

POSTMAN



Ejemplo de rutas



`https://api-test-fun.glitch.me/info`

<https://api-test-fun.glitch.me> → URL Base

`/info` → endpoint path

https://simple-books-api.glitch.me

Save  



GET  https://simple-books-api.glitch.me Send 

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

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (6) Test Results  200 OK 806 ms 258 B [Save Response](#) 

Pretty Raw Preview Visualize JSON  

```
1 {  
2   "message": "Welcome to the Simple Books API."  
3 }
```



Body de un GET

GET

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

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

Respuesta del Server

```
Body Cookies (1) Headers (7) Test Results 200 OK 367 ms 734 B
Pretty Raw Preview Visualize JSON
1 {
2   "args": {},
3   "headers": {
4     "x-forwarded-proto": "http",
5     "x-forwarded-port": "80",
6     "host": "postman-echo.com",
7     "x-amzn-trace-id": "Root=1-60c3cd85-5ecde63341aef1056b6e88cf",
8     "content-length": "10",
9     "content-type": "application/json",
10    "user-agent": "PostmanRuntime/7.28.0",
11    "accept": "*/*",
12    "postman-token": "f76012a6-a195-47f3-a73c-7b730d8728f9",
13    "accept-encoding": "gzip, deflate, br"
14  },
15  "url": "http://postman-echo.com/get"
16 }
```

Qué es un Ambiente y una Colección?

Colecciones y Ambientes

Cada clase del curso será una nueva **colección** con request

Un **ambiente** contiene una o varias colecciones.

Podemos crear variables de ambiente y otras de colección.

Set as Variable

Simple Api Call / API Status

GET ▼ <https://simple-books-api.glitch.me/status>

Params Authorization Headers (6) Body Pre-request Script

Query Params

KEY	VALUE
-----	-------

Elegir el tipo de variable

Simple Api Call / API Status

GET

https://simple-books-api.glitch.me

Params

Authorizat

Test:

Query Params

KEY

Key

ody

Cookies

Head

Pretty

Raw

Preview

1

```
{
```

2

```
  "status": "OK"
```

Set as new variable

Name

baseUrl

Value

https://simple-books-api.glitch

Scope

Select A Scope



Global



Environment



Collection: Simple ...

Editar Variables de Colección



Variables de Colección

Simple Api Call



Run

Authorization Pre-request Script Tests Variables ●

These variables are specific to this collection and its requests. [Learn more about collection variables.](#) ↗

	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ
<input checked="" type="checkbox"/>	baseUrl	https://simple-books-api.glit...	https://simple-books-api.glitch.me
	Add a new variable		

Simple Api Call 👁️ 0 🔗 0 ▶ Run

Authorization Pre-request Script Tests Variables ●

These variables are specific to this collection and its requests. [Learn more about collection variables.](#) ➤

	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ
<input checked="" type="checkbox"/>	baseUrl	https://simple-books-api.glit...	https://simple-books-api.glitch.me
	Add a new variable		

- **Variable:** nombre de la variable
- **Initial value:** es el valor inicial que toma la variable
- **Current value:** valor actual, que puede ser eventualmente actualizada

	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ
<input checked="" type="checkbox"/>	baseURL	https://simple-books-api.glit...	https://simple-books-api.glitch.me

Initial Value: valor inicial que es público para las personas que compartas la colección

Current Value: valor interno dentro de mi Postman

Simple Api Call / API Status

GET



{{baseURL}}

Params


Authorization

Request Script

Query Params

KEY

Key

 baseURL

INITIAL https://simple-books-api.glitch.me

CURRENT https://simple-books-api.glitch.me

SCOPE Collection

Body

Cookies

Headers (6)

Test Results

Simple Api Call / API Status

GET



{{baseUrl}}/status

Params

Authorization

Headers (6)

Body

Pre-request Script

Query Params

KEY	VALUE



Query Parameters

GET



Enter request URL

Send



Params Authorization Headers (8) Body Pre-request Script Tests Settings

Cookies

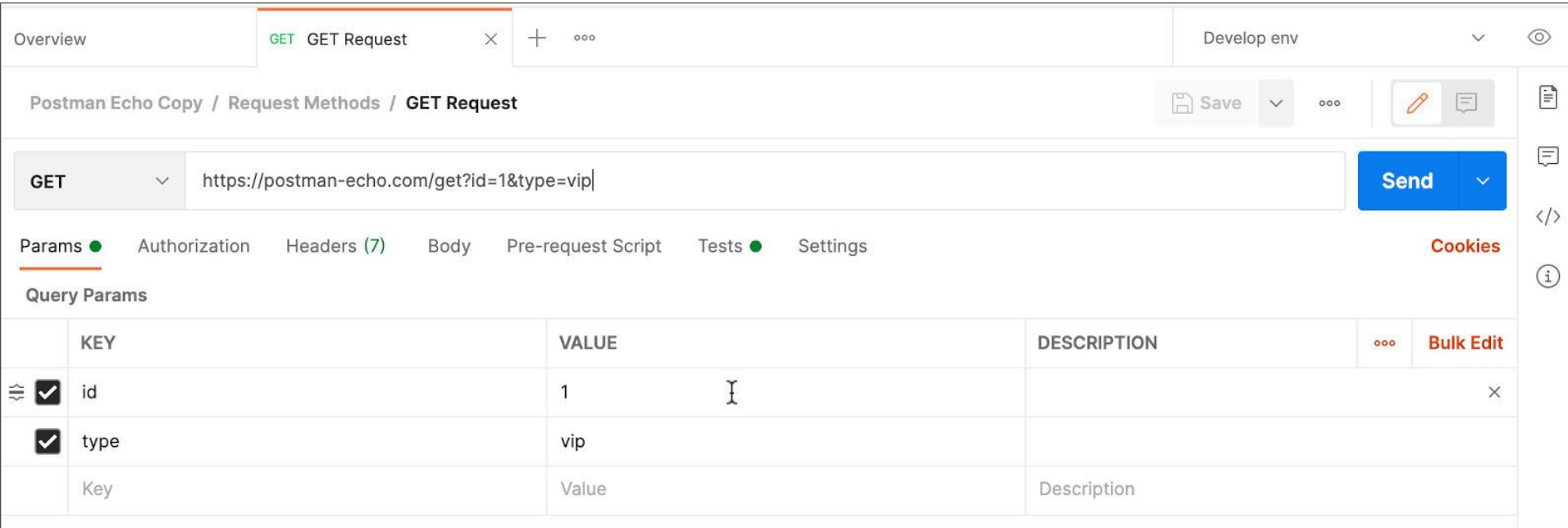
Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Response



Envío de Parámetros



The screenshot shows the Postman interface for a GET request. The URL is `https://postman-echo.com/get?id=1&type=vip`. The 'Params' tab is active, displaying a table of query parameters.

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	1			
<input checked="" type="checkbox"/>	type	vip			
	Key	Value	Description		

Envío de Parámetros

Se pueden enviar parámetros usando la pestaña Params

Los parámetros de consulta se añaden al final de la URL de la solicitud, a continuación? y enumerados en pares clave-valor, separados por & usando la siguiente sintaxis: ? id = 1 & type = new

GET ▼ `{{baseUrl}}/books?type=fiction&limit=2`

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

<input checked="" type="checkbox"/>	type	fiction
<input checked="" type="checkbox"/>	limit	2

Body Cookies Headers (6) Test Results  200 OK ;

Pretty Raw Preview Visualize JSON ▼ 

```
1  [
2    {
3      "id": 1,
4      "name": "The Russian",
5      "type": "fiction",
6      "available": true
7    },
8    {
9      "id": 3,
10     "name": "The Vanishing Half",
11     "type": "fiction",
```

Path Variables

Path Variables

Los parámetros de ruta forman parte de la URL de la solicitud y se hace referencia a ellos mediante marcadores precedidos por `:` como en el siguiente ejemplo: `/customer/:id`

GET



{{BooksUrl}}/orders/:orderId

Params ●

Authorization ●

Headers (9)

Body ●

Pre-request Script

Query Params

	KEY	VALUE
	Key	Value

Path Variables

	KEY	VALUE
	orderId	{{OrderId}}



Ejercicios

Qué se envía en el body
en un POST?

POST → Body: Row Data

Para enviar cualquier dato en formato

Params Authorization Headers (10) **Body** Pre-request Script Tests Settings

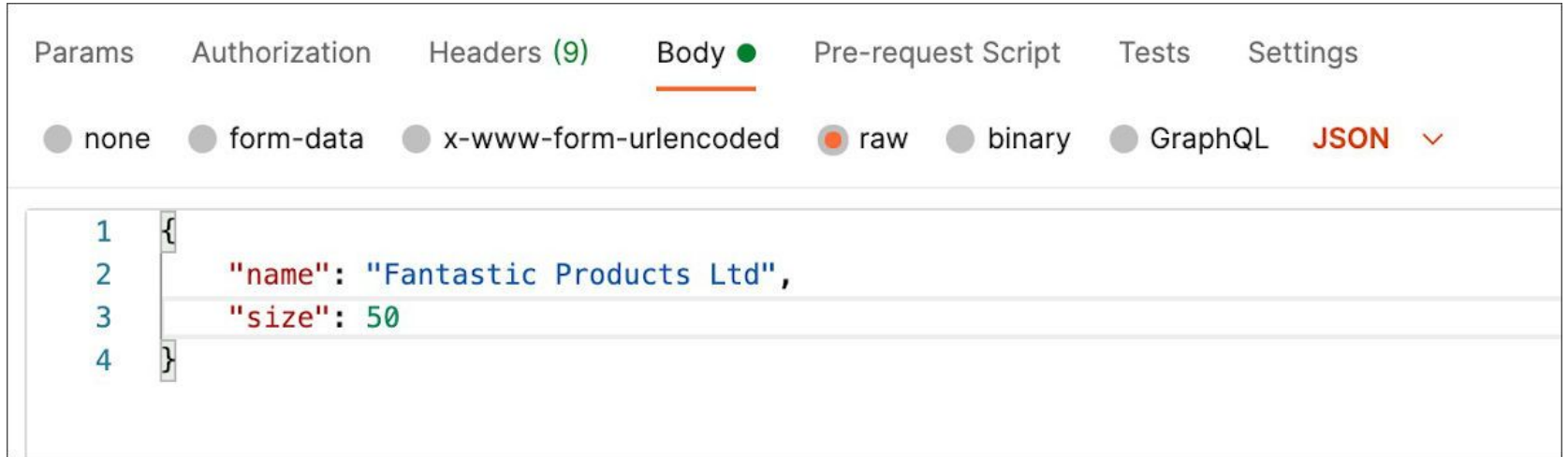
none form-data x-www-form-urlencoded raw binary GraphQL **JSON** ^

```
1 {  
2  
3  
4 ...  
5 }
```

- Text
- JavaScript
- JSON
- HTML
- XML

El body

Para enviar cualquier dato en formato JSON



Introducción a JSON

JSON

```
{ "name" : "John" , "age" : 30 , "car" : null }
```

Qué es un JSON?

- JSON - JavaScript Object Notation
- JSON es un formato liviano de intercambio de datos
- JSON es escrito en texto plano basado en notación Javascript
- JSON es usado para enviar y recibir datos

Un dato en JSON - Nombre y Valor

JSON se escribe en pares nombre/valor

```
"firstName" : "John"
```

Objetos en Json

Se escriben entre llaves

```
{ "firstName" : "John" , "lastName" : "Doe" }
```


Arreglos en JSON

Van entre parentesis rectos

```
"employees" : [  
  { "firstName" : "John", "lastName" : "Doe" },  
  { "firstName" : "Anna", "lastName" : "Smith" },  
  { "firstName" : "Peter", "lastName" : "Jones" }  
]
```