

BACK TO TRACKS

1. Using APIs and Ajax

45%

(/m/126/6265)

API

(/m/

Useful URLs

(/m/

Ajax

(/m/

Pokemon

(/m/

Traversing JSON

(/m/

Building HTML

(/m/

Quiz

(/m/

What's that Pokemon? Build...

(/m/

API Key

(/m/

Your Very Own Weather For...

(/m/

Game of Thrones (Optional)

(/m/

Chapter Survey

Construyendo HTML

¿Qué pasaría si quisiéramos mostrar los datos JSON en nuestro sitio web en lugar de simplemente en la consola? Podemos incluir los datos del objeto JSON en nuestro HTML para que los usuarios de diferentes tipos de Bulbasaur son veneno y hierba.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Gotta Catch'em All</title>
    <script src="https://code.jquery.com/jquery-2.1.3.min.js"></script>
    <script>
      $(document).ready(function() {
        $.get("https://pokeapi.co/api/v2/pokemon/1/", function(res) {
          var html_str = "";
          html_str += "<h4>Types</h4>";
          html_str += "<ul>";
          for(var i = 0; i < res.types.length; i++) {
            html_str += "<li>" + res.types[i].type.name + "</li>";
          }
          html_str += "</ul>";
          $("#bulbasaur").html(html_str);
        }, "json");
      })
    </script>
  </head>
  <body>
    <div id="bulbasaur">
    </div>
  </body>
</html>
```

Estamos pasando tres argumentos a la función `$.get`.

1. La URL desde la que queremos cargar información (es decir, "https://pokeapi.co/api/v2/pokemon/1/").
2. Función que queremos ejecutar después de que la información se haya cargado correctamente.
 - Queremos construir una cadena que se vea como "<h4> Tipos </h4> veneno hierba ". El navegador representará como HTML cuando llamemos a `$("#bulbasaur").html(html_str)`.
3. El tipo de información que estamos esperando, que en nuestro caso es `json`.