

Create AJAX requests to the PokéAPI

In this tutorial, you will learn how to create an AJAX request to the PokéAPI.

On this page:

- [What is the PokéAPI?](#)
- [What is AJAX?](#)
- [Use AJAX for a web application](#)
- [Create an AJAX request to the PokéAPI](#)

What is the PokéAPI?

PokéAPI is a full RESTful API linked to an extensive database detailing everything about the Pokémon main game series. Authentication isn't required to use the PokéAPI and all resources are fully open and available. For more information, see [About · PokéAPI](#).

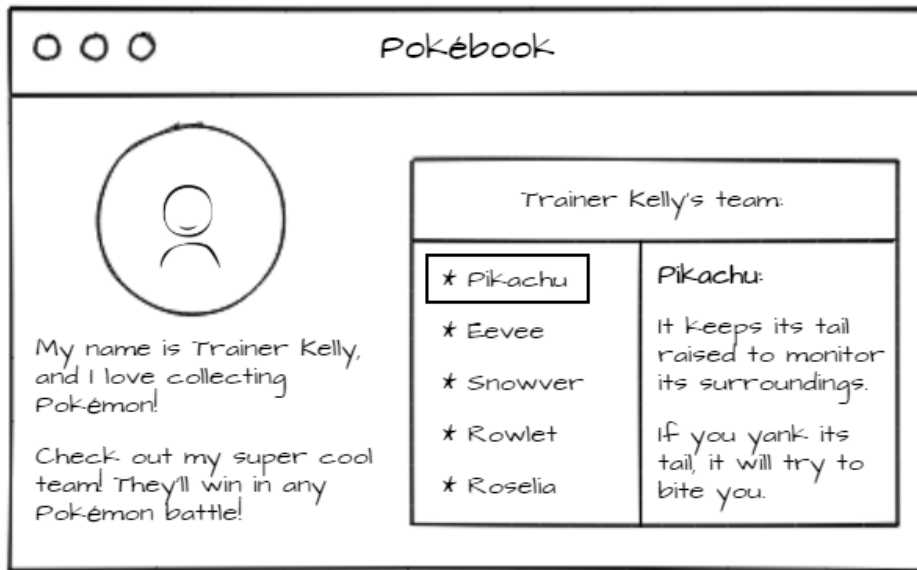
What is AJAX?

AJAX, which stands for Asynchronous JavaScript and XML, is a technique for sending and receiving data asynchronously. JavaScript and JavaScript code libraries, like jQuery, are synchronous—the code executes in the order it was written. In comparison, asynchronous code *does not* necessarily execute in order. When an application or web page uses AJAX, components update apart from each other. That is, the entire page or application does not need to update for any one part to update.

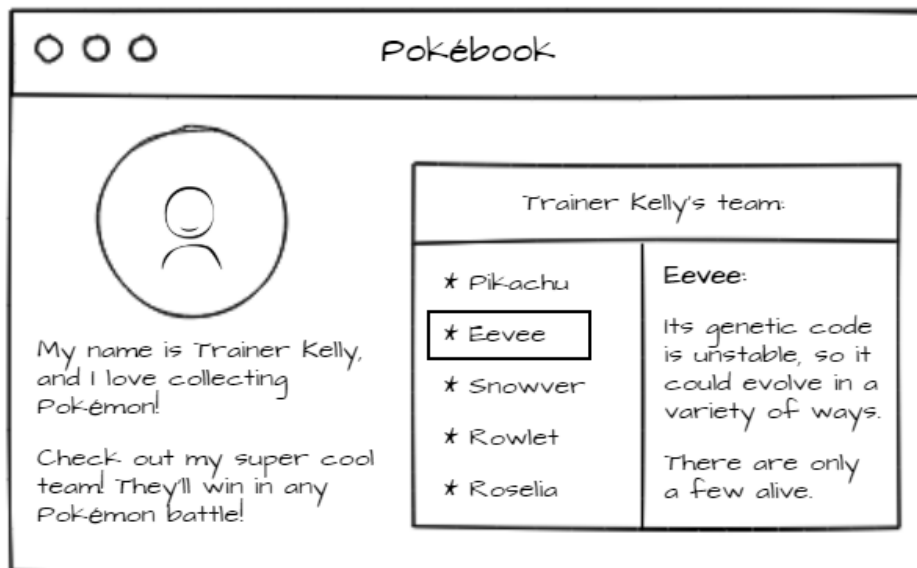
Use AJAX for a web application

If you were building a social networking website for Pokémon trainers, you could use AJAX to retrieve information about a trainer's Pokémon from the PokéAPI, and then dynamically display that information on a trainer's profile page. Moreover, while a component updates, your user can perform other actions on the site, allowing for a better overall user experience.

In the example below, the PokéAPI receives an AJAX request each time a user selects a Pokémon in Trainer Kelly's team.



An AJAX request is sent to the PokéAPI each time a user selects a Pokémon



Create an AJAX request to the PokéAPI

This example uses CodePen. CodePen is an online community for testing and showcasing HTML, CSS and JavaScript code snippets.

To create an AJAX request to the PokéAPI with CodePen, complete the following steps.

1. [Create a new code snippet](#).
2. In the **HTML** pane, add the following HTML code.

```
<script src="https://code.jquery.com/jquery-2.1.1.min.js"></script>

<h1>Trainer information</h1>

<h2>About me</h2>
<p>I'm a Pokémon trainer, and I love collecting Pokémon!</p>
<p>Check out my super cool team! They'll win in any Pokémon battle!</p>

<h2>My Pokémon</h2>
<ul>
  <li><button type="button" id="pButton">Pikachu</button></li>
  <li><button type="button" id="eButton">Eevee</button></li>
</ul>

<h2>Pokémon information</h2>
<p class="PokeInfo">
  {jQuery replaces this text when you create an AJAX request!}
</p>
```

3. In the **JavaScript** pane, add the following JavaScript code.

```
$('#pButton').click(function() {
  $('.text').text('loading...');
  $.ajax({
    type:"GET",
    url:"https://pokeapi.co/api/v2/pokemon-species/pikachu/",
    success: function(data) {
      var pokeDescription =
```

```

    JSON.stringify(data.flavor_text_entries[2].flavor_text);
var prettyDescription =
    pokeDescription.replace(/\n/g, ' ');
$('.PokeInfo').text(prettyDescription);
},
});
});

$('#eButton').click(function() {
    $('.text').text('loading...');
$.ajax({
    type:"GET",
    url:"https://pokeapi.co/api/v2/pokemon-species/eevee/",
    success: function(data) {
        var pokeDescription =
            JSON.stringify(data.flavor_text_entries[2].flavor_text);
        var prettyDescription =
            pokeDescription.replace(/\n/g, ' ');
        $('.PokeInfo').text(prettyDescription);
    },
    });
});
});

```

4. In the My Pokémon section of the rendered HTML, click **Pikachu** or **Eevee** to create an AJAX request and learn more about that Pokémon! For more information on how CodePen's JavaScript engine executes the above code, read below.
 1. Each time you click **Pikachu** or **Eevee**, the code creates an AJAX request to the PokéAPI, specifically to the `/pokemon-species` resource.
 1. For more information on `/pokemon-species`, see [Pokémon species](#).
 2. After the PokéAPI sends a JSON response, jQuery manipulates the page by replacing the text in `PokeInfo` with the text value of `flavor_text_entries[2].flavor_text`.
 1. **Note:** The code modifies the raw text value of `flavor_text_entries[2].flavor_text` for a prettier text output.