

# Documentación de API de coches

- Esta es una API que permite realizar operaciones de crear, leer, actualizar y eliminar sobre una base de datos de coches. La API está construida con Flask y utiliza una base de datos MySQL.
- La base de datos se llama cochesapi, la única tabla que tiene es “coches”. Está hecha utilizando MySQL y la herramienta PHP My Admin.

## Endpoints:

GET: <http://127.0.0.1:5000/coches>

Descripción: Retorna todos los coches guardados en la base de datos.

Respuesta exitosa:

```
[
  {
    "año": 2009,
    "bastidor": 1232,
    "id": 1,
    "modelo": "MX-5",
    "titular": "Aleksandr Lilie"
  },
  {
    "año": 2002,
    "bastidor": 1234,
    "id": 3,
    "modelo": "Expedition",
    "titular": "Maurene Erik"
  },
  {
    "año": 2000,
    "bastidor": 0,
    "id": 4,
    "modelo": "Stratus",
    "titular": "Raf Quilty"
  },
]
```

GET: <http://127.0.0.1:5000/coches/<int:id>>

Ejemplo: <http://127.0.0.1:5000/coches/3>

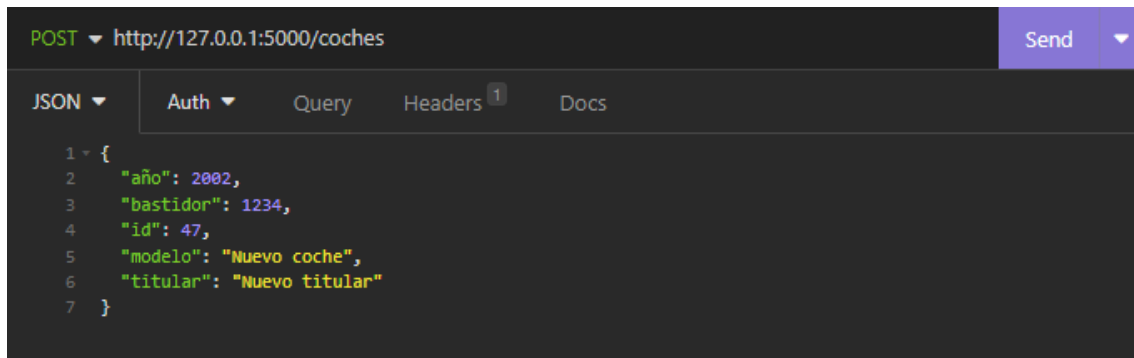
Descripción: Retorna el coche con el id 3.

Respuesta exitosa:

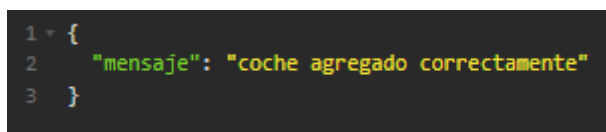
```
{
  "año": 2002,
  "bastidor": 1234,
  "id": 3,
  "modelo": "Expedition",
  "titular": "Maurene Erik"
}
```

POST: <http://127.0.0.1:5000/coches>

Descripción: agrega un nuevo coche a la base de datos a través de la aplicación Insomnia pasándole un cuerpo en JSON de la siguiente manera:



Respuesta exitosa:



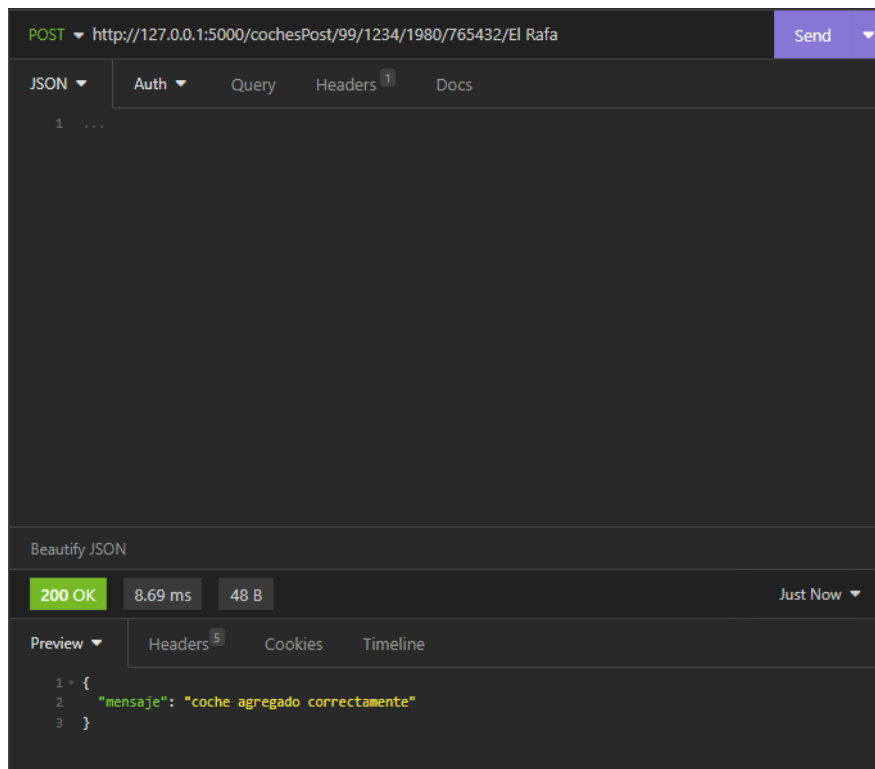
POST:

<http://127.0.0.1:5000/cochesPost/<int:id>/<modelo>/<int:anio>/<bastidor>/<titular>>

Descripción: agrega un nuevo coche a la base de datos a través de una URL. La URL espera los siguientes parámetros:

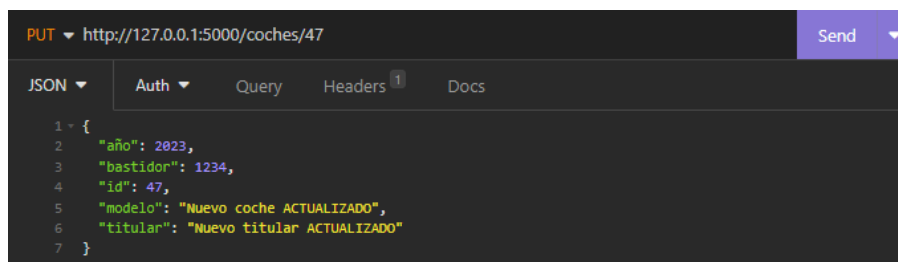
- id: el ID del coche (entero)
- modelo: el modelo del coche (cadena de texto)
- anio: el año del coche (entero)
- bastidor: el número de bastidor del coche (cadena de texto)
- titular: el nombre del titular del coche (cadena de texto)

Ejemplo: `http://127.0.0.1:5000/cochesPost/96/1234/1980/765432/El Rafa`

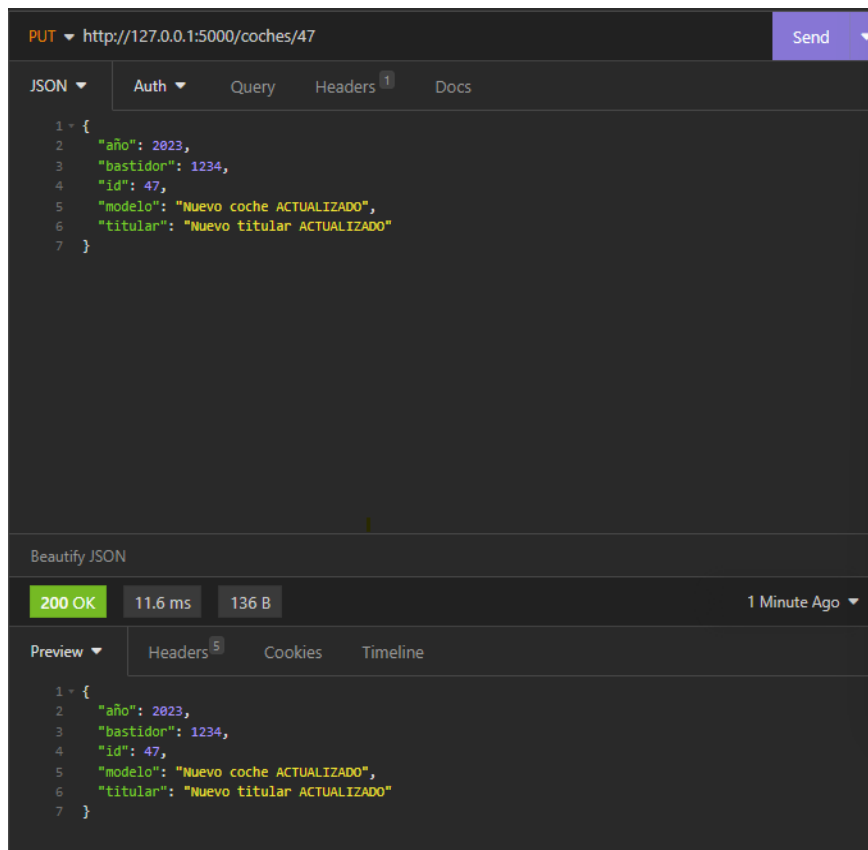


PUT: <http://127.0.0.1:5000/coches/<int:id>>

**Descripción:** Toma como parámetro el ID del coche a actualizar a través de la URL y agrega el coche actualizado a la base de datos a través de la aplicación Insomnia pasándole un cuerpo en JSON de la siguiente manera:

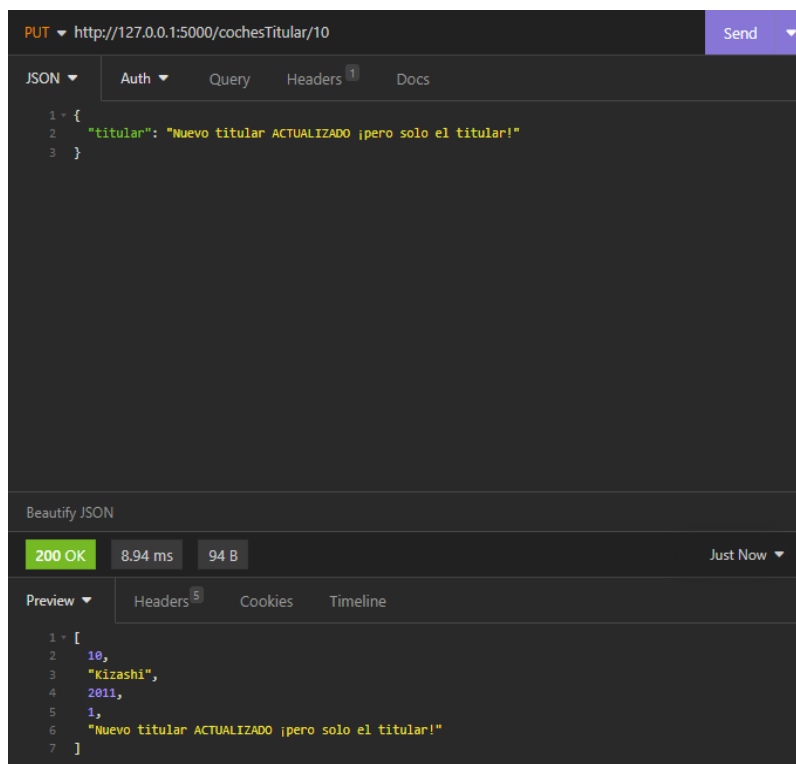


En este caso la respuesta exitosa es la propia actualización:



PUT: <http://127.0.0.1:5000/cochesTitular/<int:id>>

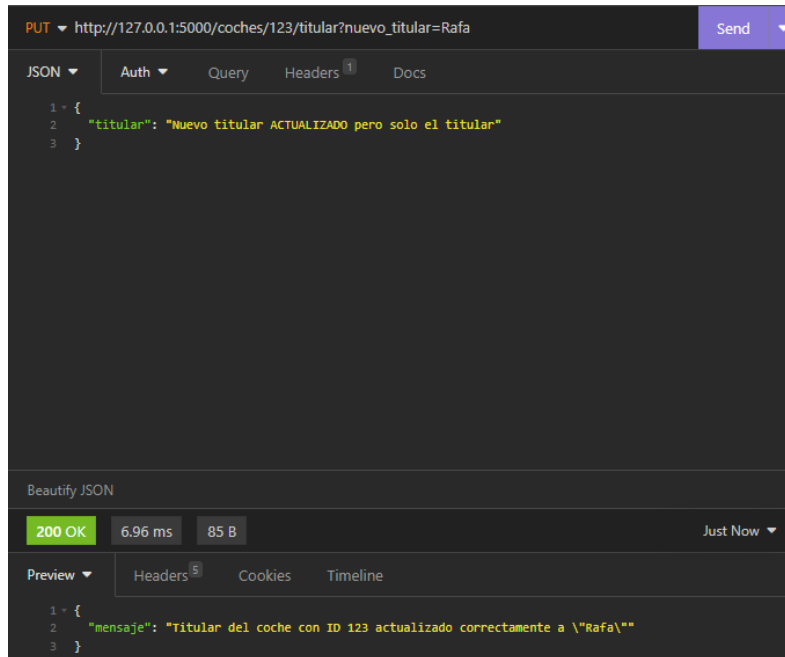
Descripción: Permite actualizar el titular de un coche en la base de datos a través de su id.



PUT: <http://127.0.0.1:5000/coches/<int:id>/titular>

**IMPORTANTE:** La URL para actualizar el titular del coche a Rafa con ID 123 sería:  
[/coches/123/titular?nuevo\\_titular=Rafa](http://127.0.0.1:5000/coches/123/titular?nuevo_titular=Rafa)

**Descripción:** Permite actualizar el titular de un coche en la base de datos. El parámetro id en la ruta especifica el ID del coche que se quiere actualizar. El nuevo titular se obtiene de un parámetro de la URL llamado nuevo titular.



DELETE: <http://127.0.0.1:5000/cochesDelete/<int:id>>

**Descripción:** define la ruta para eliminar un coche de la base de datos utilizando su ID.

