

# RestAPIs

## Acerca de mí



**Cristian Buitrago**  
Software Engineer

- **Software Engineer** who loves working with Javascript and System Design .
- Anime Lover.
- **Education Enthusiast.**

Github @buty619

## Puntos importantes



Identifícate en Zoom utilizando tu nombre y apellido.



Mantén tu micrófono apagado durante el transcurso de la sesión.



Utiliza el chat para hacer tus preguntas durante la sección de Q&A.



Procura enfocar tus preguntas al tema presentado.



Apaga tu cámara en caso de tener problemas con tu conexión.

## Código de conducta



Sé respetuoso, no hay preguntas o ideas malas.



Sé cordial y paciente.



Sé cuidadoso con tus palabras.

## Objetivos de la sesión

**Al final de la sesión serás capaz de:**

- Entender que es una RestAPI.
- Diferenciar una RestAPI de una SOAP.
- Entender JSON.
- Entender Postman y cómo usarlo.



# Contenido del curso

**1**

## Rest vs SOAP

¿Cómo se diferencian?

**2**

## Metodos HTTP

¿Cuáles son los principales métodos HTTP?

**3**

## JSON

¿Qué es json?

**4**

## Postman

¿Qué es postman y qué características y ventajas ofrece?



# Rest vs SOAP

¿Cómo se diferencian?

# SOAP



# VS



# REST



## SOAP

*SOAP* es un protocolo estándar que se creó originalmente para posibilitar la comunicación entre las aplicaciones que se diseñan con diferentes lenguajes y en distintas plataformas. Impone **reglas integradas** que aumentan la complejidad y la sobrecarga.

## REST

*REST* es un conjunto de principios arquitectónicos que se ajusta a las necesidades de las aplicaciones móviles y los servicios web ligeros. Es un **conjunto de pautas** donde la implementación de las recomendaciones depende de los desarrolladores.

## SOAP

- Centrada en los servicios
- Estandarizado
- Requiere mayor ancho de banda
- **Respuesta:** XML
- **Pruebas:** Requiere software especializado
- **Método:** POST

## REST

- Centrada en datos
- Flexible
- Liviano
- **Respuesta:** Diferentes tipos de respuesta
- **Pruebas:** En cualquier software que soporte HTTP
- **Método:** GET

## Pautas Arquitectonicas REST

Se considera que una aplicación es *RESTful* si cumple con ciertas pautas arquitectónicas. Una aplicación de *RESTful* debe tener lo siguiente:

- Arquitectura cliente-servidor
- Comunicación cliente-servidor sin estado
- Datos que pueden almacenarse en caché
- Interfaz uniforme (más detalles a continuación)
- Restricción del sistema en capas

## Interfaz Uniforme

Una interfaz uniforme es útil para:

- Identificar de recursos en las peticiones
- Manipular de recursos a través de representaciones
- Mandar mensajes auto-descriptivos
- Usar hipermedia como motor del estado de la aplicación

# Contenido del curso

**1**

## Rest vs SOAP

¿Cómo se diferencian?

**2**

## Metodos HTTP

¿Cuáles son los principales métodos HTTP?

**3**

## JSON

¿Qué es json?

**4**

## Postman

¿Qué es postman y qué características y ventajas ofrece?



# Metodos HTTP

¿Cuáles son los principales métodos HTTP?

# Metodos HTTP

**HTTP** define un conjunto de métodos de petición para indicar la acción que se desea realizar para un recurso determinado.

Method	Original Purpose	RESTful Purpose
GET	Request a file.	Request an object.
HEAD	Request a file, but only return the HTTP headers.	
POST	Submit data.	
PUT	Upload a file.	Create an object.
DELETE	Delete a file	Delete an object.
CONNECT	Establish a connection to another system.	
OPTIONS	List supported HTTP methods.	Perform a <a href="#">CORS Preflight</a> request.
TRACE	Echo the HTTP request for debug purposes.	
PATCH		Modify an object.

# Contenido del curso

1

## Rest vs SOAP

¿Cómo se diferencian?

2

## Metodos HTTP

¿Cuáles son los principales métodos HTTP?

3

## JSON

¿Qué es json?

4

## Postman

¿Qué es postman y qué características y ventajas ofrece?



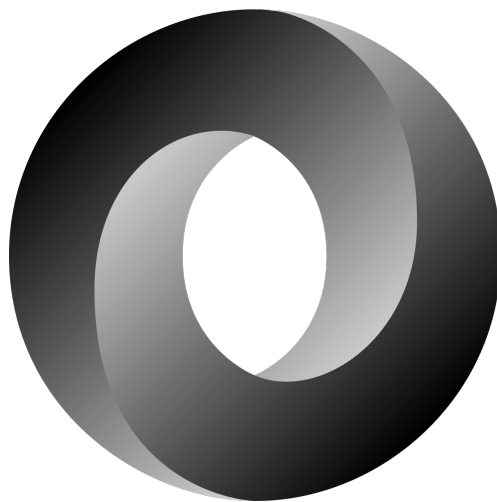


# JSON

¿Qué es json?

# JSON

JSON (*JavaScript Object Notation - Notación de Objetos de JavaScript*) es un formato ligero de intercambio de datos. Está basado en un subconjunto del **Lenguaje de Programación JavaScript**. JSON es un formato de texto independiente del lenguaje pero utiliza convenciones que son ampliamente conocidas por los programadores lo cual hace que sea un lenguaje ideal para el intercambio de datos.



# Contenido del curso

1

## Rest vs SOAP

¿Cómo se diferencian?

2

## Metodos HTTP

¿Cuáles son los principales métodos HTTP?

3

## JSON

¿Qué es json?

4

## Postman

¿Qué es postman y qué características y ventajas ofrece?



# Postman

¿Qué es postman, qué características y ventajas ofrece?



cURL es una herramienta de línea de comandos que nos permite ejecutar llamadas HTTP a recursos en línea o locales. Es sencillo de usar pero también implica ciertos desafíos, como la sencillez de lectura de las respuestas.

**\*\*ejemplos cURL, sin headers, con headers, HTTP, cambios de status code\*\***

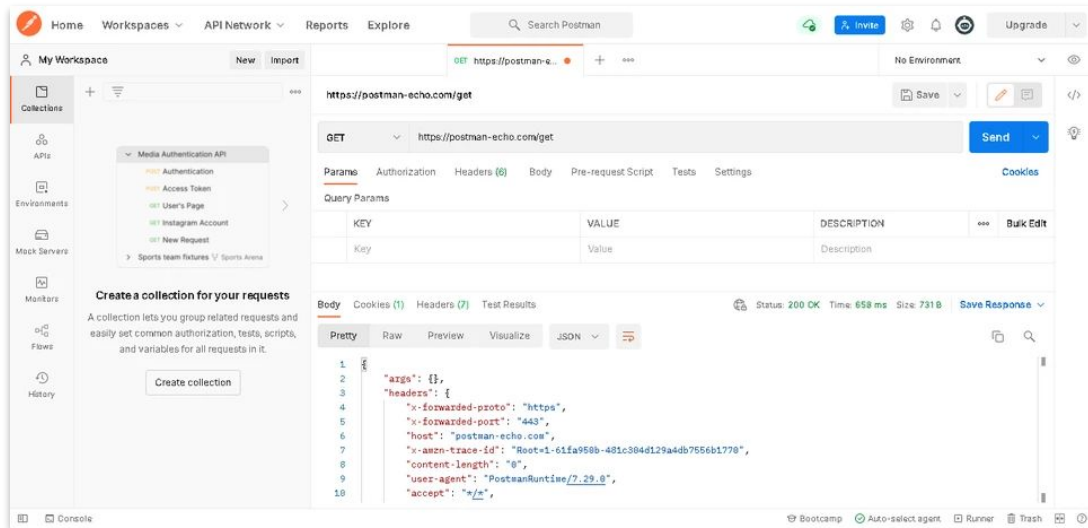
## ¿Qué es Postman?

**Postman** es una plataforma de desarrollo de APIs enfocada en simplificar el ciclo de vida de las API y mejorar la colaboración con el equipo. Lo usamos habitualmente para probar nuestras API.



POSTMAN

# Manos a la obra



# Crear un ambiente en Postman

The screenshot shows the Postman interface with the 'Environments' tab selected in the left sidebar. The main panel displays a GET request with a query parameter 'somevariable' set to '{{parameter\_value}}'. The 'Query Params' table shows the key 'somevariable' with a checked checkbox and the value '{{parameter\_value}}'.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> somevariable	{{parameter_value}}	
Key	Value	Description



# Variables en Postman

The screenshot displays the Postman web interface. The top navigation bar includes 'Home', 'Workspaces', 'API Network', and 'Explore'. A search bar labeled 'Search Postman' is present. On the right, there are buttons for 'Invite', 'Upgrade', and a dropdown menu. The left sidebar shows the 'My Workspace' section with 'New' and 'Import' buttons. Below this, there are icons for 'Collections', 'APIs', 'Environments' (highlighted), 'Mock Servers', 'Monitors', 'Flows', and 'History'. The 'Environments' panel shows a list of environments: 'Globals' and 'Dev' (selected). The 'Dev' environment is active, and its variables are displayed in a table.

VARIABLE	TYPE ①	INITIAL VALUE ①	CURRENT VALUE ①	...	Persist All	Reset All
<input checked="" type="checkbox"/> base_url	default ▾	httpbin.org/	httpbin.org/			
<input checked="" type="checkbox"/> parameter_value	default ▾	50000	50000			
<input checked="" type="checkbox"/> json_value	default ▾	This is a json text value	This is a json text value			
Add a new variable						

# Content types

*Content-Type* es la propiedad de cabecera (*header*) usada para indicar el media type (en-US) del recurso. *Content-Type* dice al cliente qué tipo de contenido será retornado. En solicitudes como **POST** o **PUT**, el cliente indica al servidor qué tipo de dato es enviado actualmente.


GET <https://images.ctfassets.net/yad1kx9rmg0/wtrtHxeu3zEoEe2MokCSi/cf6f68efdcf625fd060607df0f3baef/qwwowoybuqbl6ntbox3.jpg> Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Headers [Hide auto-generated headers](#)

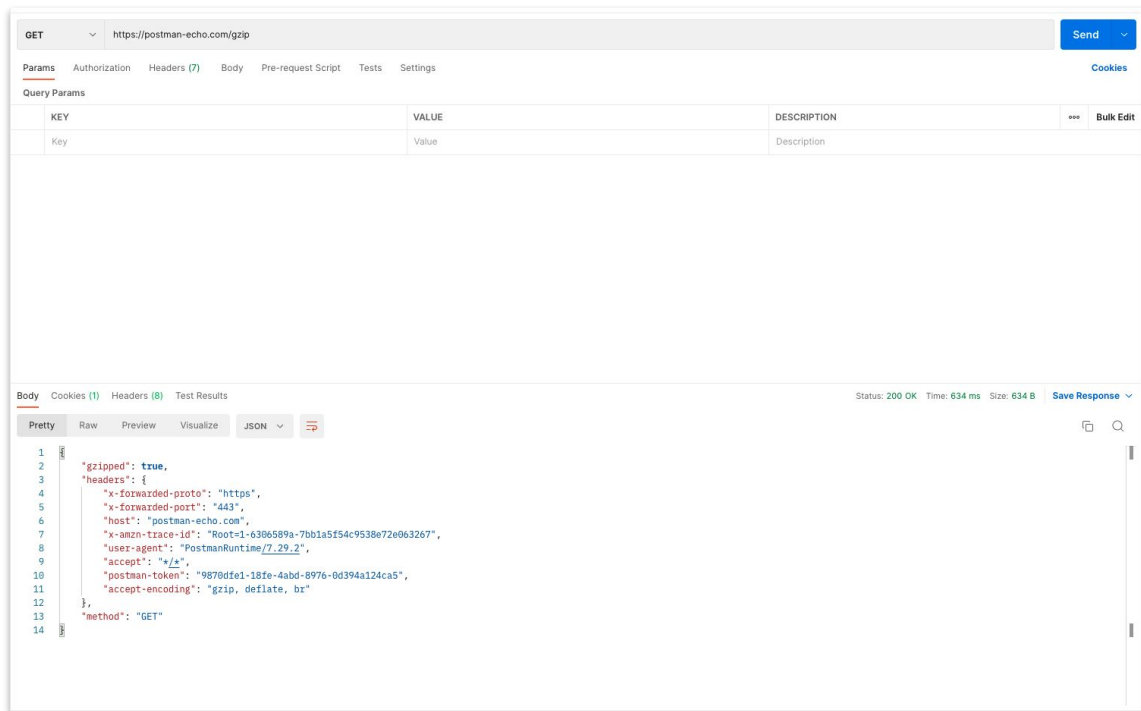
KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Postman-Token	<calculated when request is sent>				
<input checked="" type="checkbox"/> Host	<calculated when request is sent>				
<input checked="" type="checkbox"/> User-Agent	PostmanRuntime/7.29.2				
<input checked="" type="checkbox"/> Accept	*/*				
<input checked="" type="checkbox"/> Accept-Encoding	gzip, deflate, br				
<input checked="" type="checkbox"/> Connection	keep-alive				
Key	Value	Description			

Body Cookies Headers (14) Test Results Status: 200 OK Time: 122 ms Size: 27.06 KB Save Response



# Respuestas comprimidas

Postman es capaz de manejar respuestas que se encuentran comprimidas con algún algoritmo, más comúnmente **gzip**. Esta operación se realiza automáticamente.

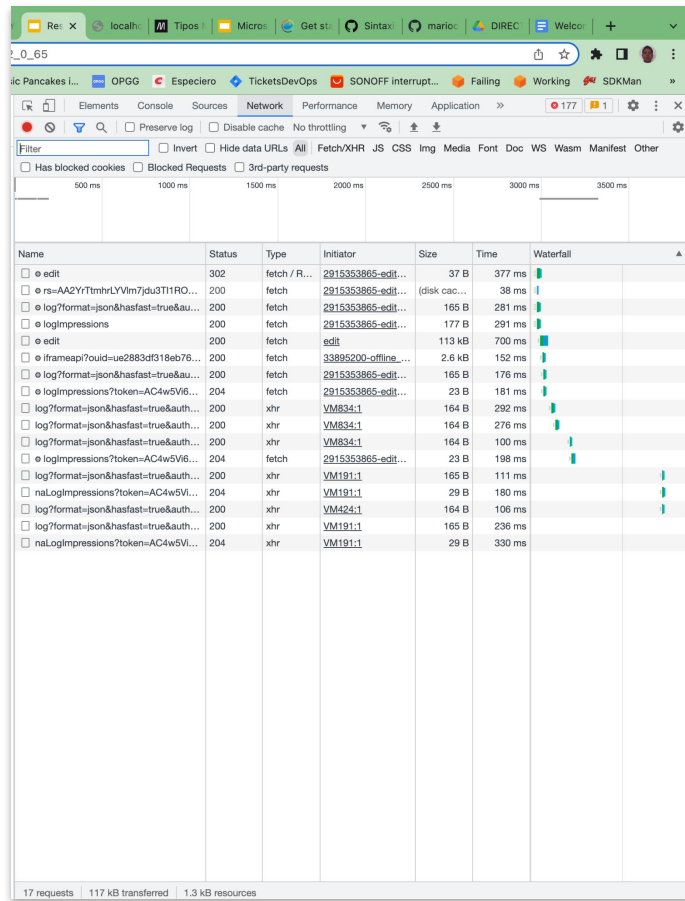




## Importar llamadas curl a Postman

**Postman** puede importar llamadas cURL a su interfaz mediante muy sencillos pasos. Esto es importante cuando rastreamos algún problema que es visible en el navegador y sospechamos que es responsabilidad de los servicios que sirven dicha información.

**Requisitos:** Para comenzar, en **Google Chrome**, abre la pestaña **Red** de las herramientas de desarrollador.

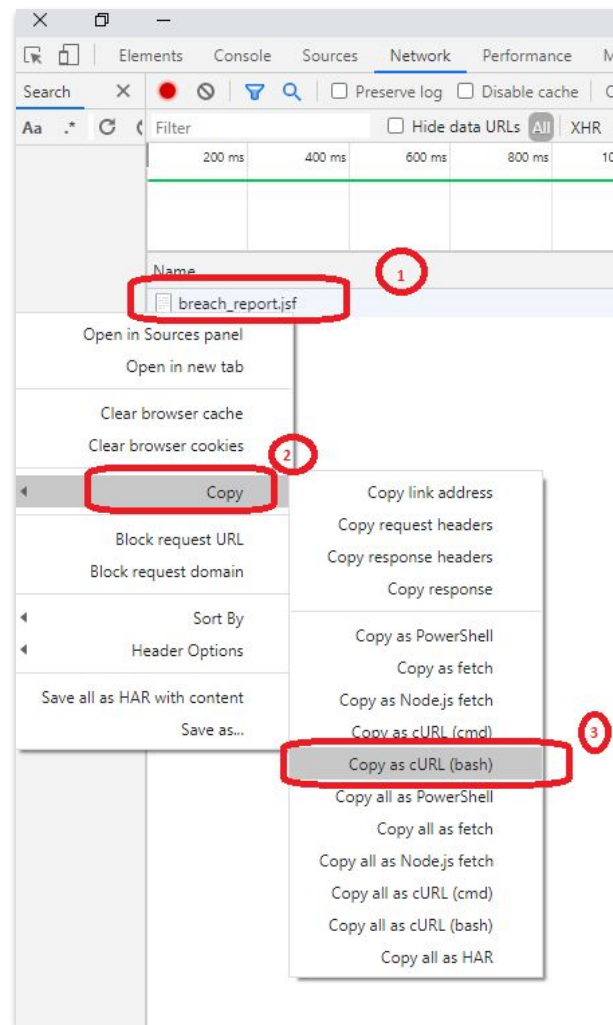




# Importar llamadas cURL a Postman

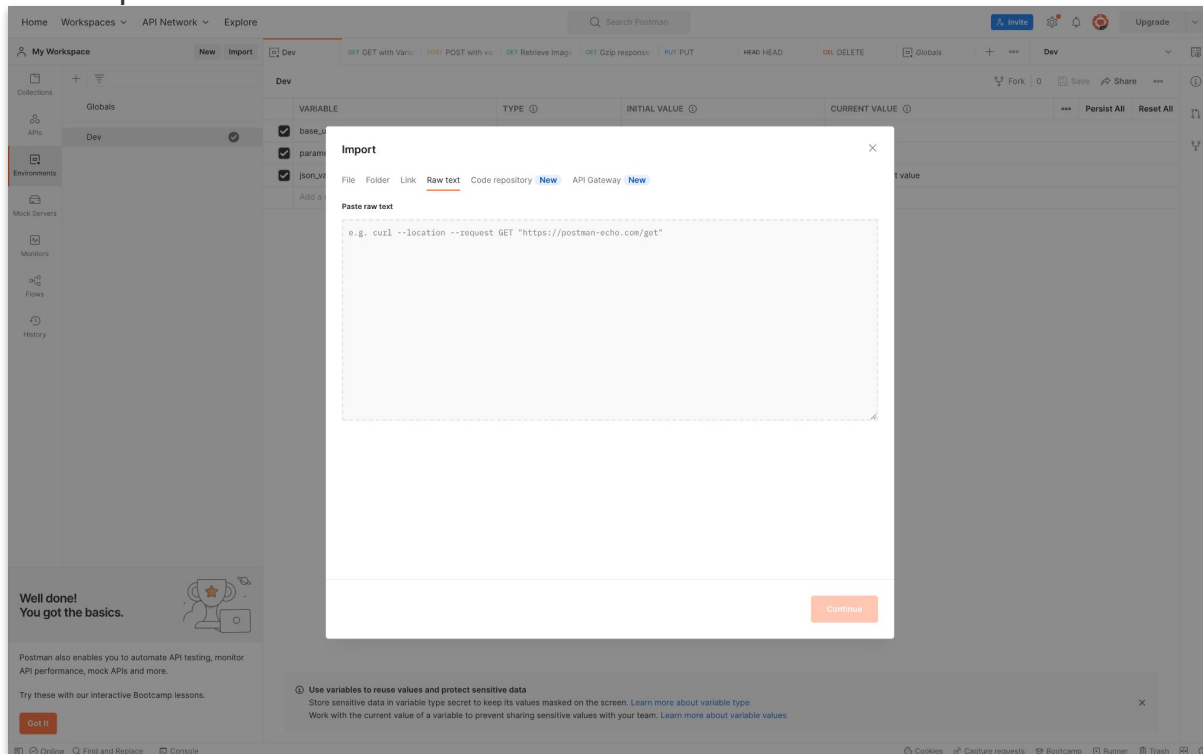
Para importar llamadas cURL a Postman:

1. Haz click secundario a la llamada que vas a importar a Postman
2. Selecciona **Copy as cURL (bash)**.



# Importar llamadas curl a Postman

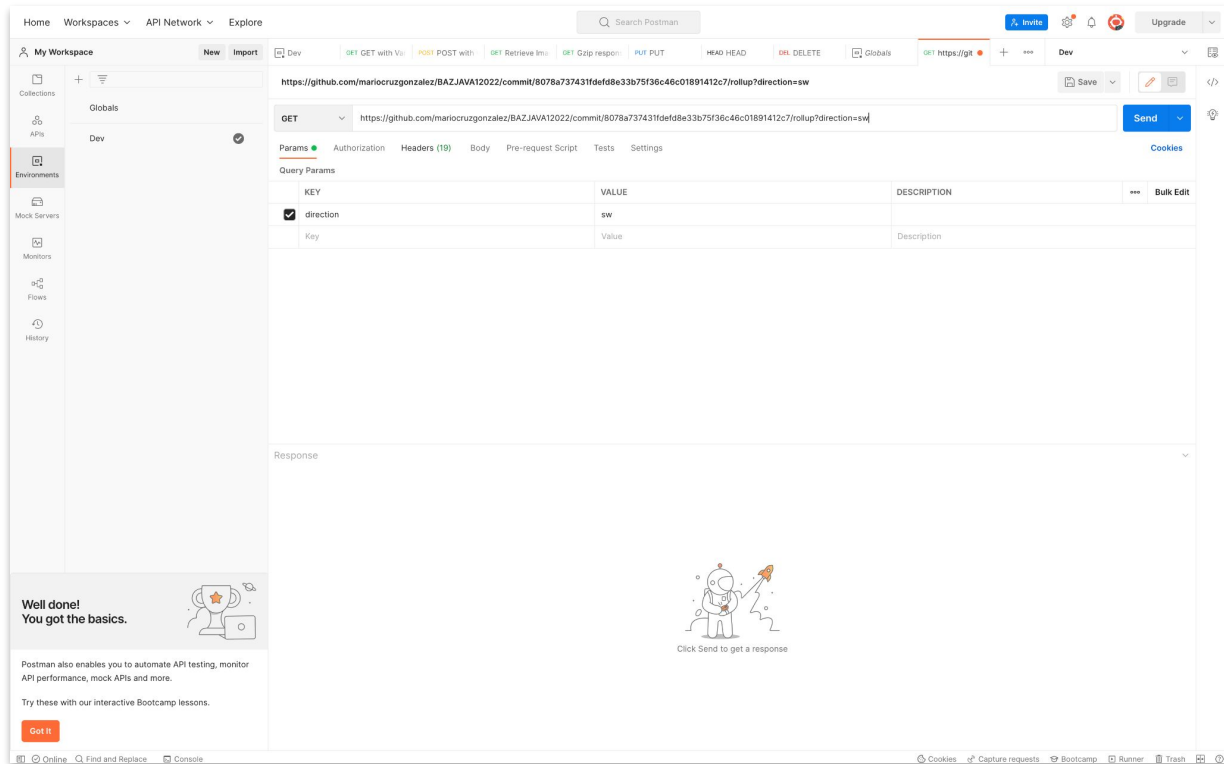
3. En Postman, haz click en **Import**.
4. Selecciona la opción de **Raw Text**.



# Importar llamadas curl a Postman

5. Haz click en **Continuar**.

6. Haz click en **Import** y obtendrás la llamada en Postman.

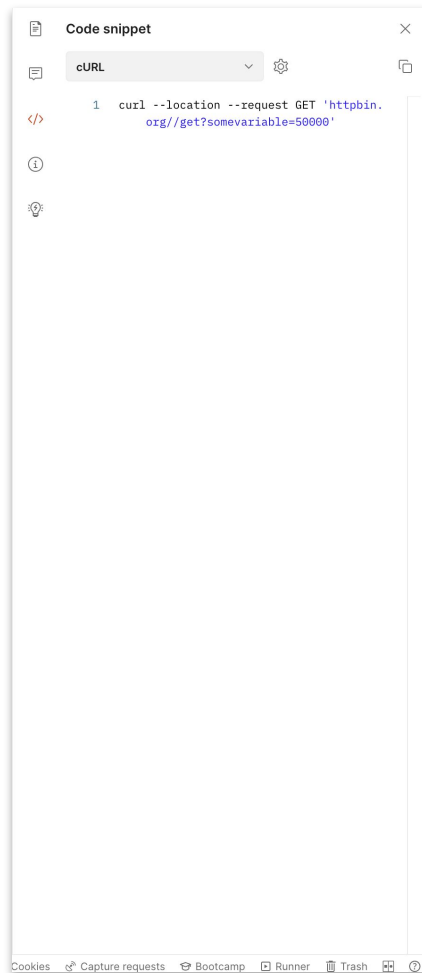




## Llamadas a código

Se hace de la misma forma en Postman al tener una llamada. Por ejemplo la parte de una colección ajena, podemos exportar dicha llamada a código, para este ejemplo, **cURL**:

1. Haz click en Postman a la derecha, en la opción **code**.
2. Elige el lenguaje en el que queremos que se exporte la llamada, por ejemplo, **cURL**.
3. Ejecuta el código en terminal o impórtalo a nuestros proyectos en código.





Queremos conocer tu opinión



[HTTPS://forms.gle/WKtc8wZeSxWnjGo8A](https://forms.gle/WKtc8wZeSxWnjGo8A)



Gracias