

UNIVERSIDAD POLITECNICA SALESIANA
INGENIERIA EN CIENCIAS DE LA COMPUTACIÓN

NOMBRE: Nicole Paredes
CARRERA: Computación
TALLER: N°3

ASIGNATURA: Programación y plataformas web
FECHA: 5/11/2025
TEMA: Intercambio de Datos

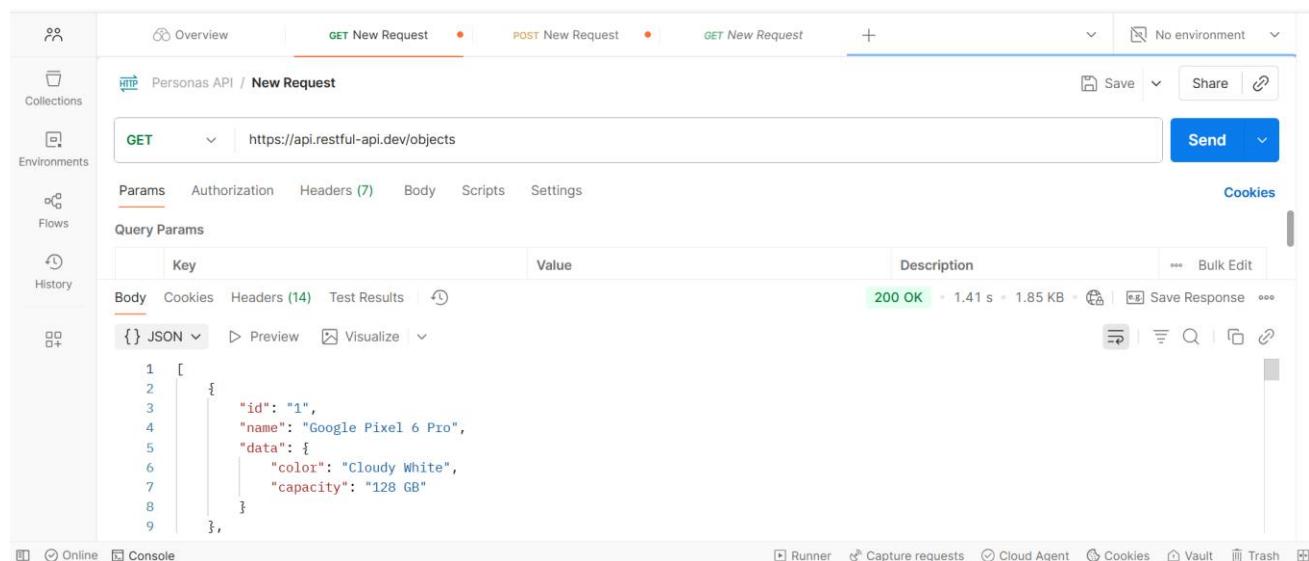
OBJETIVOS

- Aprender a utilizar la herramienta Postman para enviar y recibir datos mediante los métodos HTTP (GET, POST, PUT, DELETE), comprobando cómo cada uno de ellos permite crear, consultar, actualizar y eliminar información en una API de manera práctica.
- Aprender a leer y enviar datos en formato JSON mediante el uso de los métodos HTTP en Postman, comprendiendo cómo se realiza el intercambio de información entre el cliente y una API.

Repositorio online de datos JSON utilizado: <https://restful-api.dev/>

1. METODO GET

Se utilizo en método GET para obtener la lista de todos los datos



The screenshot shows the Postman interface with the following details:

- Request Type:** GET
- URL:** https://api.restful-api.dev/objects
- Headers:** Authorization, Headers (7), Body, Scripts, Settings
- Body:** JSON (Preview shows a single item in a list: { "id": "1", "name": "Google Pixel 6 Pro", "data": { "color": "Cloudy White", "capacity": "128 GB" } })
- Response Status:** 200 OK
- Response Time:** 1.41 s
- Response Size:** 1.85 KB

2. METODO POST

Se utilizo el método POST para crear un nuevo producto, en el Body se enviaron los datos en formato JSON

Home Workspaces API Network

Personas API | GET New Request | POST New Request | PUT New Request | DEL New Request | POST https://api.restful-api.c | +

Collections

Environments

Flows

History

https://api.restful-api.dev/objects

POST https://api.restful-api.dev/objects

Params Authorization Headers (9) Body Scripts Settings

Body (raw JSON)

```

1 {
2   "name": "Apple Ipad Pro 16, 8 gen",
3   "data": {
4     "year": 2019,
5     "price": 849.99,
6     "CPU model": "Intel Core i9",
7     "Hard disk size": "1 TB"
8   }
9
10
11 }
```

Body Cookies Headers (14) Test Results 200 OK

{ } JSON ▾ Preview Visualize ▾

```

1 {
2   "id": "ff8081819782e69e019a2cc3232d72f5",
3   "name": "Apple Ipad Pro 16, 8 gen",
4   "createdAt": "2025-10-28T21:39:37.389+00:00",
5   "data": {
6     "year": 2019,
7     "price": 849.99,
8     "CPU model": "Intel Core i9",
9     "Hard disk size": "1 TB"
10   }
11 }
```

3. METODO PUT

Se actualizo un registro existente (el que fue creado)

Personas API | GET New Request | POST New Request | PUT New Request | DEL New Request | PUT https://api.restful-api.dk | +

Collections

Environments

Flows

History

https://api.restful-api.dev/objects/ff8081819782e69e019a2cc3232d72f5

PUT https://api.restful-api.dev/objects/ff8081819782e69e019a2cc3232d72f5

Params Authorization Headers (9) Body Scripts Settings

Body (raw JSON)

```

1 {
2   "id": "ff8081819782e69e019a2cc3232d72f5",
3   "name": "Apple Ipad Pro Max 16, 8 gen",
4   "data": {
5     "year": 2019,
6     "price": 899.99,
7     "CPU model": "Intel Core i9",
8     "Hard disk size": "1 TB"
9   }
10 }
```

Body Cookies Headers (14) Test Results 200 OK

{ } JSON ▾ Preview Visualize ▾

```

1 {
2   "id": "ff8081819782e69e019a2cc3232d72f5",
3   "name": "Apple Ipad Pro Max 16, 8 gen",
4   "updatedat": "2025-10-28T21:40:42.511+00:00",
5   "data": {
6     "year": 2019,
7     "price": 899.99,
8     "CPU model": "Intel Core i9",
9     "Hard disk size": "1 TB"
10   }
11 }
```

Se realizo un GET para comprobar que la información fue actualizada correctamente

The screenshot shows the Postman interface with a successful GET request to `https://api.restful-api.dev/objects/ff8081819782e69e019a2cc3232d72f5`. The response body is a JSON object:

```
1 {  
2   "id": "ff8081819782e69e019a2cc3232d72f5",  
3   "name": "Apple Ipad Pro Max 16, 8 gen",  
4   "data": {  
5     "year": 2019,  
6     "price": 899.99,  
7     "CPU model": "Intel Core i9",  
8     "Hard disk size": "1 TB"  
9   }  
10 }
```

4. METODO DELETE

Se utilizo el método DELETE para eliminar el producto que se creo

The screenshot shows the Postman interface with a successful DELETE request to `https://api.restful-api.dev/objects/ff8081819782e69e019a2cc3232d72f5`. The response body is a JSON object:

```
1 {  
2   "message": "Object with id = ff8081819782e69e019a2cc3232d72f5 has been deleted."  
3 }
```

Se ejecuto el método GET y se comprobó que el producto ya no existía

The screenshot shows the Postman interface with a left sidebar containing icons for Home, Workspaces, API Network, Collections, Environments, Flows, and History. The main workspace displays a GET request to the URL <https://api.restful-api.dev/objects/ff8081819782e69e019a2cc3232d72f5>. The request details show the method as GET, the URL, and various tabs like Params, Authorization, Headers, Body, Scripts, and Settings. Under the Body tab, the response is shown as JSON, indicating a 404 Not Found error with the message: "Object with id=ff8081819782e69e019a2cc3232d72f5 was not found."

RESULTADOS OBTENIDOS

- Se logro ejecutar correctamente los métodos HTTP GET, POST, PUT, DELETE con la API <https://restful-api.dev/>. Con GET se obtuvo la lista de todos los productos existentes en formato JSON, con POST se creo un nuevo producto y se verifico su existencia con un GET. Con PUT se actualizo el producto que fue creado. Finalmente, con DELETE se elimino el producto creado y se comprobó con un GET que ya no existía.

CONCLUSIONES

- Los métodos HTTP GET, POST,PUT y DELETE permiten manipular la información de forma efectiva en las APIS.
- Postman es una herramienta útil para probar estos métodos HTTP, facilita la visualización de datos y la manipulación sin necesidad de programar.

RESPALDO EN GIT HUB

Repositorio con evidencia de la práctica:

<https://github.com/NicoleParedes24/PostmanNicole>