermap.org/img/wn/" + iconID.toString() + ".png'></li><li>Temp: " + tempDay.toString() + "</li> <li>Wind: " + windSpeed.toString() + " MPH</li> <li>Humidity: " + humid.toString() + "%</li> <li id='uvI'>UV Index: " + uvIndex.toString() + "</li>";

        //             var uvIEl = document.getElementById('uvI');

        //             console.log(uvIEl);

        //             if (uvIndex <= 2) {

        //                 uvIEl.classList.add("low");

        //             }

        //             else if (uvIndex > 7) {

        //                 uvIEl.classList.add("high");

        //             }

        //             else {

        //                 uvIEl.classList.add("medium");

        //             }

        //         }

        //         // 5 day forecast formatting

        //         else {

        //             weatherULEl[i].innerHTML = "<li><label>Date:</label>" + dateFormatted + "</li> <li><img src='https://openweathermap.org/img/wn/" + iconID.toString() + ".png'></li> <li>Temp: " + tempDay.toString() + "</li> <li>Wind: " + windSpeed.toString() + " MPH</li> <li>Humidity: " + humid.toString() + "%</li>"

        //         }

        //     };

        searchButton.addEventListener("click", (event)=>{

            // console.log(event)

            event.preventDefault();

            var val=inputValue.value;

            // console.log(val)

            if(cities.includes(val)){

                console.log("Cities: ", cities)

                fetch(`https://api.openweathermap.org/data/2.5/weather?q=${val},usa&APPID=b3d8a2004c09344e9c1563543c626c92`)

                .then(response => response.json())

                .then(data => console.log(data));

                console.log("data type: ", data)

                var temp = data.main.temp;

                console.log(temp)

                something.innerHTML = temp;

                var tempDay = data.daily[i].temp.day;

                var windSpeed = data.daily[i].wind\_speed;

                var humid = data.daily[i].humidity;

                // var uvIndex = data.daily[i].uvi;

                // var iconID = data.daily[i].weather[0].icon;

                // var utcDate = data.daily[i].dt;

                var date = new Date(utcDate \* 1000);

                var day = date.getDate();

                var month = date.getMonth() + 1;

                var year = date.getFullYear();

                var dateFormatted = month.toString() + '/' + day.toString() + '/' + year.toString();

                //  adding the formatting to display on website

                // current day formatting

                if (i === 0) {

                    currentDataList.innerText = searchCity + "  (" + dateFormatted + ")";

                    weatherULEl[i].innerHTML = "<li><img src='https://openweathermap.org/img/wn/" + iconID.toString() + ".png'></li><li>Temp: " + tempDay.toString() + "</li> <li>Wind: " + windSpeed.toString() + " MPH</li> <li>Humidity: " + humid.toString() + "%</li> <li id='uvI'>UV Index: " + uvIndex.toString() + "</li>";

                    var uvIEl = document.getElementById('uvI');

                    console.log(uvIEl);

                    if (uvIndex <= 2) {

                        uvIEl.classList.add("low");

                    }

                    else if (uvIndex > 7) {

                        uvIEl.classList.add("high");

                    }

                    else {

                        uvIEl.classList.add("medium");

                    }

                }

                // 5 day forecast formatting

                else {

                    weatherULEl[i].innerHTML = "<li><label>Date:</label>" + dateFormatted + "</li> <li><img src='https://openweathermap.org/img/wn/" + iconID.toString() + ".png'></li> <li>Temp: " + tempDay.toString() + "</li> <li>Wind: " + windSpeed.toString() + " MPH</li> <li>Humidity: " + humid.toString() + "%</li>"

                }

            }else{

                console.log(typeof("Local Storage City: ", cities[0]))

                console.log(typeof("Input Value: ", val))

                console.log("Wrong City")

            }

            // var storageItem=JSON.parse(window.localStorage.getItem("savedCities")) || [];

            // storageItem.push(val);

            // window.localStorage.setItem("savedCities", JSON.stringify(storageItem))

        })

        // var storageItems=JSON.parse(window.localStorage.getItem("savedCities"))

    </script>

</body>

</html>

 <!-- <!DOCTYPE html>

<html lang="en">

<head>

     <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

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

     <title>Document</title>

 </head>

 <body>

     <input type="text" id="city">

     <button id="search">Search</button>

     <div id="storagebuttonscontainer"></div>

 <script>

     var inputValue=document.getElementById("city");

     var searchButton=document.getElementById("search");

     searchButton.addEventListener("click", (event)=>{

        search  event.preventDefault();

         var val=inputValue.value;

         console.log(val)

         var storageItem=JSON.parse(window.localStorage.getItem("savedCities")) || [];

         storageItem.push(val);

         window.localStorage.setItem("savedCities", JSON.stringify(storageItem))

     })

     var storageItems=JSON.parse(window.localStorage.getItem("savedCities"))

     for (let i = 0; i < storageItems.length; i++) {

         const element = storageItems[i];

         var storageButton=document.createElement("button");

         storageButton.textContent=element;

         document.getElementById("storagebuttonscontainer").append(storageButton)

     }

 </script>

 </body>

 </html>  -->