
PROGRAMACION WEB AVANZADA

Actividad Práctica: Explorando Diferentes Tipos de Peticiones HTTP

Author

Jairo Quilumbaquin

Sangolqui, Ecuador

Miércoles 19 de Noviembre del 2023

Contents

1	Introducción	3
1.1	Las peticiones HTTP	3
2	Desarrollo	3
2.1	Elección de una API Ficticia	3
2.2	Realización de Peticiones	3
2.2.1	Peticion GET	3
2.2.2	Petición POST	4
2.2.3	Petición PUT	5
2.2.4	Petición PATCH	6
2.2.5	Petición DELETE	7

1 Introducción

1.1 Las peticiones HTTP

El protocolo utilizado para transferir datos entre servidores y clientes son los mensajes HTTP. Hay dos tipos de mensajes: las respuestas, que son las respuestas del servidor, y las peticiones, que son entregadas por el cliente para solicitar el inicio de una operación.[1]

2 Desarrollo

2.1 Elección de una API Ficticia

Elegimos la api sugerida por el documento de guía de la practica [https JSONPlaceholder](#)[2]

2.2 Realización de Peticiones

Para realizar las distintas peticiones HTTP usaremos el software de Insomnia, para testear la API y realizar el desarrollo de la actividad.

2.2.1 Petición GET

Obtenga información de un recurso:

URL usada para realizar la petición **GET** a la API:

```
1 url = 'https://jsonplaceholder.typicode.com/todos/1'
```

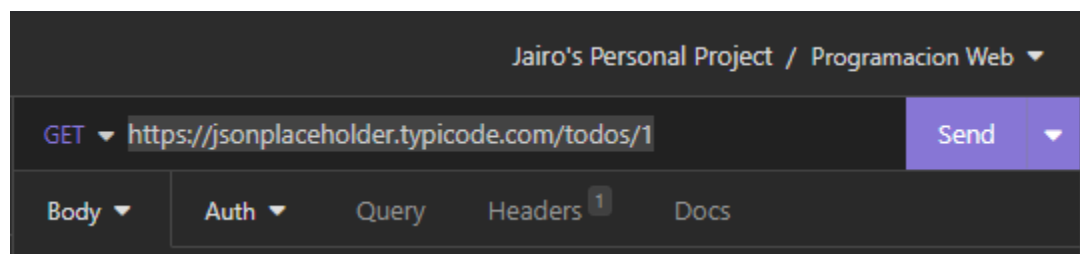


Figure 1: Uso de la URL en Insomnia

En la figura 2 se visualiza la respuesta obtenida desde la API:

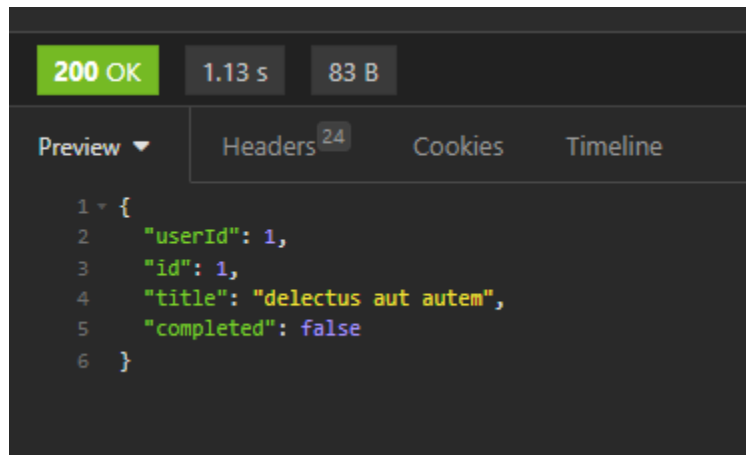


Figure 2: Respuesta recibida desde la API

Análisis de la respuesta:

- Cuerpo de la respuesta: la respuesta se entrega en formato JSON
- Código de estado: el código que recibe la aplicación 200 OK que indica una petición exitosa.

2.2.2 Petición POST

Cree un nuevo recurso.

```
1 url = 'https://jsonplaceholder.typicode.com/post'
```

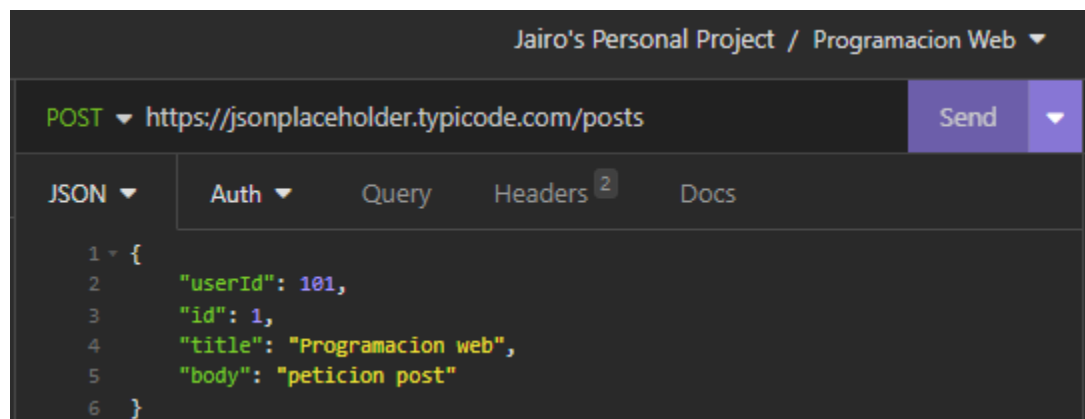


Figure 3: Uso de la URL en Insomnia

Análisis de la solicitud:

- Cuerpo de la solicitud: se envía un archivo en formato Json con los datos para la crear el nuevo recurso.

En la figura 4 se visualiza la respuesta obtenida desde la API:

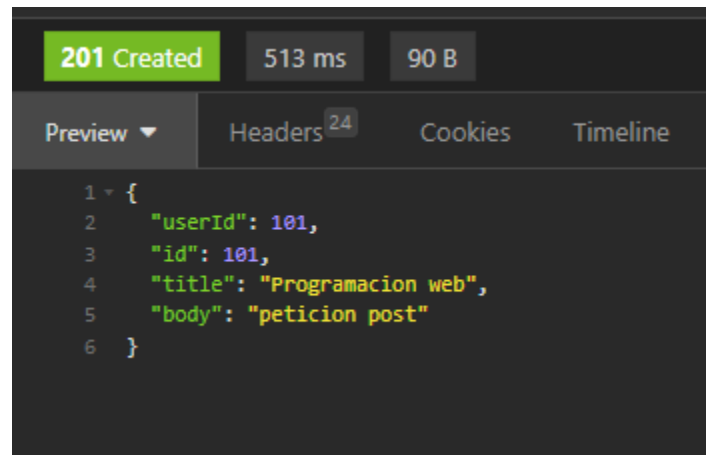


Figure 4: Respuesta recibida desde la API a la petición POST

Análisis de la respuesta:

- Cuerpo de la respuesta: la respuesta se entrega en formato JSON
- Código de estado: el código que recibe la aplicación 201 OK que indica que se creo exitosamente el nuevo recurso.

2.2.3 Petición PUT

Actualice completamente un recurso existente.

```
1 url = 'https://jsonplaceholder.typicode.com/posts/1'
```

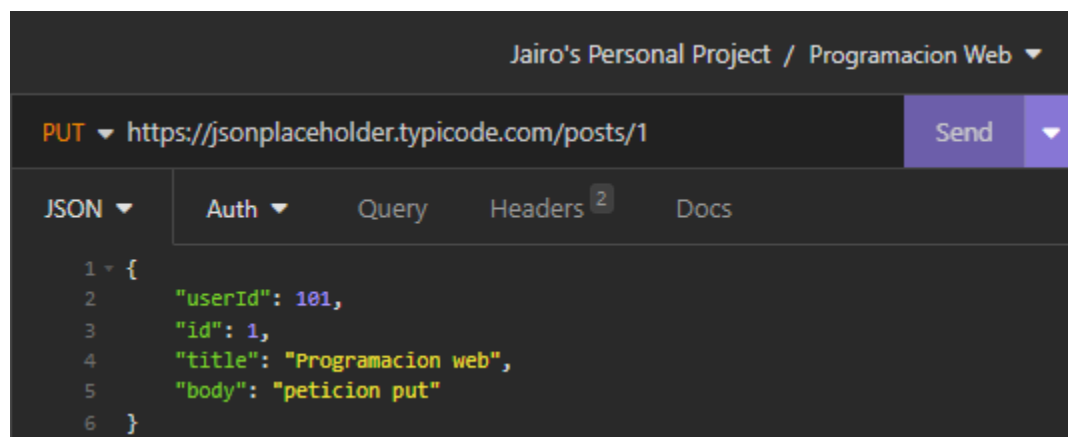


Figure 5: Uso de la URL en Insomnia

Análisis de la solicitud:

- Cuerpo de la solicitud: se envía un archivo en formato Json con los datos para actualizar el campo body de **petición post** a **petición put**

En la figura 6 se visualiza la respuesta obtenida desde la API:

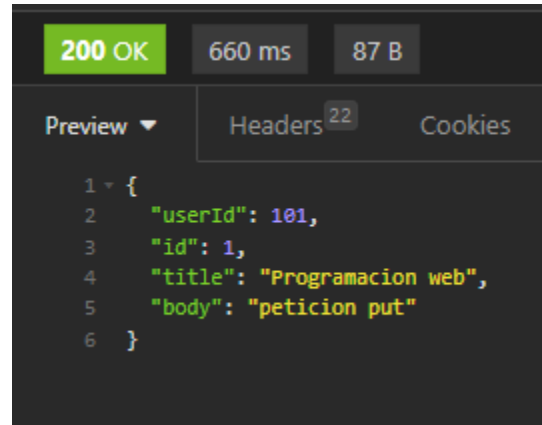


Figure 6: Respuesta recibida desde la API a la petición PUT

Análisis de la respuesta:

- Cuerpo de la respuesta: la respuesta se entrega en formato JSON
- Código de estado: el código que recibe la aplicación 200 OK que indica una petición exitosa

2.2.4 Petición PATCH

Actualice parcialmente un recurso existente.

```
1 url = 'https://jsonplaceholder.typicode.com/posts/1'
```

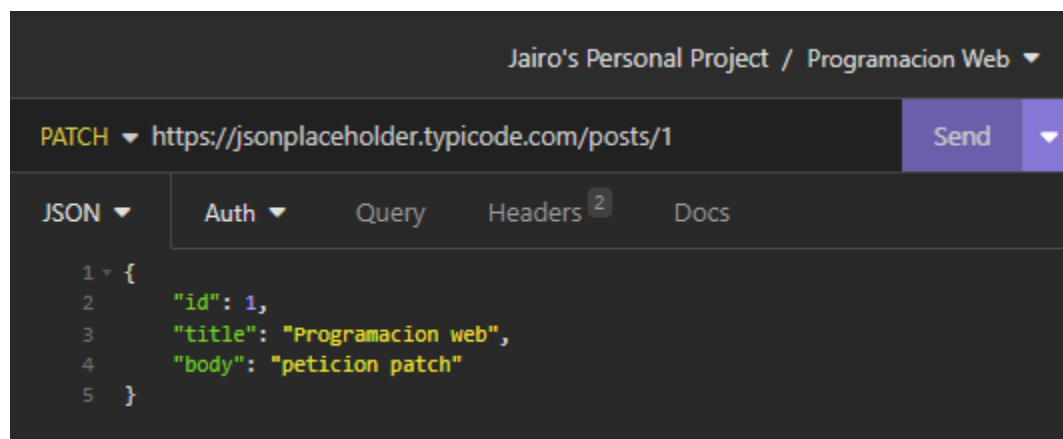


Figure 7: Uso de la URL en Insomnia

Análisis de la solicitud:

- Cuerpo de la solicitud: se envía un archivo en formato Json con los datos para actualizar el campo body parcialmente de **petición put** a **petición patch**

En la figura 8 se visualiza la respuesta obtenida desde la API:

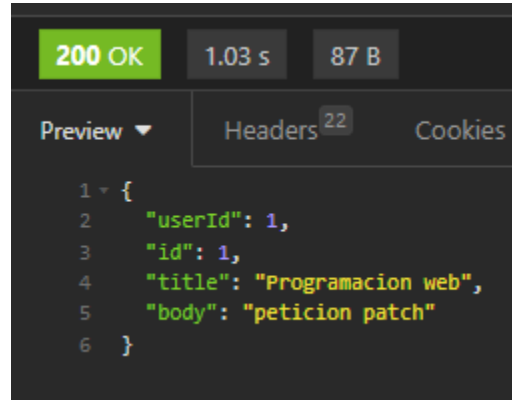


Figure 8: Respuesta recibida desde la API a la petición PATCH

Análisis de la respuesta:

- Cuerpo de la respuesta: la respuesta se entrega en formato JSON
- Código de estado: el código que recibe la aplicación 200 OK que indica una petición exitosa

2.2.5 Petición DELETE

Actualice parcialmente un recurso existente.

```
1 url = 'https://jsonplaceholder.typicode.com/posts/1'
```

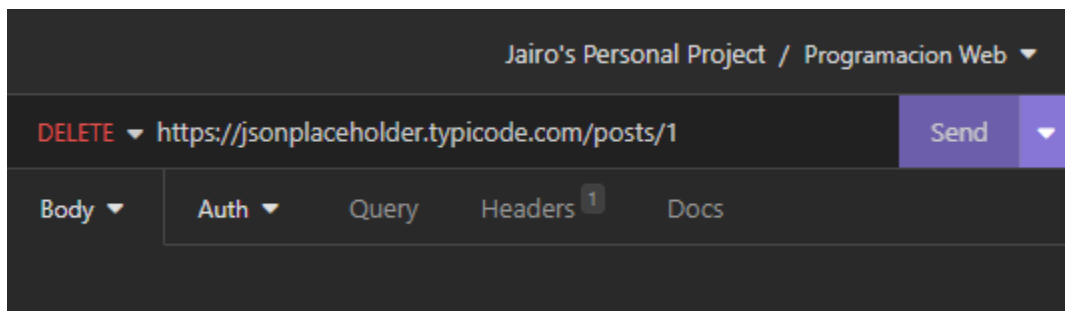


Figure 9: Uso de la URL en Insomnia

En la figura 10 se visualiza la respuesta obtenida desde la API:

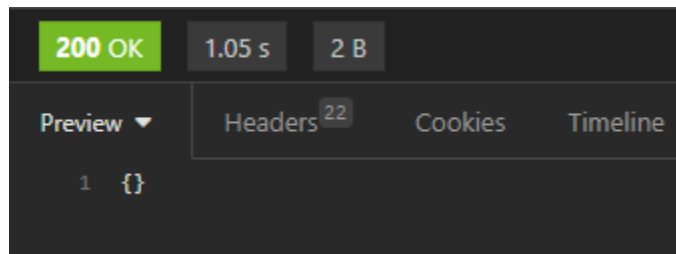


Figure 10: Respuesta recibida desde la API a la petición DELETE

Análisis de la respuesta:

- Cuerpo de la respuesta: no se recibe una respuesta en formato Json.
- Código de estado: el código que recibe la aplicación 200 OK que indica una petición exitosa

References

- [1] M. D. Network. Mensajes http. Recuperado el 19/11/2023. [Online]. Available: <https://developer.mozilla.org/es/docs/Web/HTTP/Messages>
- [2] Typicode. Jsonplaceholder - fake online rest api for testing and prototyping. Recuperado el [fecha de acceso]. [Online]. Available: <https://jsonplaceholder.typicode.com/guide/>