**Advanced weather app by using API**

1. [Scaffolding the Weather App](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#scaffolding)
   1. [Find a Weather API](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#weatherapi)
   2. [What is OpenWeather?](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#OpenWeather)
   3. [Where To Find Weather Icons and Weather UI Kits](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#wheretofind)
2. [Define the Page Markup](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#PageMarkup)
3. [Specify Some Basic Styles](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#BasicStyles)
4. [Set the Main Styles](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#MainStyles)
5. [Add the JavaScript](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#addJavaScript)
6. [Bonus: Get Real-Time Weather Data With the weatherstack API](https://webdesign.tutsplus.com/tutorials/build-a-simple-weather-app-with-vanilla-javascript--cms-33893#weatherstack)

**1. Scaffolding the Weather App**

Before start creating our app, there are a few things that we have to take into consideration.

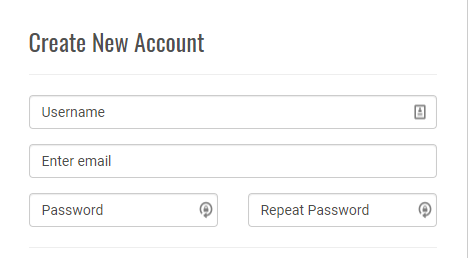
**Find a Weather API**

First things first, we have to find a provider that will let us incorporate its weather data into our app. Luckily enough, there are several [different providers](https://rapidapi.com/search/weather) out there for developing weather apps. Most of them include a free package along with premium subscriptions that scale depending on the services/features.

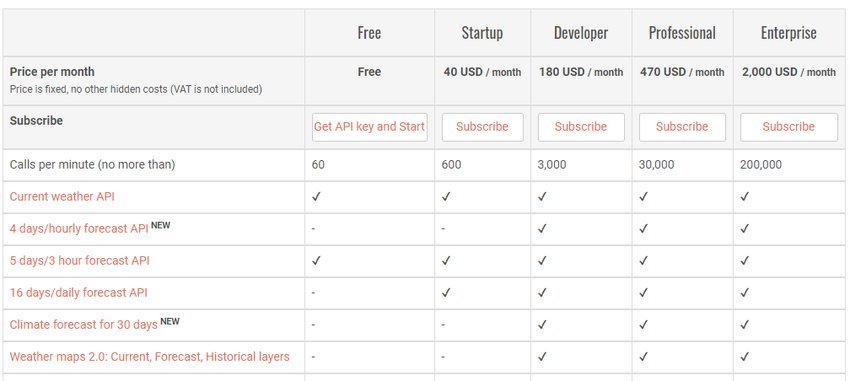
**What is OpenWeather?**

In our case, we’re going to use [OpenWeatherMap](https://openweathermap.org/" \t "_blank), one of the most popular free choices. OpenWeather describes itself as a group of IT experts and data scientists that does deep weather data science since 2014. For each point on Earth, OpenWeather provides reliable historical, current and forecasted weather data via light-speed APIs.

To take advantage of its capabilities, first, we have to [sign up](https://home.openweathermap.org/users/sign_up) for an API key:



This service comes with different packages. As you can see from the visualization below, the starter (free) one allows 60 calls per minute which suits our needs:



So before continuing, please make sure that you’ve registered for an API key. Later, we’ll include that key in our script.

Keep in mind that the best way to test the app is by [forking the CodePen demo](https://codepen.io/tutsplus/details/gObLaEP) and including your own key. If we all share the same key, the app will probably not work due to API call limits.

**Where To Find Weather Icons and Weather UI Kits**

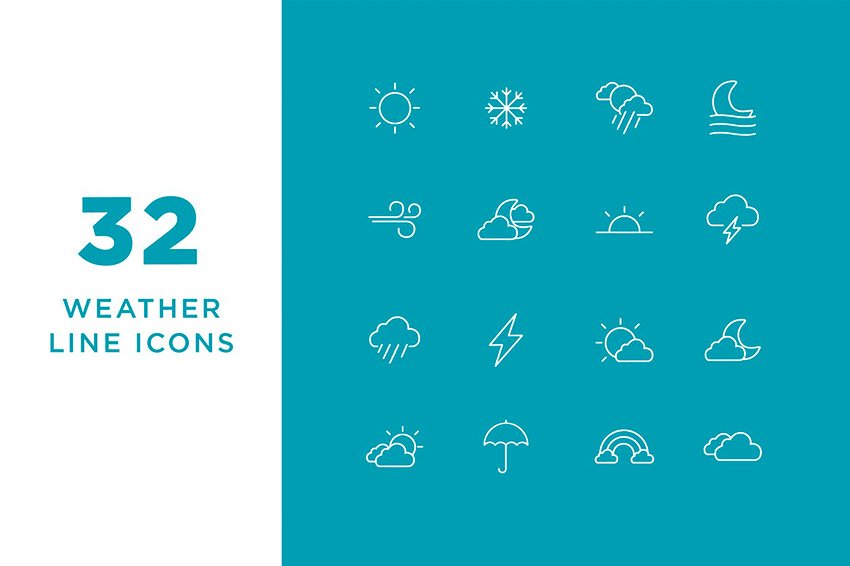
Before we start with the weather API JavaScript code, we’ll need weather app icons. It’s worth noting that OpenWeatherMap comes with its own icon set and we’ll take a look at those. However, we’ll go one step further and use some custom ones.

Now, if you’re a web designer or work on multiple web design projects, Envato Elements is the best option for you.

For a low monthly fee, this subscription-based marketplace gives you unlimited access to everything you’ll need for your projects.

Here we’ll review some top [weather icon vector sets](https://elements.envato.com/graphics/icons/weather) and weather UI kits. Keep browsing Envato Elements [weather UI kit library](https://elements.envato.com/graphic-templates/ux-and-ui-kits/weather) for more!

**1.** [**Weather Icons and Font**](https://elements.envato.com/weather-icons-and-font-S6LQ7C)

[](https://elements.envato.com/weather-icons-and-font-S6LQ7C)

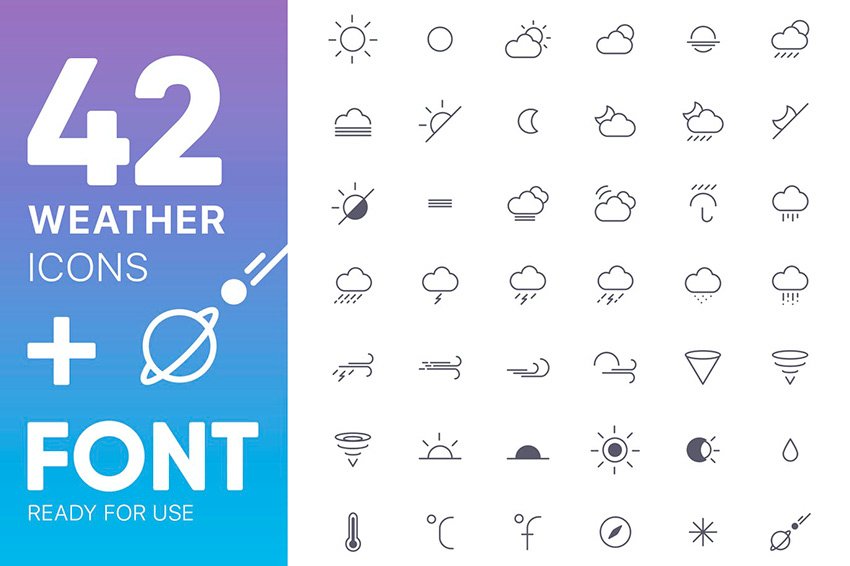
If you like simple weather app icons, these are for you. This weather icon vector set comes with 32 vector icons and a Forecast font. This complete weather icon vector set includes 32 SVG files as well as HTML and CSS references.

**2.** [**UICON Weather Icons**](https://elements.envato.com/uicon-weather-icons-4E5KUZ)

[](https://elements.envato.com/uicon-weather-icons-4E5KUZ)

This weather icon vector set features a modern and fresh look. There are 12 weather app icons in two color versions. All weather icons are crafted in EPS with a minimal design.

**3.** [**Vector Weather Icons and Fonts**](https://elements.envato.com/weather-icons-and-fonts-DEFH5V)

[](https://elements.envato.com/weather-icons-and-fonts-DEFH5V)

If you’re looking for a complete weather icon vector set, check this out. The weather app icons pack comes with 42 unique icons and one font. Weather icons come in AI, PSD and SVG formats.

**4.** [**Weather App Template**](https://elements.envato.com/weather-app-template-KUVX3AK)

[](https://elements.envato.com/weather-app-template-KUVX3AK)

Now, if you’d like to build a weather app with a template, we’ve got this weather UI kit. You can edit your own weather app using this template in Photoshop, Illustrator and Adobe XD.

**5.** [**Weather Small Widget For Web or Mobile**](https://elements.envato.com/weather-small-widget-for-webmobile-84XHUX)

[](https://elements.envato.com/weather-small-widget-for-webmobile-84XHUX)

This is another cool weather UI kit if you want to build a different weather app. This weather UI kit features a clean and minimal design. You can edit it using Sketch.

**2. Define the Page Markup**

Now that we’ve seen some top weather app icons, it’s time to build a weather website with JavaScript. We’ll define two sections.

The first section will include a heading, a search form, and an empty span element. This element will become visible with an appropriate message under certain conditions. Specifically, if there isn’t any weather data available for a requested city or the data for this city are already known.

The second section will include a list of cities. By default, it won’t contain any cities. But, as we start searching for the weather for a specific city, if weather data is available, a corresponding list item (city) will be appended to the unordered list.

Here’s the initial page markup:

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15 | <section class="top-banner">    <div class="container">      <h1 class="heading">Simple Weather App</h1>      <form>        <input type="text" placeholder="Search for a city" autofocus>        <button type="submit">SUBMIT</button>        <span class="msg"></span>      </form>    </div>  </section>  <section class="ajax-section">    <div class="container">      <ul class="cities"></ul>    </div>  </section> |

**Note:** In our CodePen demo, the autofocus attribute of the search field won’t work. In fact, it’ll throw the following error which you can see if you open your browser console:

The cross-origin error due to the autofocus attribute

However, if you run this app locally (not as a CodePen project), this issue won’t exist.

And here’s the markup associated with a list item that we’ll generate dynamically through JavaScript:

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11 | <li class="city">    <h2 class="city-name" data-name="...">      <span>...</span>      <sup>...</sup>    </h2>    <span class="city-temp">...<sup>°C</sup></span>    <figure>      <img class="city-icon" src="..." alt="...">      <figcaption>...</figcaption>    </figure>  </li> |

**3. Specify Some Basic Styles**

With the markup for the weather app ready, we’ll forge on with the CSS. The first step, as always, is to specify some CSS variables and common reset styles:

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49 | :root {    --bg\_main: #0a1f44;    --text\_light: #fff;    --text\_med: #53627c;    --text\_dark: #1e2432;    --red: #ff1e42;    --darkred: #c3112d;    --orange: #ff8c00;  }    \* {    margin: 0;    padding: 0;    box-sizing: border-box;    font-weight: normal;  }    button {    cursor: pointer;  }    input {    -webkit-appearance: none;  }    button,  input {    border: none;    background: none;    outline: none;    color: inherit;  }    img {    display: block;    max-width: 100%;    height: auto;  }    ul {    list-style: none;  }    body {    font: 1rem/1.3 "Roboto", sans-serif;    background: var(--bg\_main);    color: var(--text\_dark);    padding: 50px;  } |

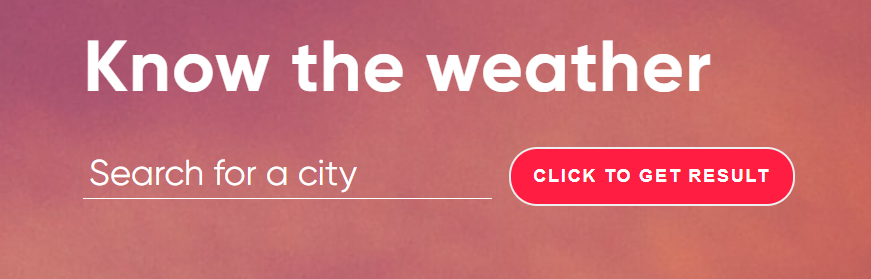
**4. Set the Main Styles**

Let’s now discuss the main styles of our weather app.

**Section #1 Styles**

First, we’ll add some straightforward styles to the elements of the first section.

On medium screens and above (>700px) the layout should look like this:

****

**Html code**

**<!doctype html>**

**<html>**

**<head>**

**<title>My weather App</title>**

**<link rel="stylesheet" href="style.css" />**

**<style>**

**body {**

**background-image: url('img2.jpg');**

**}**

**</style>**

**</head>**

**<body style="background-image:img1.jpg;">**

**<div class="api">**

**<div class="container" style="font-size:20px;color: green; font-family:gilroy;">You can get the <B>API</B> from openweather by clicking on this link <a target="\_blank" href="https://home.openweathermap.org/users/sign\_up" style="color:blue;">Get yours here for free!</a>**

**</div>**

**</div>**

**<section class="top-banner">**

**<div class="container">**

**<h1 class="heading" style="font-family:gilroy;">Know the weather</h1>**

**<form>**

**<input type="text" placeholder="Search for a city" autofocus style="font-family:gilroy;">**

**<button type="submit" style="border:2px solid #eeeeee; border-radius: 25px;">CLICK TO GET RESULT</button>**

**<span class="msg"></span>**

**</form>**

**</div>**

**</section>**

**<section class="ajax-section">**

**<div class="container">**

**<ul class="cities"></ul>**

**</div>**

**</section>**

**<script src="scripts.js">**

**</script>**

**<footer class="page-footer">**

**<div class="container" >**

**<strong style="font-size:30px; color:black; font-family:gilroy;"><center>Made with <span>❤</span> by <a href="http://georgemartsoukos.com/" target="\_blank">Muzeeburrahaman</a><center>**

**</strong>**

**</div>**

**</footer>**

**</body>**

**</html>**

**Css code**

**:root {**

**--bg\_main: #0a1f44;**

**--text\_light: #fff;**

**--text\_med: #53627c;**

**--text\_dark: #1e2432;**

**--red: #ff1e42;**

**--darkred: #c3112d;**

**--orange: #ff8c00;**

**}**

**\* {**

**margin: 0;**

**padding: 0;**

**box-sizing: border-box;**

**font-weight: normal;**

**}**

**a {**

**color: inherit;**

**text-decoration: none;**

**}**

**button {**

**cursor: pointer;**

**}**

**input {**

**-webkit-appearance: none;**

**}**

**button,**

**input {**

**border: none;**

**background: none;**

**outline: none;**

**color: inherit;**

**}**

**img {**

**display: block;**

**max-width: 100%;**

**height: auto;**

**}**

**ul {**

**list-style: none;**

**}**

**body {**

**font: 1rem/1.3 "Roboto", sans-serif;**

**background: var(--bg\_main);**

**color: var(--text\_dark);**

**padding: 70px;**

**}**

**.container {**

**width: 100%;**

**max-width: 1200px;**

**margin: 0 auto;**

**padding: 0 15px;**

**}**

**/\* SECTION #1**

**–––––––––––––––––––––––––––––––––––––––––––––––––– \*/**

**.top-banner {**

**color: var(--text\_light);**

**}**

**.heading {**

**font-weight: bold;**

**font-size: 4rem;**

**letter-spacing: 0.02em;**

**padding: 0 0 30px 0;**

**}**

**.top-banner form {**

**position: relative;**

**display: flex;**

**align-items: center;**

**}**

**.top-banner form input {**

**font-size: 2rem;**

**height: 40px;**

**padding: 5px 5px 10px;**

**border-bottom: 1px solid;**

**}**

**.top-banner form input::placeholder {**

**color: currentColor;**

**}**

**.top-banner form button {**

**font-size: 1rem;**

**font-weight: bold;**

**letter-spacing: 0.1em;**

**padding: 15px 20px;**

**margin-left: 15px;**

**border-radius: 5px;**

**background: var(--red);**

**transition: background 0.3s ease-in-out;**

**}**

**.top-banner form button:hover {**

**background: var(--darkred);**

**}**

**.top-banner form .msg {**

**position: absolute;**

**bottom: -40px;**

**left: 0;**

**max-width: 450px;**

**min-height: 40px;**

**}**

**/\* SECTION #2**

**–––––––––––––––––––––––––––––––––––––––––––––––––– \*/**

**.ajax-section {**

**margin: 70px 0 20px;**

**}**

**.ajax-section .cities {**

**display: grid;**

**grid-gap: 32px 20px;**

**grid-template-columns: repeat(4, 1fr);**

**}**

**.ajax-section .city {**

**position: relative;**

**padding: 40px 10%;**

**border-radius: 20px;**

**background: var(--text\_light);**

**color: var(--text\_med);**

**}**

**.ajax-section .city::after {**

**content: '';**

**width: 90%;**

**height: 50px;**

**position: absolute;**

**bottom: -12px;**

**left: 5%;**

**z-index: -1;**

**opacity: 0.3;**

**border-radius: 20px;**

**background: var(--text\_light);**

**}**

**.ajax-section figcaption {**

**margin-top: 10px;**

**text-transform: uppercase;**

**letter-spacing: 0.05em;**

**}**

**.ajax-section .city-temp {**

**font-size: 5rem;**

**font-weight: bold;**

**margin-top: 10px;**

**color: var(--text\_dark);**

**}**

**.ajax-section .city sup {**

**font-size: 0.5em;**

**}**

**.ajax-section .city-name sup {**

**padding: 0.2em 0.6em;**

**border-radius: 30px;**

**color: var(--text\_light);**

**background: var(--orange);**

**}**

**.ajax-section .city-icon {**

**margin-top: 10px;**

**width: 100px;**

**height: 100px;**

**}**

**/\* FOOTER**

**–––––––––––––––––––––––––––––––––––––––––––––––––– \*/**

**.page-footer {**

**text-align: right;**

**font-size: 1rem;**

**color: var(--text\_light);**

**margin-top: 40px;**

**}**

**.page-footer span {**

**color: var(--red);**

**}**

**/\* MQ**

**–––––––––––––––––––––––––––––––––––––––––––––––––– \*/**

**@media screen and (max-width: 1000px) {**

**body {**

**padding: 30px;**

**}**

**.ajax-section .cities {**

**grid-template-columns: repeat(3, 1fr);**

**}**

**}**

**@media screen and (max-width: 700px) {**

**.heading,**

**.ajax-section .city-temp {**

**font-size: 3rem;**

**}**

**.ajax-section {**

**margin-top: 20px;**

**}**

**.top-banner form {**

**flex-direction: column;**

**align-items: flex-start;**

**}**

**.top-banner form input,**

**.top-banner form button {**

**width: 100%;**

**}**

**.top-banner form button {**

**margin: 20px 0 0 0;**

**}**

**.top-banner form .msg {**

**position: static;**

**max-width: none;**

**min-height: 0;**

**margin-top: 10px;**

**}**

**.ajax-section .cities {**

**grid-template-columns: repeat(2, 1fr);**

**}**

**}**

**@media screen and (max-width: 500px) {**

**body {**

**padding: 15px;**

**}**

**.ajax-section .cities {**

**grid-template-columns: repeat(1, 1fr);**

**}**

**}**

**/\* API Key banner**

**–––––––––––––––––––––––––––––––––––––––––––––––––– \*/**

**.api {**

**background: #fffbbc;**

**position: fixed;**

**top: 0;**

**left: 0;**

**width: 100%;**

**padding: 10px;**

**}**

**.api a {**

**text-decoration: underline;**

**}**

**.api a:hover {**

**text-decoration: none;**

**}**

**Javascript code**

**/\*SEARCH BY USING A CITY NAME (e.g. athens) OR A COMMA-SEPARATED CITY NAME ALONG WITH THE COUNTRY CODE (e.g. athens,gr)\*/**

**const form = document.querySelector(".top-banner form");**

**const input = document.querySelector(".top-banner input");**

**const msg = document.querySelector(".top-banner .msg");**

**const list = document.querySelector(".ajax-section .cities");**

**/\*SUBSCRIBE HERE FOR API KEY: https://home.openweathermap.org/users/sign\_up\*/**

**const apiKey = "caa7a056213fc43a1ae5e3c0250b606a";**

**form.addEventListener("submit", e => {**

**e.preventDefault();**

**let inputVal = input.value;**

**//check if there's already a city**

**const listItems = list.querySelectorAll(".ajax-section .city");**

**const listItemsArray = Array.from(listItems);**

**if (listItemsArray.length > 0) {**

**const filteredArray = listItemsArray.filter(el => {**

**let content = "";**

**//athens,gr**

**if (inputVal.includes(",")) {**

**//athens,grrrrrr->invalid country code, so we keep only the first part of inputVal**

**if (inputVal.split(",")[1].length > 2) {**

**inputVal = inputVal.split(",")[0];**

**content = el**

**.querySelector(".city-name span")**

**.textContent.toLowerCase();**

**} else {**

**content = el.querySelector(".city-name").dataset.name.toLowerCase();**

**}**

**} else {**

**//athens**

**content = el.querySelector(".city-name span").textContent.toLowerCase();**

**}**

**return content == inputVal.toLowerCase();**

**});**

**if (filteredArray.length > 0) {**

**msg.textContent = `You already know the weather for ${**

**filteredArray[0].querySelector(".city-name span").textContent**

**} ...otherwise be more specific by providing the country code as well 😉`;**

**form.reset();**

**input.focus();**

**return;**

**}**

**}**

**//ajax here**

**const url = `https://api.openweathermap.org/data/2.5/weather?q=${inputVal}&appid=${apiKey}&units=metric`;**

**fetch(url)**

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

**.then(data => {**

**const { main, name, sys, weather } = data;**

**const icon = `https://s3-us-west-2.amazonaws.com/s.cdpn.io/162656/${**

**weather[0]["icon"]**

**}.svg`;**

**const li = document.createElement("li");**

**li.classList.add("city");**

**const markup = `**

**<h2 class="city-name" data-name="${name},${sys.country}">**

**<span>${name}</span>**

**<sup>${sys.country}</sup>**

**</h2>**

**<div class="city-temp">${Math.round(main.temp)}<sup>°C</sup></div>**

**<figure>**

**<img class="city-icon" src="${icon}" alt="${**

**weather[0]["description"]**

**}">**

**<figcaption>${weather[0]["description"]}</figcaption>**

**</figure>**

**`;**

**li.innerHTML = markup;**

**list.appendChild(li);**

**})**

**.catch(() => {**

**msg.textContent = "Please search for a valid city 😩";**

**});**

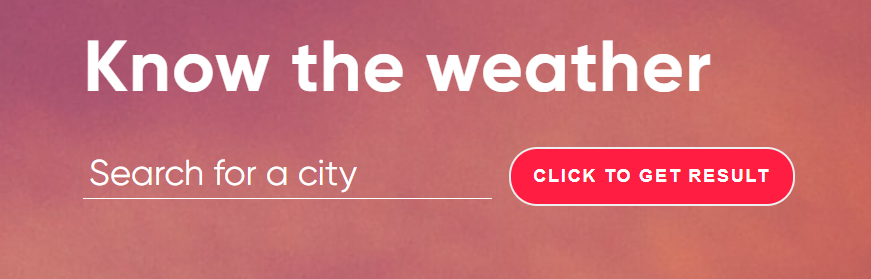
**msg.textContent = "";**

**form.reset();**

**input.focus();**

**});**

**APP SCREENSHOT**

**** 