

¿Cómo hacer una integración API?

Ahora que comprendemos mejor qué son las API, veamos el proceso de integración de una API real y hagamos nuestras primeras solicitudes. Empezaremos de forma sencilla con una [API de broma](#).

Primero, dirígete a esta [URL](#).

Toda la documentación se encuentra en este archivo [README.md](#).

Official Joke API

Endpoints:

Grab a random joke!

https://official-joke-api.appspot.com/random_joke

<https://official-joke-api.appspot.com/jokes/random>

Grab ten random jokes!

https://official-joke-api.appspot.com/random_ten

<https://official-joke-api.appspot.com/jokes/ten>

Grab jokes by type!

The endpoints are `jokes/:type/random` or `jokes/:type/ten`. For example:

<https://official-joke-api.appspot.com/jokes/programming/random>

<https://official-joke-api.appspot.com/jokes/programming/ten>

Al mirar la documentación, puedo ver que tenemos tres puntos finales.

Si queremos "captar un chiste al azar", se nos ofrecen dos sintaxis posibles para ello. No hay nada intrínsecamente diferente en esos dos vínculos; El autor de la API le ofrece dos formas de abordar el uso de la API.

Con esta API, puedes visitar la URL en tu navegador y verás la respuesta.

Consejo de desarrollo: Firefox es excelente para formatear los datos de una manera fácil de leer.



A cambio de nuestra solicitud, recibimos una carga útil JSON con cuatro propiedades: la **identificación** de este chiste aleatorio, su **tipo**, la **configuración** y el **remate** del chiste.

Una cosa que me gusta de esta API de chistes es que es relativamente simple y de código abierto. Nos permite ver el código que devuelve nuestros chistes.

Todos los chistes se almacenan en un archivo JSON [aquí](#). Cuando realizamos nuestra solicitud GET, [index.js](#) es lo que maneja nuestra solicitud llamando a la función apropiada. Las funciones se almacenan [aquí en handler.js](#) y solo hay unas pocas funciones.

Recomiendo echar un vistazo a esos tres archivos, incluso si no comprende completamente lo que están haciendo. Verás que las API no tienen por qué ser complicadas. Aquí la 'base de datos' es un único archivo JSON.

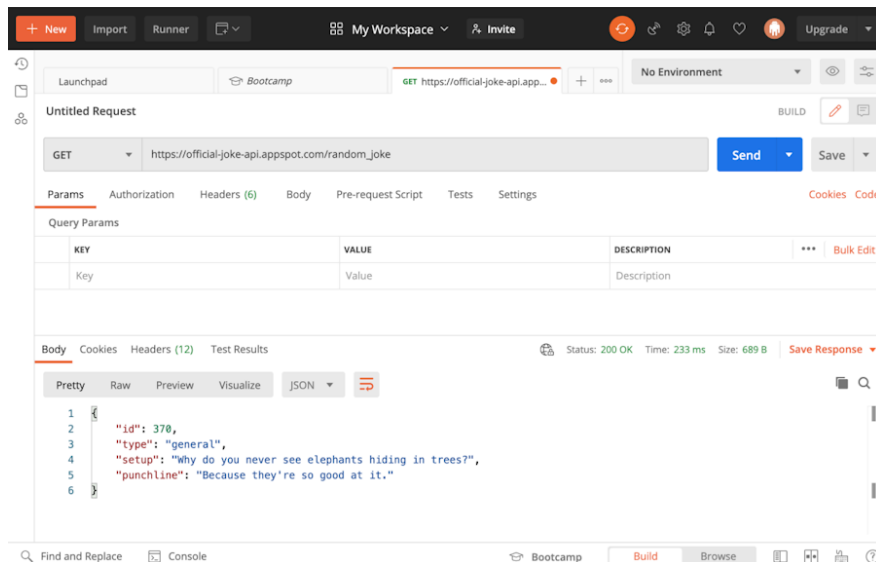
Usando una herramienta API - Postman

Cuando realizamos solicitudes de API, generalmente no es escribiéndolas en un navegador sino a través de código. Puede llevar mucho tiempo codificar las solicitudes de API cuando solo intentas probar si una API funciona. Por suerte existe un programa gratuito llamado Postman que puedes utilizar online.

Postman es un programa sólido en el que no profundizaré mucho, pero quiero que te sientas cómodo creando una solicitud GET con él.

El verbo de acción HTTP predeterminado es **GET**, por lo que puede dejarlo y pegarlo **https://official-joke-api.appspot.com/random_joke** como URL de solicitud.

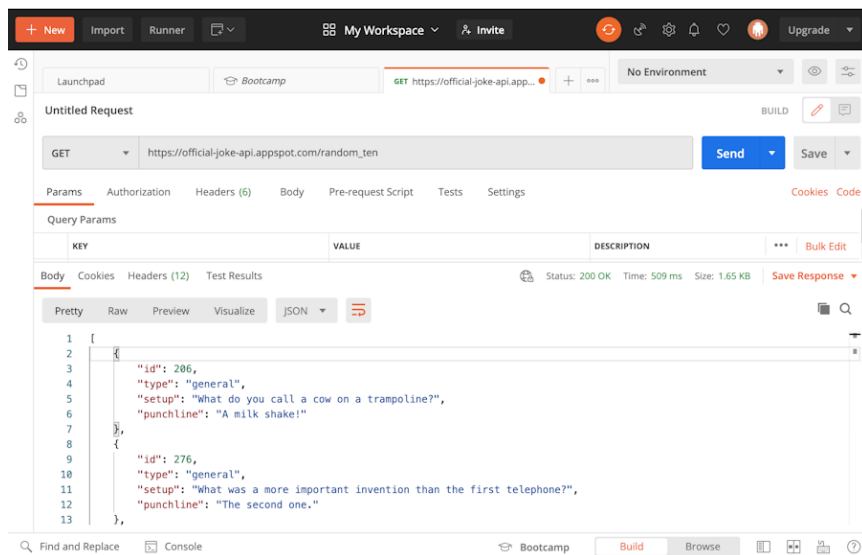
Haga clic en Enviar para enviar su solicitud y verá su respuesta en el panel inferior.



¡Eso es todo! Obtienes una gran cantidad de información fácilmente accesible con Postman. Puede ver el estado, **200 OK** el tiempo que tardó en finalizar la solicitud y mucho más si navega por las diferentes pestañas.

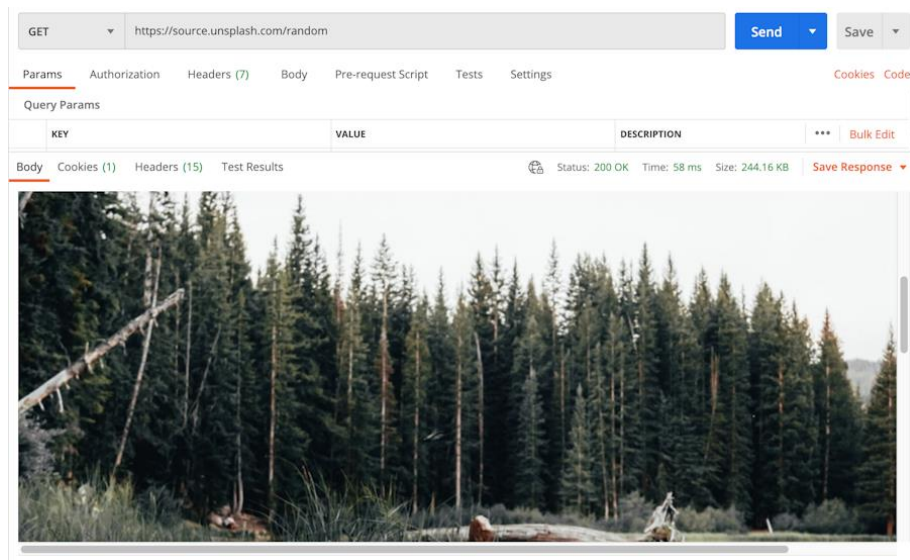
Ahora acceda a este punto final en Postman: **`https://official-joke-api.appspot.com/random_ten`**

Ahora solicitamos una serie de diez objetos de chiste, por lo que la forma de la respuesta ha cambiado.



Observe que el cuerpo de la respuesta ahora comienza con corchetes, `[]` en lugar de corchetes, `{ }`.

Algunas API, como la API Unsplash, devuelven una imagen real para la carga útil de respuesta. Pruebe este punto final y vea lo que obtiene: **`https://source.unsplash.com/random`**



Familiarizarse con Postman le ayudará a seguir utilizando las API y algún día crear las suyas propias.