

CENTRO DE ENSEÑANZA TÉCNICA Y SUPERIOR



Escuela de Ingeniería

Desarrollo para Plataformas Heterogéneas

Creación de un API para un Carrito de Compras

Presenta:  
Anahi Hull m042124

Mexicali, B.C., 03 de febrero de 2025

# API de Carrito de Compras

Este API permite gestionar productos en un carrito de compras, realizar actualizaciones en las cantidades, eliminar productos específicos o vaciar el carrito completamente. El API está construido con Express.js y se conecta a una base de datos MySQL.

## Base URL

La base URL de este API es:

```
http://localhost:3000/api
```

## Endpoints

### 1. Obtener Productos

- **URL:** `/products`
- **Método:** `GET`
- **Descripción:** Obtiene todos los productos disponibles en la base de datos.
- **Respuesta:**

```
[
  {
    "id": 1,
    "nombre": "Producto 1",
    "precio": 100,
    "descripcion": "Descripción del producto 1",
    "imagen": "url imagen"
  },
  {
    "id": 2,
    "nombre": "Producto 2",
    "precio": 150,
    "descripcion": "Descripción del producto 2",
    "imagen": "url imagen"
  }
]
```

GET

http://localhost:3000/api/products

Send

QueryHeaders<sup>2</sup>AuthBody<sup>1</sup>TestsPre Run

Status: 200 OKSize: 710 BytesTime: 5 ms

ResponseHeaders<sup>8</sup>CookiesResultsDocs

```
1  [
2    {
3      "id": 1,
4      "nombre": "Kuromi",
5      "precio": 350,
6      "descripcion": "Kuromi Morada",
7      "imagen": "https://i5.walmartimages.com/asr/da96d257-d030-4cda-bd2e-ca3231017319
8        .3f456afb29acdd0793f6729eb15d75ec.jpeg?odnHeight=612&odnWidth=612&odnBg=FFFFFF"
9    },
10   {
11     "id": 2,
12     "nombre": "My Melody",
13     "precio": 450,
14     "descripcion": "My Melody Rosita",
15     "imagen": "https://i5.walmartimages.com/asr/534d2bb3-bd6d-4460-ba32-1cc033516bd2
16       .e221100409d355ee49384c74cd8fbab2.jpeg?odnHeight=612&odnWidth=612&odnBg=FFFFFF"
17   },
18   {
19     "id": 3,
20     "nombre": "Pochacho McDonalds",
21     "precio": 500,
22     "descripcion": "Pochaco McDonalds",
23     "imagen": "https://i5.walmartimages.com/asr/b19055cc-3ffa-4930-8764-a25ea5f9e918
24       .0ee2d32d82de3fed435369c8b0ae3911.webp?odnHeight=612&odnWidth=612&odnBg=FFFFFF"
25   }
26 ]
```

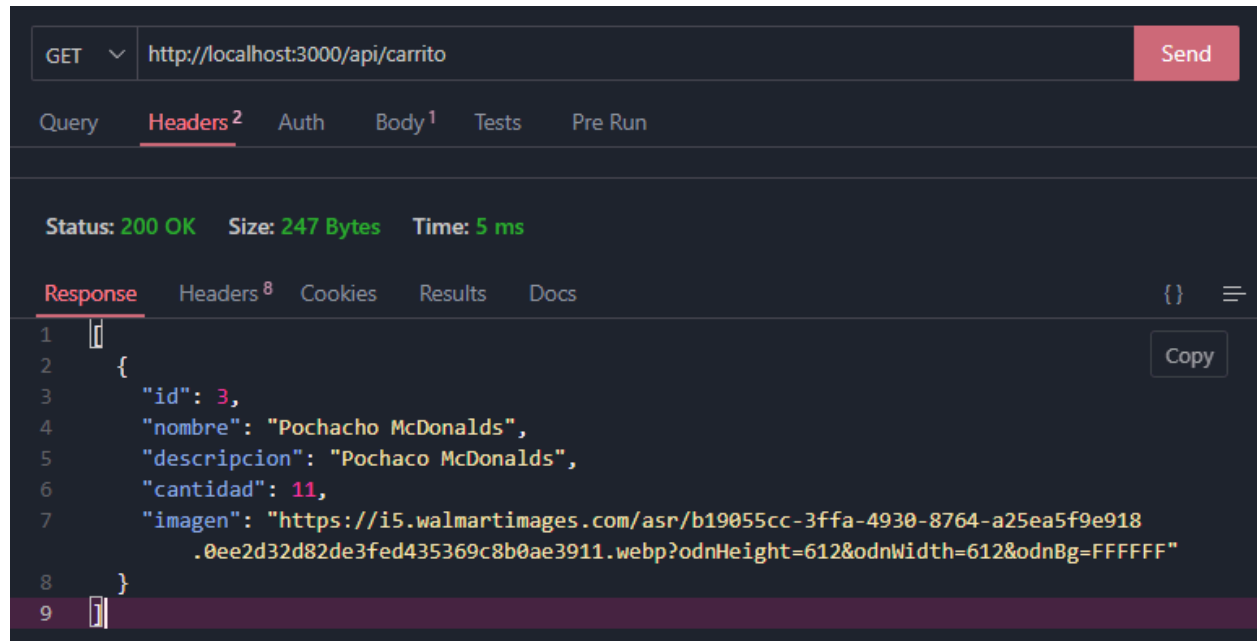
Copy

## 2. Obtener Carrito

- URL: `/carrito`
- Método: `GET`
- Descripción: Obtiene todos los productos en el carrito, con detalles como la cantidad y la descripción.
- Respuesta:

```
[
  {
    "id": 1,
    "nombre": "Producto 1",
```

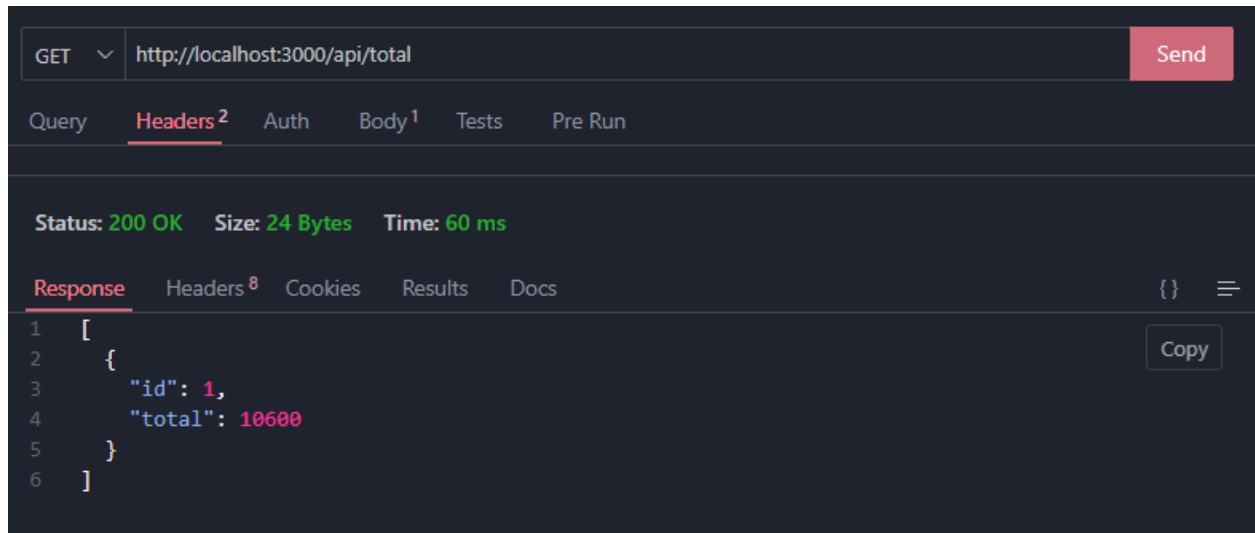
```
    "descripcion": "Descripción del producto 1",
    "cantidad": 2,
    "imagen": "url imagen"
  }
]
```



### 3. Obtener Total del Carrito

- **URL:** /total
- **Método:** GET
- **Descripción:** Obtiene el total de la compra en el carrito.
- **Respuesta:**

```
[
  {
    "id": 1,
    "total": 200
  }
]
```



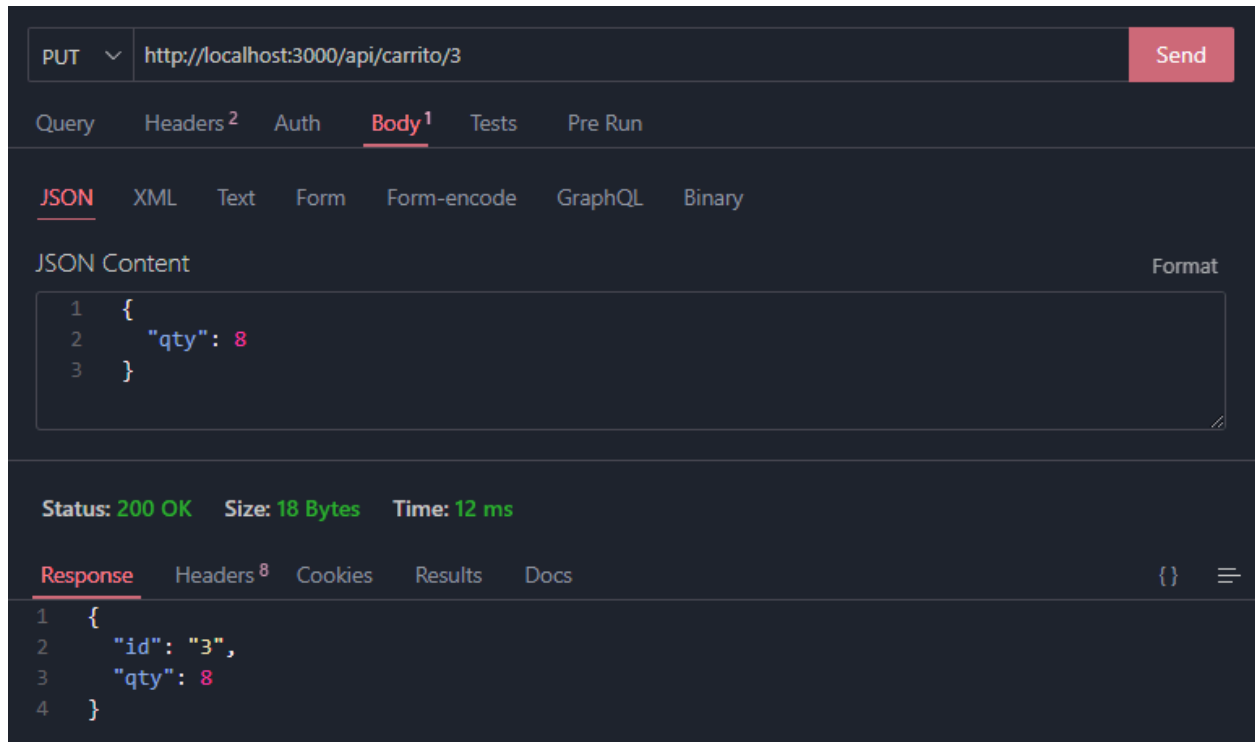
#### 4. Actualizar Cantidad de un Producto en el Carrito

- **URL:** `/carrito/:id`
- **Método:** `PUT`
- **Descripción:** Actualiza la cantidad de un producto específico en el carrito.
- **Parámetros:**
  - `id` (requerido): ID del producto a actualizar.
- **Body:**

```
{
  "qty": 3
}
```

- **Respuesta:**

```
{
  "id": 1,
  "qty": 3
}
```

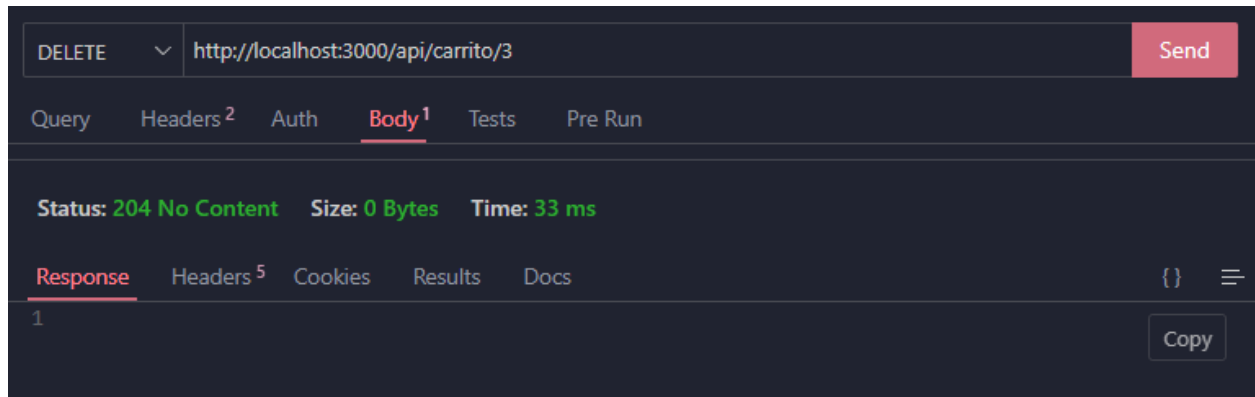


## 5. Eliminar Producto del Carrito

- **URL:** `/carrito/:id`
- **Método:** `DELETE`
- **Descripción:** Elimina un producto específico del carrito.
- **Parámetros:**
  - `id` (requerido): ID del producto a eliminar.
- **Respuesta:** (sin contenido)

**Status 204:**

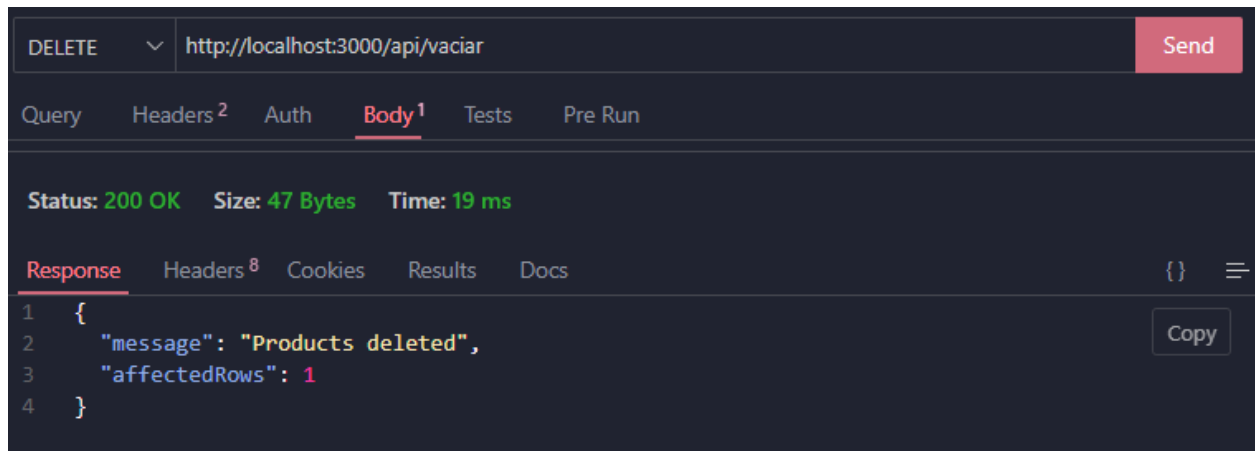
```
// No hay respuesta, solo el código de estado 204.
```



## 6. Vaciar el Carrito

- URL: `/vaciar/`
- Método: `DELETE`
- Descripción: Elimina todos los productos del carrito.
- Respuesta:

```
{
  "message": "Products deleted",
  "affectedRows": 5
}
```



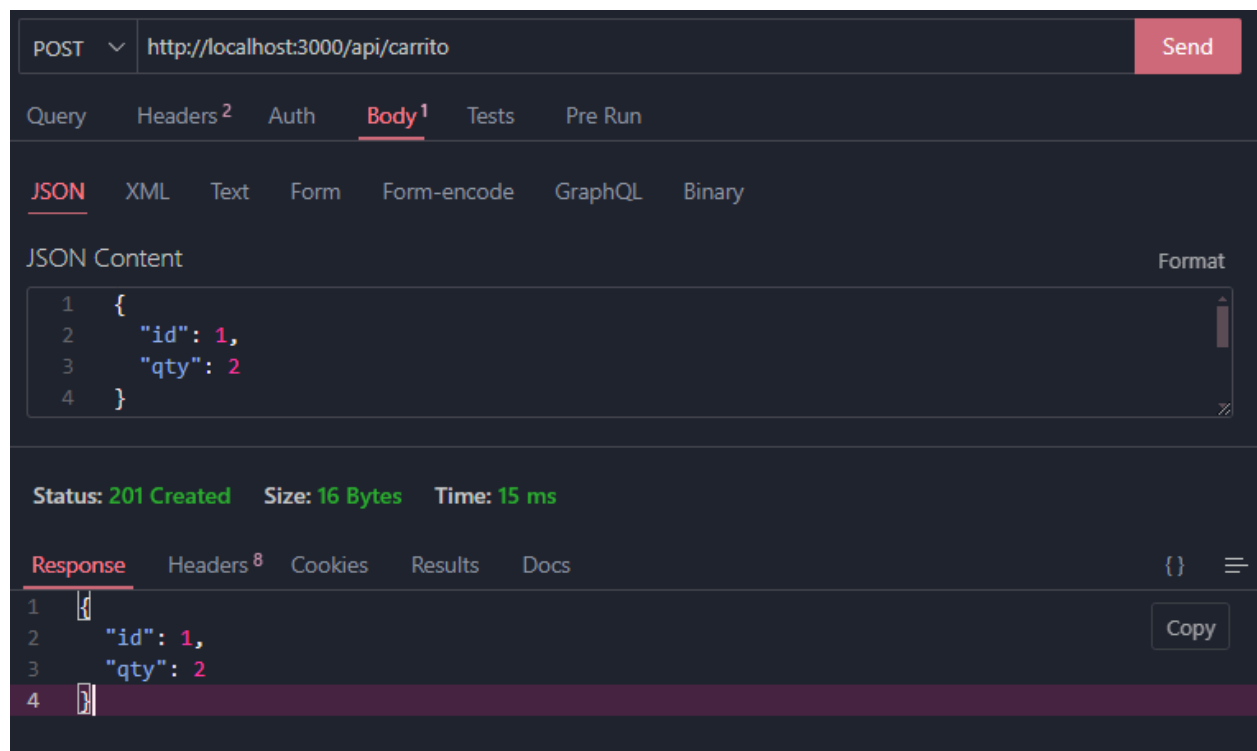
## 7. Agregar Producto al Carrito

- URL: `/carrito`
- Método: `POST`
- Descripción: Agrega un producto al carrito con una cantidad específica.
- Body:

```
{
  "id": 1,
  "qty": 2
}
```

- **Respuesta:**

```
{
  "id": 1,
  "qty": 2
}
```



## Headers

En general, no se requiere ningún header especial para las solicitudes, excepto cuando se realiza una solicitud de tipo **POST**, **PUT** o **DELETE**, en cuyo caso se debe incluir el siguiente header para indicar que el cuerpo de la solicitud está en formato JSON:

```
Content-Type: application/json
```



## **Autenticación**

Este API no requiere autenticación (token o contraseña). Sin embargo, para ambientes de producción, se recomienda implementar autenticación adecuada (como JWT) para proteger los endpoints, especialmente para operaciones sensibles como eliminar productos o vaciar el carrito.

## Referencias

Express.js. *Express API Reference* (v5.x). Express.js. <https://expressjs.com/en/5x/api.html>