



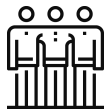
RestAPIs

Mario Alberto Cruz Gonzalez

Academy Code of Conduct



Seamos respetuosos, no existen preguntas malas



Seamos pacientes



Cuidemos nuestro lenguaje

Objetivos de la sesión

Al final de la sesión seremos capaces de:

- Entender que es una RestAPI
- Como se diferencia una RestAPI de una SOAP
- Qué es JSON.
- Qué es postman y como lo podemos usar.

Tabla de contenido

Rest vs SOAP

¿Cómo se diferencian?



Metodos HTTP

¿Cuáles son los principales métodos HTTP?



JSON

¿Qué es json?



Postman

¿Qué es postman y qué características y ventajas ofrece?



SOAP



VS



REST

REST

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

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

- Centrada en datos
- Flexible
- Diferentes tipos de respuesta
- Liviano
- Se puede probar en cualquier software que soporte http
- *GET*

SOAP

- Centrada en los servicios
- Estandarizado
- XML
- Requiere mayor ancho de banda
- Requiere software especializado para probarse
- *POST*

Pautas Arquitectonicas REST

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

- Una arquitectura cliente-servidor
- Una comunicación cliente-servidor sin estado
- Datos que pueden almacenarse en caché
- Una interfaz uniforme (más detalles a continuación)
- Una restricción del sistema en capas

Interfaz Uniforme

- Identificación de recursos en las peticiones
- Manipulación de recursos a través de representaciones
- Mensajes auto-descriptivos
- Hipermedia como motor del estado de la aplicación

Tabla de contenido

Rest vs SOAP

¿Cómo se diferencian?



Metodos HTTP

¿Cuáles son los principales métodos HTTP?



JSON

¿Qué es json?



Postman

¿Qué es postman y qué características y ventajas ofrece?



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 CORS Preflight request.
TRACE	Echo the HTTP request for debug purposes.	
PATCH		Modify an object.

Tabla de contenido

Rest vs SOAP

¿Cómo se diferencian?



Metodos HTTP

¿Cuáles son los principales métodos HTTP?



JSON

¿Qué es json?



Postman

¿Qué es postman y qué características y ventajas ofrece?



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 que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidas por los programadores. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

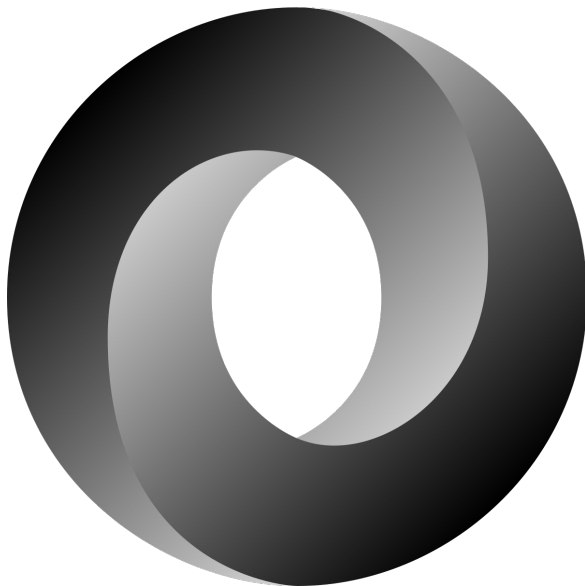


Tabla de contenido

Rest vs SOAP

¿Cómo se diferencian?



Metodos HTTP

¿Cuáles son los principales métodos HTTP?



JSON

¿Qué es json?



Postman

¿Qué es postman y qué características y ventajas ofrece?



curl

Curl es una herramienta de línea de comandos que nos permite ejecutar llamadas http a recursos en línea o locales, su uso es bastante sencillo pero también implica ciertos desafíos, como la sencillez de lectura de las respuestas.

****ejemplos curl, sin headers, con headers, http, cambios de status code****

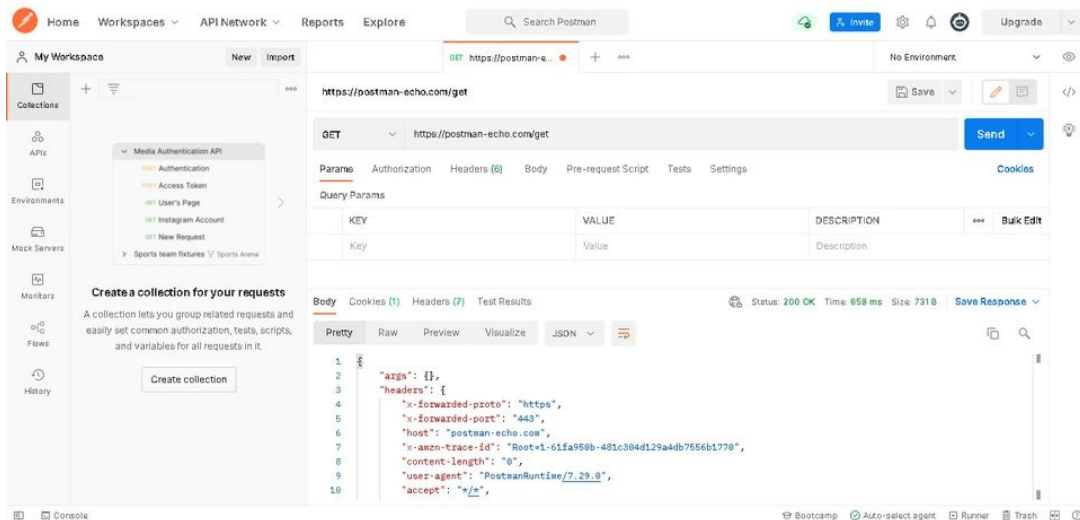
¿Qué es Postman?

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. Para lo que lo vamos a usar habitualmente es 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' and its value. The status bar at the bottom indicates a 200 OK response.

Home Workspaces ▾ API Network ▾ Explore

Search Postman

Invite Upgrade

My Workspace New Import

Dev GET GET with Variables

Save Send

GET `{{base_url}}/get?somevariable={{parameter_value}}`

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

Query Params

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

Body Cookies Headers (7) Test Results

Status: 200 OK Time: 215 ms Size: 652 B Save Response

Variables en Postman

The screenshot shows the Postman web interface. At the top, there's a navigation bar with 'Home', 'Workspaces', 'API Network', and 'Explore'. A search bar labeled 'Search Postman' is on the right. Below the navigation bar, the 'My Workspace' section is active, showing a list of environments: 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', 'Flows', and 'History'. The 'Environments' tab is selected, and the 'Dev' environment is highlighted. The 'Dev' environment is currently selected, and its variables are displayed in a table. The table has four columns: 'VARIABLE', 'TYPE', 'INITIAL VALUE', and 'CURRENT VALUE'. There are three variables listed: 'base_url', 'parameter_value', and 'json_value'. Each variable has a checkbox in the 'VARIABLE' column, a 'default' type, and its corresponding initial and current values. The 'base_url' variable has an initial value of 'httpbin.org/' and a current value of 'httpbin.org/'. The 'parameter_value' variable has an initial value of '50000' and a current value of '50000'. The 'json_value' variable has an initial value of 'This is a json text value' and a current value of 'This is a json text value'. Below the table, there is a button labeled 'Add a new variable'. The right sidebar shows the 'Dev' environment name, a 'Fork' button, a 'Save' button, a 'Share' button, and a 'Persist All' button. There are also icons for 'Reset All' and 'Info'.

Home Workspaces API Network Explore

Search Postman

Invite Upgrade

My Workspace New Import

Dev

GET GET with Variables POST POST with variables +

Dev

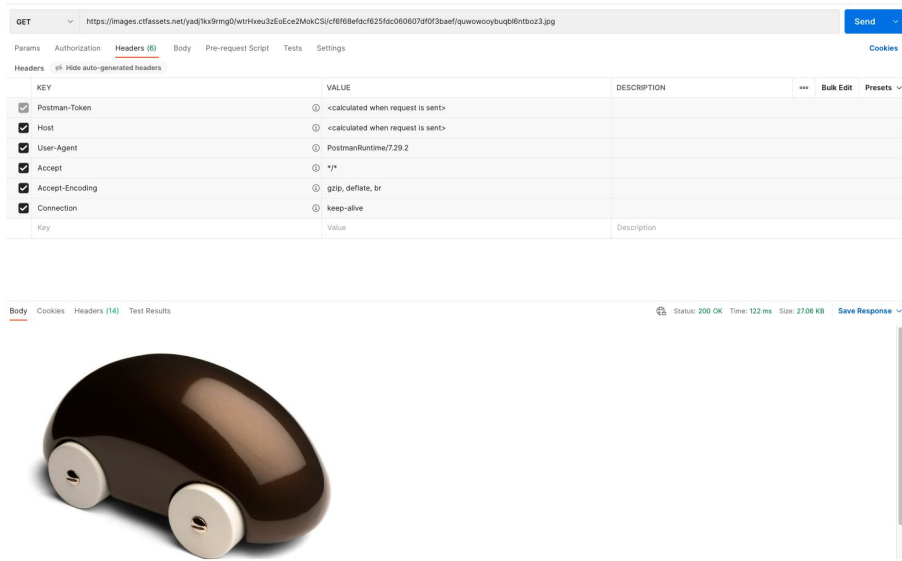
Fork 0 Save Share

VARIABLE	TYPE	INITIAL VALUE	CURRENT VALUE	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					

Proprietary + Confidential

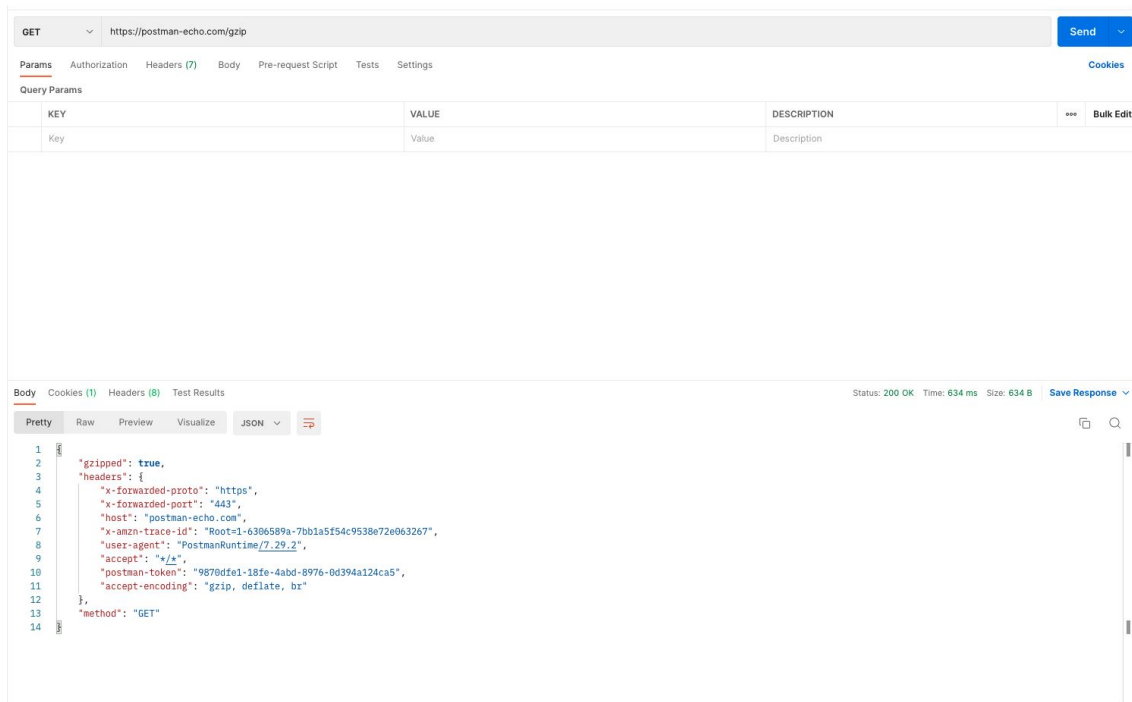
Content types

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



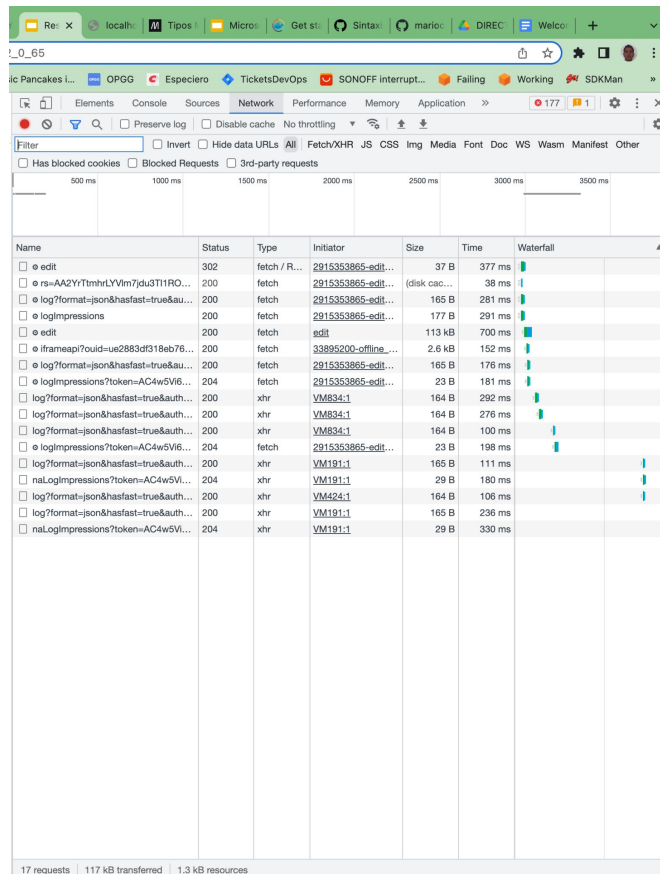
Respuestas comprimidas

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

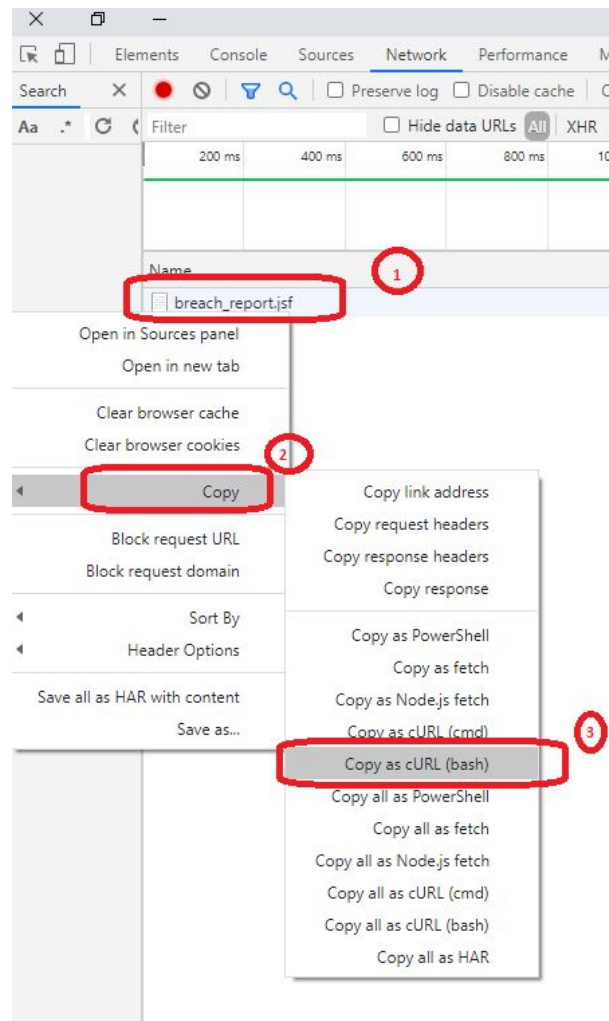


Importar llamadas curl a Postman

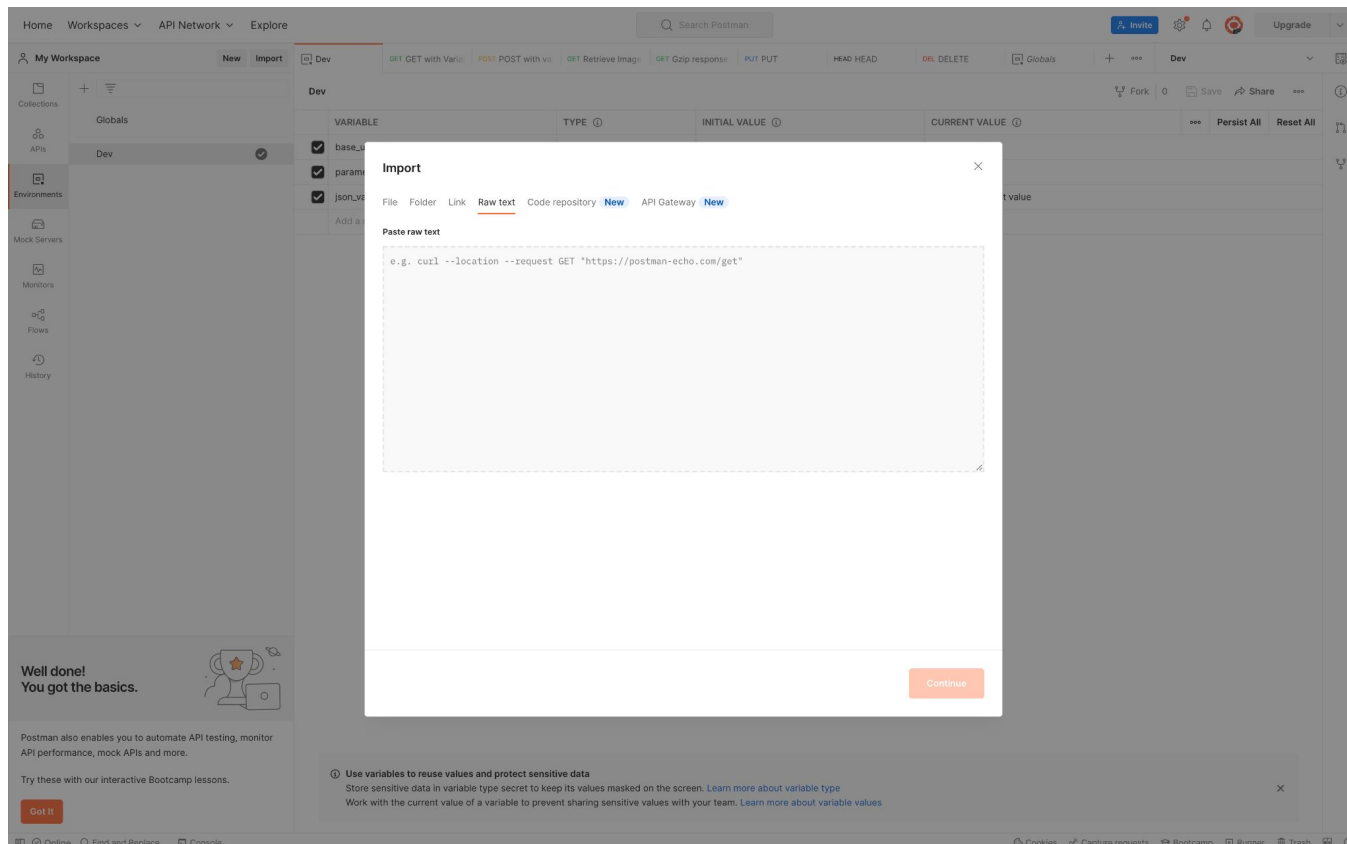
Postman puede importar llamadas curl a su interfaz mediante muy sencillos pasos, esto es importante cuando estamos rastreando algún problema que es visible en el navegador y sospechamos que es responsabilidad de los servicios que sirven dicha información. En google chrome abrimos las herramientas de desarrollador, en la pestaña de red



Damos click secundario a la llamada que queremos importar a postman y seleccionamos copy as curl



En postman clickeamos import y seleccionamos la opción de Raw Text



Clickeamos en continuar, después en import y tendremos la llamada en Postman

Home Workspaces API Network Explore

Search Postman

Invite Upgrade

My Workspace New Import

Dev GET GET with Vite POST POST with GET Retrieve limit GET Gzip response PUT PUT HEAD HEAD DELETE DELETE

Globals

Dev

https://github.com/mariacruzgonzalez/BAZJAVA12022/commit/8078a737431fdefd8e33b75f36c46c01891412c7/rollup?direction=sw

GET https://github.com/mariacruzgonzalez/BAZJAVA12022/commit/8078a737431fdefd8e33b75f36c46c01891412c7/rollup?direction=sw

Params Authorization Headers (19) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> direction	sw		
Key	Value	Description	

Response

Well done! You got the basics.

Postman also enables you to automate API testing, monitor API performance, mock APIs and more.

Try these with our interactive Bootcamp lessons.

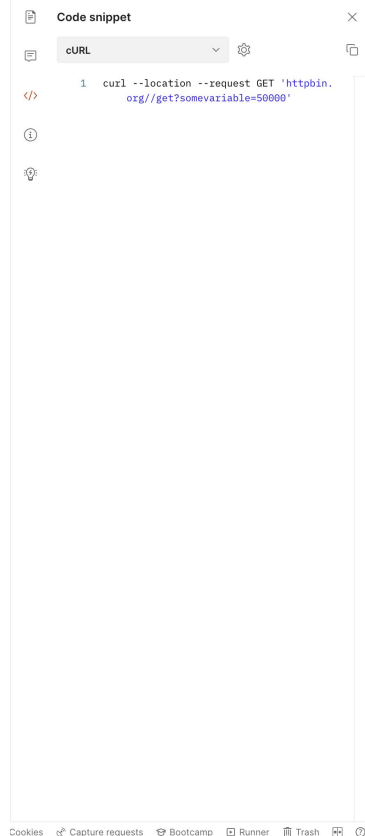
Get It

Click Send to get a response

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

Llamadas a código

De la misma forma en postman al tener una llamada, por ejemplo, parte de una colección ajena, podemos exportar dicha llamada a código, para este ejemplo, curl: clickeamos en postman a la derecha, en la opción code. Podremos elegir el lenguaje en el que queremos que se exporte la llamada, como mencione, se usará curl. Ese código lo podemos ejecutar en terminal o importarlo a nuestros proyectos en código



Feedback Form

Let us know your feedback!

<https://forms.gle/WKtc8wZeSxWnjGo8A>

