

# INSTALACIÓN Y CONFIGURACIÓN DE POSTMAN

## INSTALACIÓN DE POSTMAN

- Descargar de la página oficial Postman: <https://www.postman.com/downloads/>

The screenshot shows the 'Download Postman' page on postman.com/downloads/. At the top, there's a navigation bar with links for Product, Solutions, Pricing, Enterprise, Why Postman, and Resources. Below the navigation, the main heading is 'Download Postman'. A sub-headline says 'Download the app to get started using the Postman API Platform today. Or browser experience, you can try the web version of Postman'. Underneath, there's a section titled 'The Postman app' with a sub-instruction 'Download the app to get started with the Postman API Platform.' Below this is a large orange button with the text 'Windows 64-bit' and a small icon of a Windows logo. To the right of the main content area, there's a sidebar showing a preview of the Postman interface with tabs for Home, Workspaces, and API Network, and a collection named 'Notion's Public Workspace'.

- Ejecutar el instalable y seguir el proceso de instalación:
  1. Utilizar la cuenta de correo de educastur:

The screenshot shows a landing page for 'Working with APIs simplified with Postman'. It features a large input field labeled 'Enter your work email:' with a yellow background. Below it, a sub-instruction reads 'Your work email makes it easy for you to collaborate with your teammates.' There is a prominent orange button with the text 'Create Free Account'. Below the button, a link says 'Already have an account? Log In'. Further down, there are three additional buttons for 'Continue with Google', 'Continue with Github', and 'Single Sign On (SSO)'. At the very bottom, there is a link 'Continue without an account'.

## 2. Introducir en el navegador los datos solicitados

**Create Postman account**

Work email  
patriciasfo@educastur.org

Username  
patriciasfo

Password Show  
\*\*\*\*\*

Receive product updates, news, and other marketing communications

Stay signed in

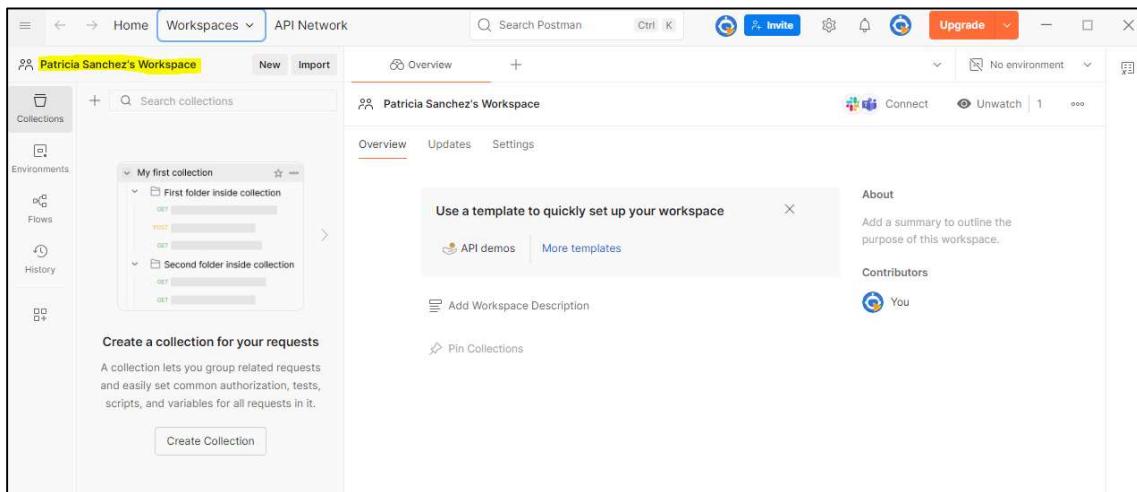
 Success!

By creating an account, you agree to our [terms](#) and [privacy policy](#).

**Create Free Account**

or

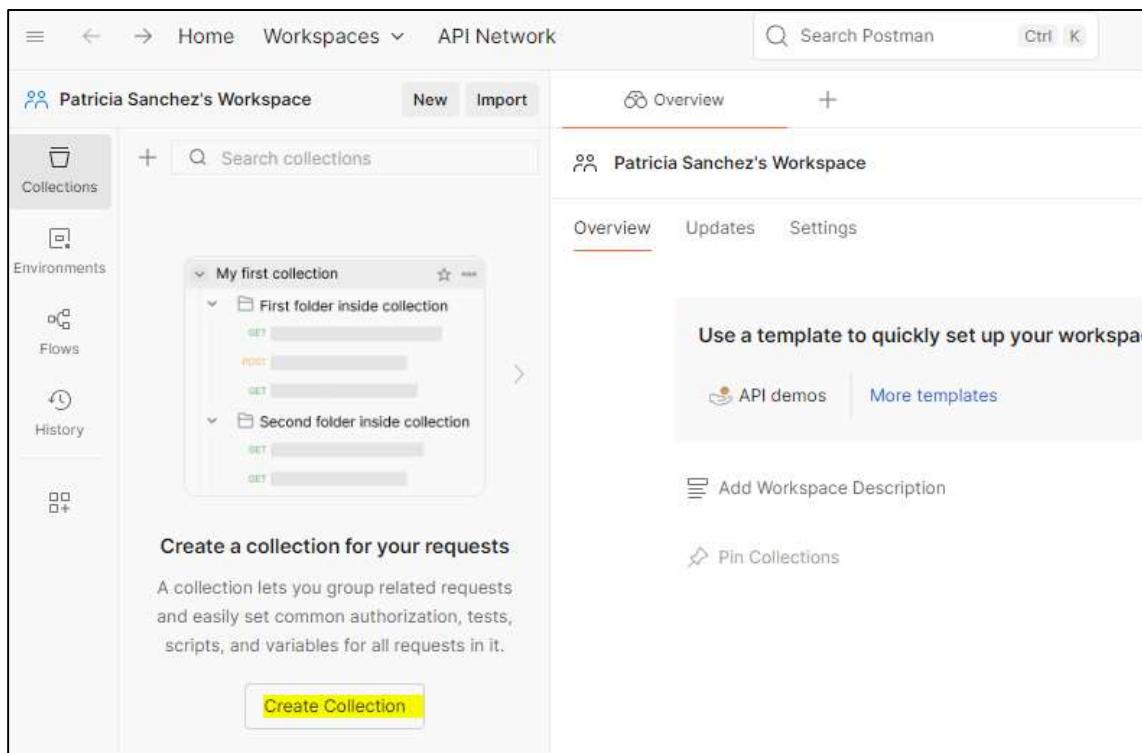
## 3. Una vez creada la cuenta y logueados, tendremos un workspace sobre el que trabajar



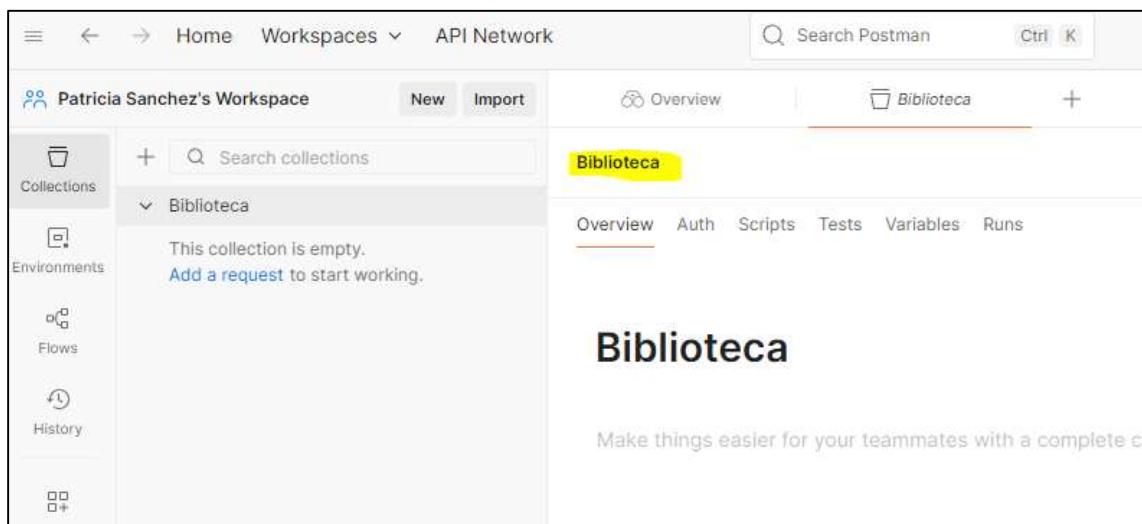
The screenshot shows the Postman application interface. At the top, there's a navigation bar with Home, Workspaces (selected), API Network, and various search and settings icons. Below the navigation is a sidebar with sections for Collections, Environments, Flows, and History. The main area displays "Patricia Sanchez's Workspace". Inside, there's an "Overview" tab showing a collection named "My first collection" which contains two folders: "First folder inside collection" and "Second folder inside collection". A modal window titled "Use a template to quickly set up your workspace" offers options like "API demos" and "More templates". To the right of the workspace, there are sections for "About" (with a placeholder for a workspace description) and "Contributors" (listing "You"). At the bottom left, there's a button to "Create Collection".

## CONFIGURACIÓN DE POSTMAN

- El primer paso es crear una colección para organizar la información y posteriormente crear las peticiones necesarias:



The screenshot shows the Postman application interface. On the left, there's a sidebar with icons for Collections, Environments, Flows, and History. The main area displays a workspace titled "Patricia Sanchez's Workspace". A collection named "My first collection" is expanded, showing two nested folders: "First folder inside collection" and "Second folder inside collection", each containing several requests (GET and POST methods). Below this, there's a section titled "Create a collection for your requests" with a "Create Collection" button.



The screenshot shows the same Postman interface after creating a new collection. The workspace is still "Patricia Sanchez's Workspace". A collection named "Biblioteca" is selected and expanded. The interface indicates that this collection is empty and suggests adding a request. The right panel shows the "Overview" tab for the "Biblioteca" collection, with tabs for Overview, Auth, Scripts, Tests, Variables, and Runs. The "Biblioteca" tab is highlighted.

- Ahora ya podemos crear las peticiones:

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'Collections' (highlighted), 'Environments', 'Flows', and 'History'. The main area shows 'Patricia Sanchez's Workspace' with a 'Biblioteca' collection expanded. A 'New Request' button is visible. The central part of the screen displays a 'GET New Request' dialog. The method is set to 'GET', the URL field contains 'Enter URL or paste text...', and the 'Send' button is highlighted in blue. Below the URL field, tabs for 'Params', 'Authorization', 'Headers (7)', 'Body', 'Scripts', 'Tests', and 'Settings' are shown, along with a 'Cookies' section. A 'Query Params' table is present with one row: 'Key' (Value) and 'Description' (Description). The right side of the interface includes various icons for saving, sharing, and upgrading.

La petición por defecto que se crea siempre es una petición GET, pero lo podremos modificar.

Lo que hay que hacer, es darle un nombre, escribir la url e indicar el cuerpo si es necesario. Veamos cómo es el proceso a través de ejemplos:

## 1. Obtener todos los libros

This screenshot shows the Postman interface after a successful API call. The left sidebar remains the same. In the center, a request named 'GET Todos los libros' is selected in the list. The URL field shows 'http://localhost:8090/api'. The 'Headers' tab indicates 6 headers. The 'Body' tab shows a JSON response: '[]'. The status bar at the bottom shows '200 OK' with a response size of 166 B. The right side of the interface includes 'Save', 'Share', and 'Send' buttons, along with other status and settings icons.

## 2. Añadir un nuevo libro

HTTP Biblioteca / Añadir un libro

POST <http://localhost:8090/api> Send

Params Authorization Headers (8) **Body** Scripts Settings Cookies

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

```
1 {  
2   "autor": "Gabriel García Márquez",  
3   "titulo": "Cien años de soledad"  
4 }
```

Body Cookies Headers (4) Test Results (1/1) 200 OK 184 ms 123 B Save Response

Raw Preview Visualize 1

This screenshot shows a successful POST request to add a book using the Postman application. The URL is <http://localhost:8090/api>. The request body is a JSON object with 'autor' set to 'Gabriel García Márquez' and 'titulo' set to 'Cien años de soledad'. The response status is 200 OK with a response time of 184 ms and a size of 123 B.

### 3. Actualizar un libro

HTTP Biblioteca / Modificar un libro

POST <http://localhost:8090/api> Send

Params Authorization Headers (8) **Body** Scripts Settings Cookies

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

```
1 {  
2   "id": 2,  
3   "autor": "Javier Castillo",  
4   "titulo": "El susurro del fuego"  
5 }
```

Body Cookies Headers (4) Test Results (1/1) 200 OK 49 ms 123 B Save Response

Raw Preview Visualize 1

This screenshot shows a successful POST request to update a book using the Postman application. The URL is <http://localhost:8090/api>. The request body is a JSON object with 'id' set to 2, 'autor' set to 'Javier Castillo', and 'titulo' set to 'El susurro del fuego'. The response status is 200 OK with a response time of 49 ms and a size of 123 B.

### 4. Obtener un libro

HTTP Biblioteca / Un libro

GET <http://localhost:8090/api/2>

Params Authorization Headers (6) Body Scripts Settings Cookies

(none) form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

1 Ctrl+Alt+P to Ask AI

Body Cookies Headers (5) Test Results (1/1)

200 OK 12 ms 230 B Save Response

{ } JSON Preview Visualize

```
1 {
2   "id": 2,
3   "titulo": "El susurro del fuego",
4   "autor": "Javier Castillo"
5 }
```

## 5. Eliminar un libro que no existe

HTTP Biblioteca / Eliminar un libro

DELETE <http://localhost:8090/api/3>

Params Authorization Headers (6) Body Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body Cookies Headers (4) Test Results

500 Internal Server Error 58 ms 260 B Save Response

{ } JSON Preview Debug with AI

```
1 {
2   "timestamp": "2025-10-01T09:44:28.663+00:00",
3   "status": 500,
4   "error": "Internal Server Error",
5   "path": "/api/3"
6 }
```

## 6. Eliminar un libro que existe

The screenshot shows a REST client interface with the following details:

- HTTP Method:** DELETE
- URL:** `http://localhost:8090/api/2`
- Headers:** (6)
- Body:** (Empty)
- Scripts:** (Empty)
- Settings:** (Empty)
- Cookies:** (Empty)
- Query Params:** (Empty)
- Body:** (Raw) 1
- Headers:** (4)
- Test Results:** (Empty)
- Response Status:** 200 OK
- Response Time:** 82 ms
- Response Size:** 123 B
- Save Response:** (Icon)
- Copy:** (Icon)
- Close:** (Icon)