

Artículo

[Nancy Martínez](#) · Sep 13, 2019 Lectura de 5 min

Vue.js: Primeros pasos con un ejemplo básico de HTML/REST/JSON

¡ Hola Comunidad!

Este es un ejemplo básico de [Vue.js](#) en JavaScript, sobre cómo es posible usar llamadas REST con HTML básico.

Simplemente, podéis copiar/pegar el siguiente código de ejemplo, guardarlo en un archivo *.html y abrirlo en el navegador:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Cach&eacute; & amp; Vue.js news</title>
    <script src="https://unpkg.com/vue"></script>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/foundation/6.3.1/css/foundation.min.css">
  </head>
  <body>
    <div class="container" id="app">
      <h3 class="text-center">Cach&eacute; & amp; Vue.js news</h3>
      <div class="columns medium-4" v-for="post in posts">
        <div class="card">
          <div class="card-divider">
            {{ post.title }}
          </div>
          <div class="card-section">
            <p>{{ post.body }}.</p>
          </div>
        </div>
      </div>
    </div>
  </body>
  <script>
    var vm = new Vue({
      el: '#app',
      data: {
        posts: [
          {title: "My very first Vue.js post", body: "lorem ipsum some test dimpsum"},
          {title: "My second Vue.js post", body: "lorem ipsum some test dimpsum"},
          {title: "My third Vue.js post", body: "lorem ipsum some test dimpsum"}
        ]
      }
    });
  </script>
</html>
```

Ahora mejoraremos este ejemplo levemente para obtener las publicaciones mediante una llamada a un endpoint

(de prueba) REST (usando la librería de ayuda de Axios):

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Cach&eacute; & #amp; # Vue.js news</title>
    <script src="https://unpkg.com/vue"></script>
    <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/foundation/6.3.1/css/foundation.min.css">
  </head>
  <body>
    <div class="container" id="app">
      <h3 class="text-center">Cach&eacute; & #amp; # Vue.js news</h3>
      <div class="columns medium-3" v-for="post in posts">
        <div class="card">
          <div class="card-divider">
            {{ post.title }}
          </div>
          <div class="card-section">
            <p>{{ post.body }}.</p>
          </div>
        </div>
      </div>
      <hr>
      <button class="button" style="display:block;margin:auto;" @click="getPostsViaREST">Get posts via REST</button>
    </div>
  </body>
  <script>
    var vm = new Vue({
      el: '#app',
      data: {
        posts: [
          {title: "My very first Vue.js post", body: "lorem ipsum some test dimpsum"},
          {title: "My second Vue.js post", body: "lorem ipsum some test dimpsum"},
          {title: "My third Vue.js post", body: "lorem ipsum some test dimpsum"}
        ]
      },
      methods: {
        getPostsViaREST: function() {
          axios.get("https://jsonplaceholder.typicode.com/posts")
            .then(response => {this.posts = response.data})
        }
      }
    });
  </script>
</html>
```

Como habréis notado, agregamos un par de cosas: la librería de ayuda de Axios en el <head>, un <button> para recuperar las publicaciones mediante REST (prestar atención a la sintaxis natural en Vue.js para vincular métodos a métodos de eventos) y luego en la instancia Vue, un método getPostsViaREST en el que se usa Axios para realizar la llamada a REST. La respuesta de REST modifica la propiedad (reactiva) data.posts. Se debe tener en cuenta que Vue.js actualiza la vista automáticamente apenas se modifique el contenido de data.posts, ¡bienvenido a la "reactividad"!

Es muy fácil sustituir este endpoint de prueba con vuestros propios endpoints QEWDjs/REST o CSP/REST Caché o, aún mejor, como se muestra en el [ejemplo de vue-qewd](#), con una conexión permanente de WebSocket a un backend QEWDjs/Caché.

Este ejemplo es solo una introducción muy básica a las capacidades de Vue.js. Para una aplicación real, seguramente utilizaréis una forma mas avanzada de programación basada en componentes utilizando módulos Node.js propios con archivos .vue.

Si queréis aprender sobre Vue.js desde cero, os recomiendo [este excelente curso](#) de Maximilian Schwarzmüller.

[#API REST](#) [#HTML](#) [#JavaScript](#) [#JSON](#) [#Vue.js](#) [#ZEN](#) [#Caché](#)

URL de fuente: <https://es.community.intersystems.com/post/vuejs-primeros-pasos-con-un-ejemplo-b%C3%A1sico-de-htmlrestjson>