



Cubstart Web Lab 5

Administrivia

1. HW 4 (GradeScope assignment) due tonight
2. HW 5: OpenWeatherMap API is due in one week.

*attendance not required today

*these slides are posted on the website



OpenWeatherMap!

Go to openweathermap.org and click “API” on navbar

Scroll down and you’ll see that OpenWeatherMap has LOTS of endpoints and LOTS of functionalities!



Which ones are we going to use in HW 5?

1. Geocoding API

a. Documentation:

https://openweathermap.org/api/geocoding-api#direct_name_how

b. “Direct geocoding converts the specified name of a location or zip/post code into the exact geographical coordinates” (latitude and longitude)

2. Current weather data API

a. Documentation: <https://openweathermap.org/current>

b. Provide a latitude and longitude as query parameters to the API call, API will respond with lots of weather information!



Understanding the HW 5 Skeleton Code



Skeleton Code

```

1  // CLUE #0: See more info in the homework spec.
2  apiKey = /* YOUR API KEY HERE */;
3
4  geocodingUrl = "http://api.openweathermap.org/geo/1.0/direct?q="; // This
5  weatherUrl = "https://api.openweathermap.org/data/2.5/weather?"; //This is
6  submitButton = document.getElementById("submit");
7
8  // DOM Targetting: Select the input element with the id "cityName" (CLUE #1)
9  cityName = /* YOUR CODE HERE */
10 mainWeather = document.getElementById("mainWeather");
11 weatherDescription = document.getElementById("weatherDescription");
12
13 // This adds an event listener to check when the submit button is clicked,
14 // then if the cityName's value is not blank, we call setWeatherDescription
15 submitButton.addEventListener("click", function () {
16     if (cityName.value != "") {
17         setWeatherDescription(cityName.value);
18     }
19 });
20

```

Lines 1-19

- Setting up variables
- DOM targeting!
- Setting up button functionality
 - On click, if the user has typed in a city name, then call the function setWeatherDescription on whatever the user inputted



Skeleton Code

```
25  async function getLatLon(city) {
26
27      // Let's create the API url. (CLUE #2)
28      let url = /* YOUR CODE HERE */
29
30      // Send a GET request to the url that you wrote above! (CLUE #3)
31      let response = /* YOUR CODE HERE */
32      let data = await response.json() //This line parses the response
33
34      // Let's return a JavaScript object here! (CLUE #4)
35      return {
36          "lat": /* YOUR CODE HERE */,
37          "lon": /* YOUR CODE HERE */
38      }
39  }
40
```

Lines 25-39

- Defining a function!
 - Inputs: city name
 - Outputs: latitude and longitude of that city



Skeleton Code

```
40
41 //This function makes a GET request to retrieve weather
42 // (CLUE #5)
43 async function getWeather(lat, lon) {
44   let url = /* Format the appropriate API url to retrieve weather
45   let response = /* Fetch data from the url you just created
46   let data = /* Parse the response as JSON here */
47
48   // Return the main weather and weather description
49   return {
50     "main": /* YOUR CODE HERE */,
51     "description": /* YOUR CODE HERE */
52   }
53 }
54
```

Lines 43-54

- Defining another function!
 - Inputs: latitude and longitude
 - Outputs: info about the weather at that latitude and longitude



Skeleton Code

```

55 // This function gets the weather using the functions yo
56 async function setWeatherDescription(city) {
57   // This line calls getLatLon on the city name provided
58   let coordinateData = await getLatLon(city)
59
60   // Extract the lat and lon from coordinateData. (CLUE
61   const lat = /* YOUR CODE HERE */
62   const lon = /* YOUR CODE HERE */
63
64
65   let weatherData = await getWeather(lat, lon)
66
67   // Same thing here, but we want to set mainWeather and
68   // (CLUE #7)
69   mainWeather.innerHTML = /* YOUR CODE HERE */
70   /* Set weatherDescription's innerHTML here */
71 }
72

```

*this is the function that runs on button click!

Lines 56-71

- Inputs: city name inputted by the user
- What does it do:
 - calls **getLatLon** to get latitude and longitude of that city
 - calls **getWeather** to get the weather at that lat and lon
 - Finally, display the weather in our HTML so it shows up on our web page



What is an API Key?

- Unique identifier (alphanumeric string) that is assigned to an API client (you!) and allows that client to access the APIs web services
- Purpose:
 - API owners can monitor usage of their API
 - Authorization (API owners can authorize specific projects with specific keys to have special access or limited access)



HW 5 Walkthrough!

