



# JUSTIFICACIONES POST-MAN

Examen Spring Boot Pablo Rivero

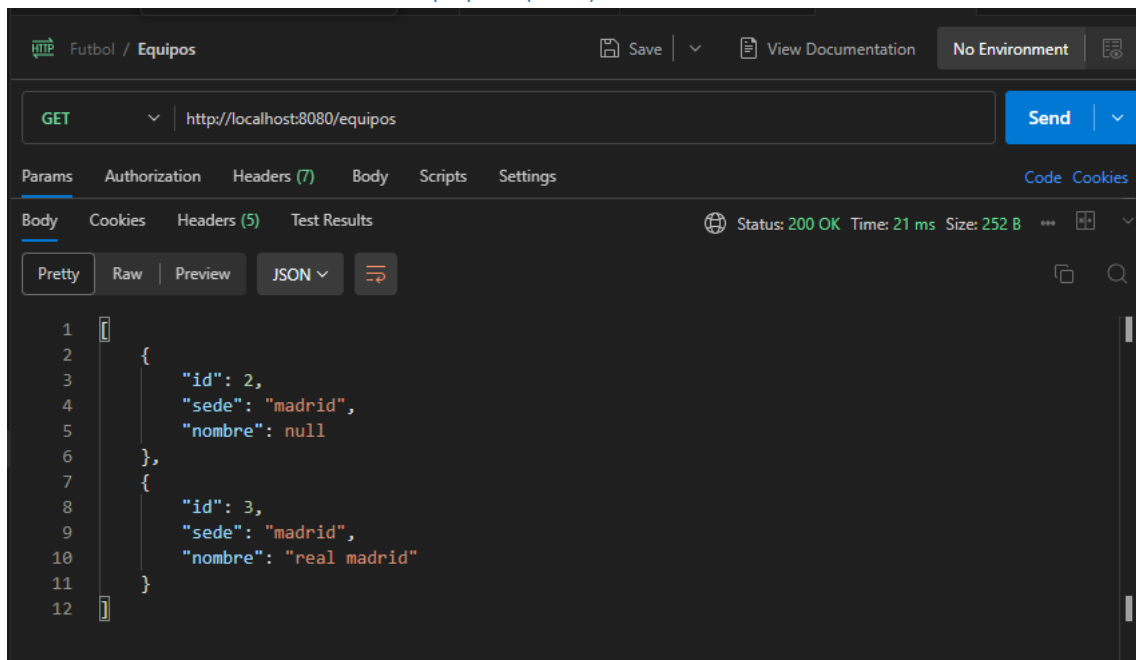
## Contenido

EQUIPO.....	2
1- Poder listar todos los equipos (GET) .....	2
2- Añadir un equipo (POST) .....	3
3- Modificar el nombre de un equipo (PUT) .....	3
4- Eliminar un equipo .....	4
JUGADORES.....	5
1- Listar jugadores (GET) .....	5
2- Listar jugador por equipo (GET) .....	6
3- Poder añadir jugadores .....	<b>¡Error! Marcador no definido.</b>
ARBITROS .....	6
1- Listar los árbitros (GET) .....	6

\*Para los métodos no he añadido variables en el postman, igualmente te paso una captura al final del documento de como serian.

## EQUIPO

### 1- Poder listar todos los equipos (GET)



\*El segundo equipo tiene nombre null por problemas con unas pruebas.

## 2- Añadir un equipo (POST)

The screenshot shows a REST client interface for a project named 'Futbol'. The active tab is 'AddEquipo'. The request is a POST to 'http://localhost:8080/equipos'. The body is set to 'raw' and contains the following JSON:

```
{
  "nombre": "CD-Laredo",
  "sede": "Laredo"
}
```

The response status is 200 OK, with a time of 65 ms and a size of 209 B. The response body is shown in 'Pretty' JSON format:

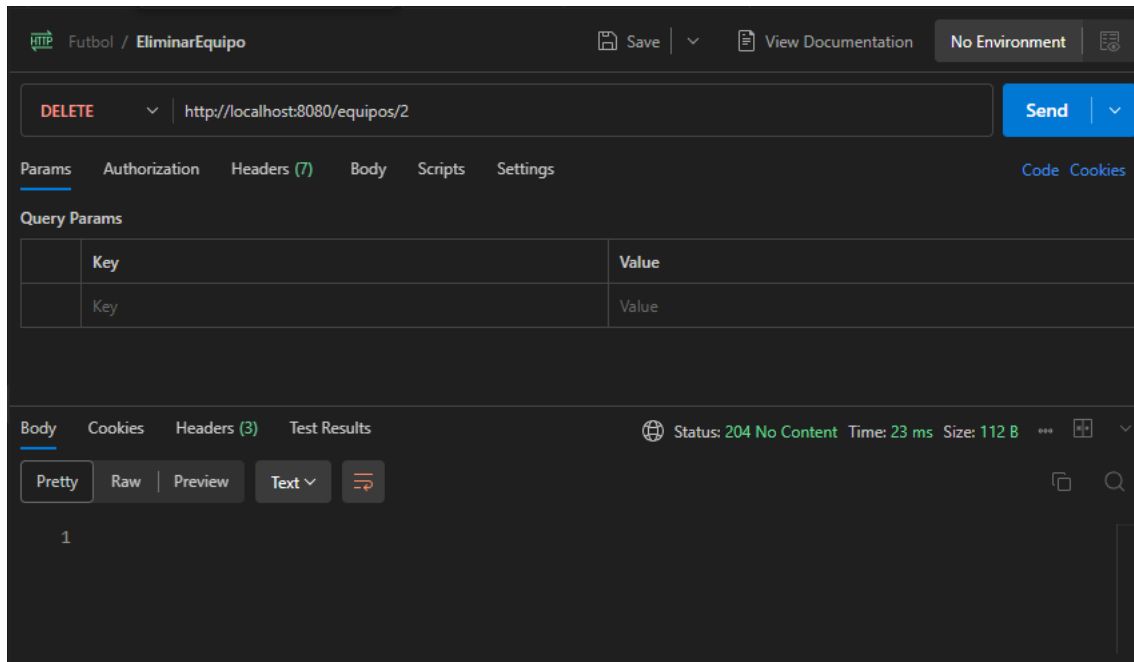
```
{
  "id": 4,
  "sede": "Laredo",
  "nombre": "CD-Laredo"
}
```

## 3- Modificar el nombre de un equipo (PUT)

The screenshot shows a REST client interface for a project named 'Futbol'. The active tab is 'EditarNombreEquipo'. The request is a PUT to 'http://localhost:8080/equipos/2'. The body is set to 'raw' and contains the text 'real madrid'. The response status is 200 OK, with a time of 34 ms and a size of 211 B. The response body is shown in 'Pretty' JSON format:

```
{
  "id": 2,
  "sede": "madrid",
  "nombre": "real madrid"
}
```

#### 4- Eliminar un equipo

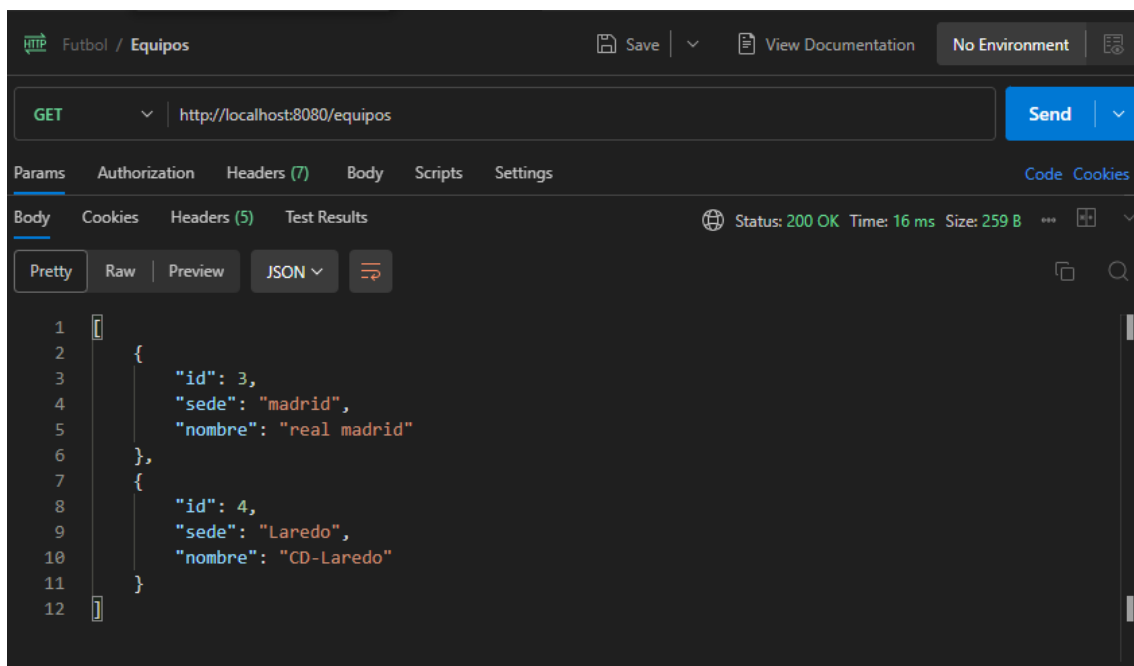


Postman interface showing a DELETE request to `http://localhost:8080/equipos/2`. The status is **204 No Content**, Time: 23 ms, Size: 112 B.

Query Params:

Key	Value
Key	Value

Body: 1



Postman interface showing a GET request to `http://localhost:8080/equipos`. The status is **200 OK**, Time: 16 ms, Size: 259 B.

Body (JSON):

```
1 {
2   {
3     "id": 3,
4     "sede": "madrid",
5     "nombre": "real madrid"
6   },
7   {
8     "id": 4,
9     "sede": "Laredo",
10    "nombre": "CD-Laredo"
11  }
12 }
```

\*Paso captura del get para comprobar que el equipo se ha borrado bien.

# JUGADORES

## 1- Listar jugadores (GET)

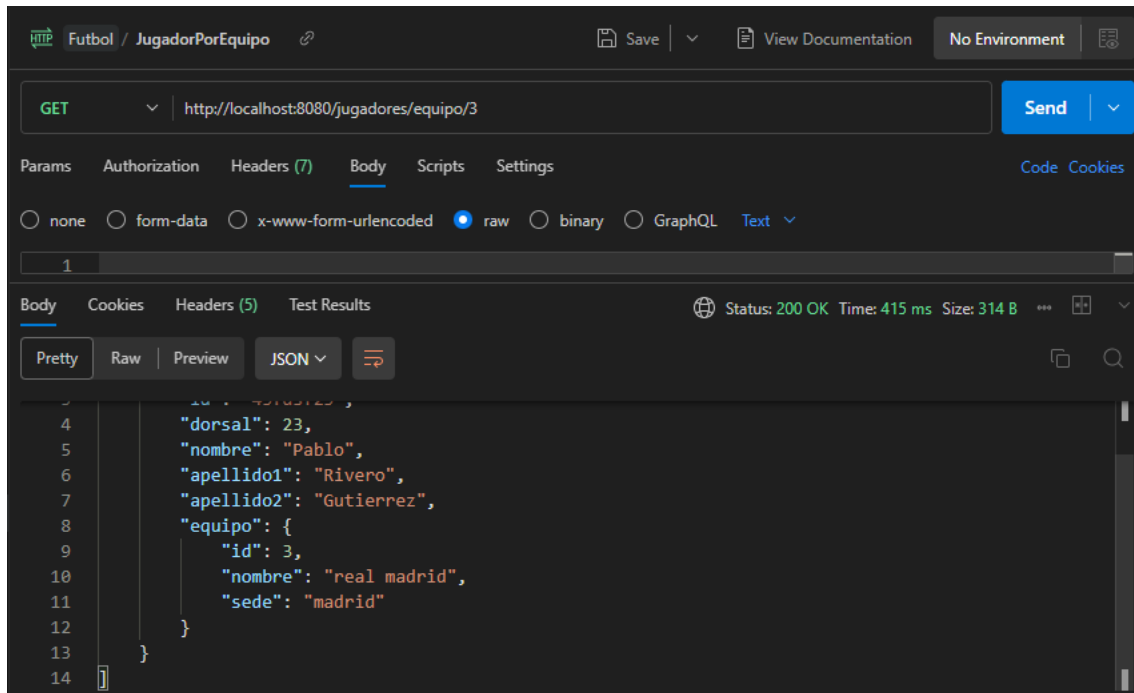
The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** http://localhost:8080/jugadores
- Status:** 200 OK
- Time:** 26 ms
- Size:** 314 B

The response body is displayed in JSON format:

```
1 [{"id": "43fdsf23", "dorsal": 23, "nombre": "Pablo", "apellido1": "Rivero", "apellido2": "Gutierrez", "equipo": {"id": 3, "nombre": "real madrid", "sede": "madrid"}}]
```

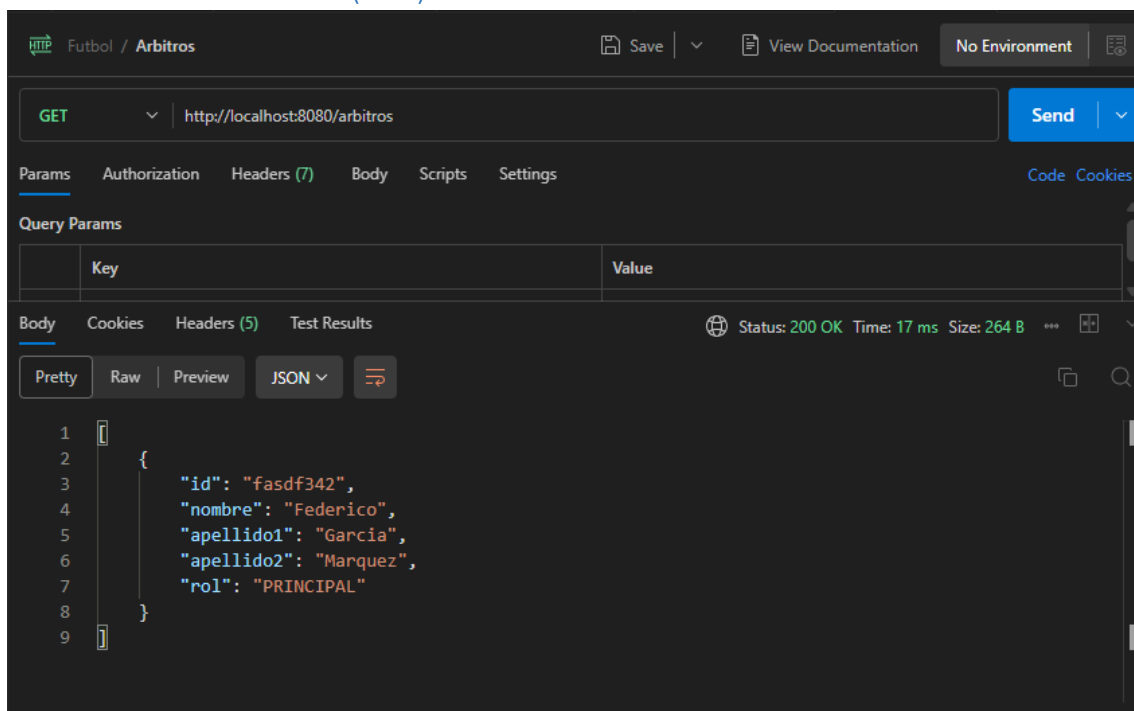
## 2- Listar jugador por equipo (GET)



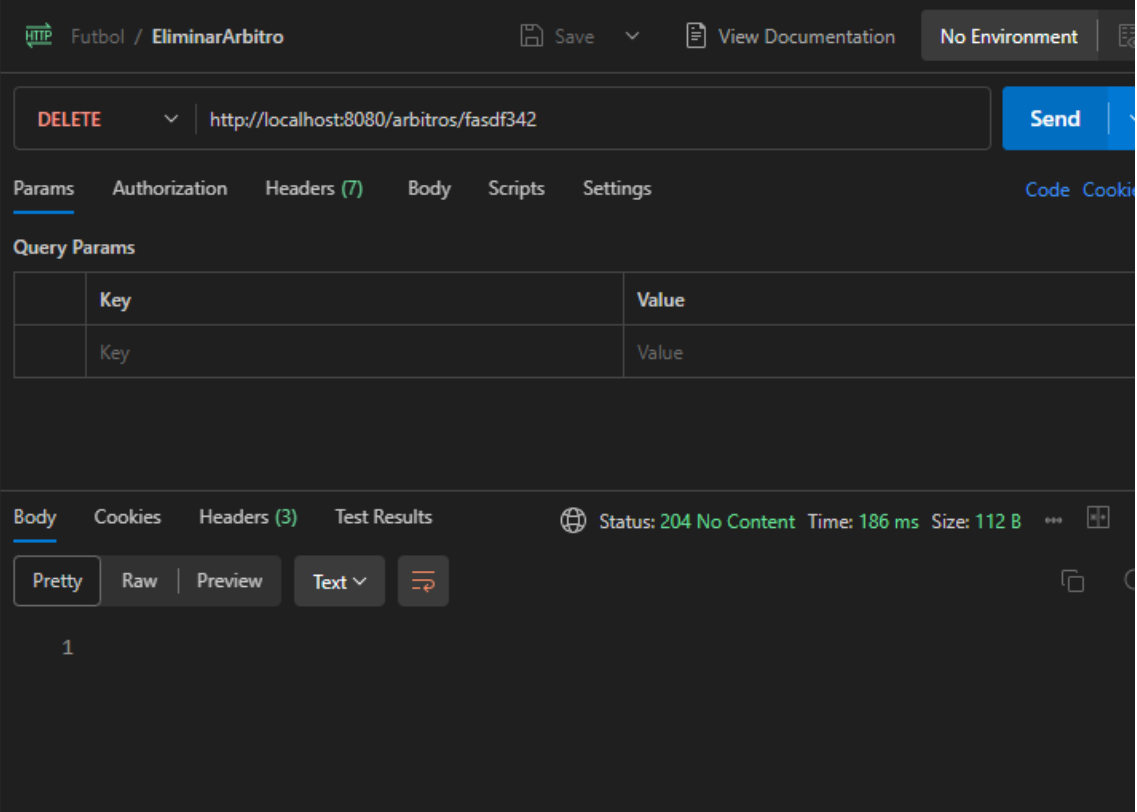
\*Lo he hecho por id ya que el método con string me daba fallo al estar los caracteres separados

## ARBITROS

### 1- Listar los árbitros (GET)



## 2- Eliminar Arbitro



HTTP Futbol / **EliminarArbitro** Save View Documentation No Environment

DELETE http://localhost:8080/arbitros/fasdf342 Send

Params Authorization Headers (7) Body Scripts Settings Code Cookies

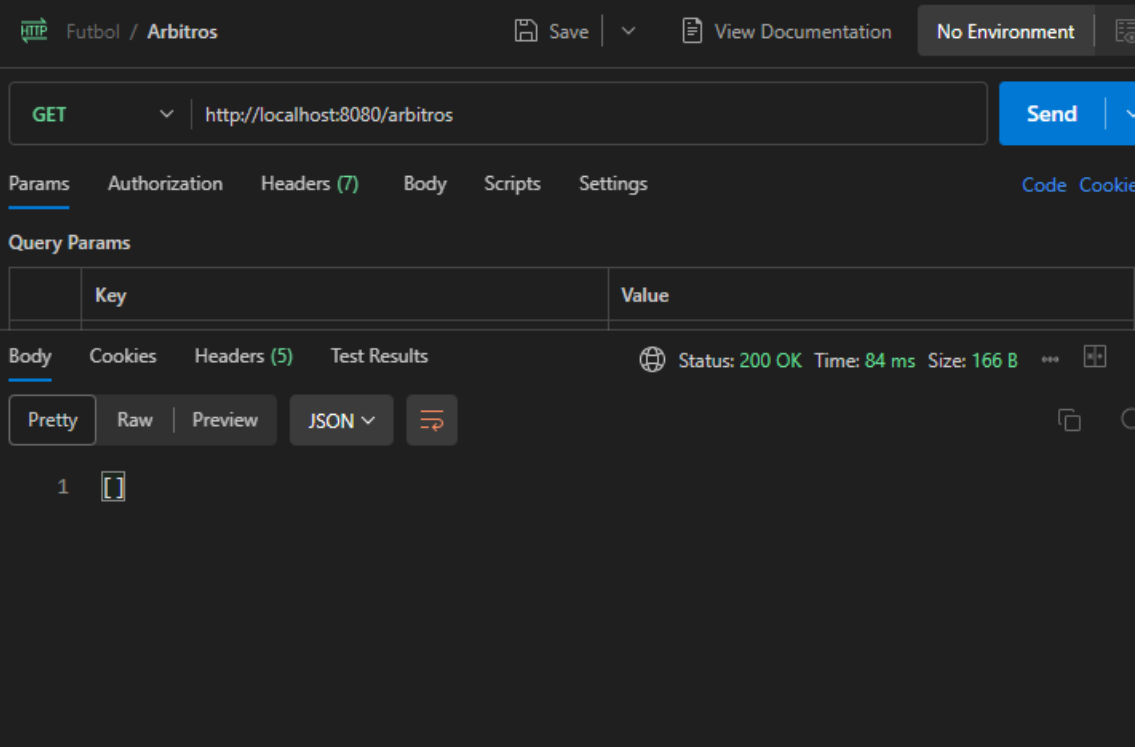
Query Params

Key	Value
Key	Value

Body Cookies Headers (3) Test Results Status: 204 No Content Time: 186 ms Size: 112 B

Pretty Raw Preview Text

1



HTTP Futbol / **Arbitros** Save View Documentation No Environment

GET http://localhost:8080/arbitros Send

Params Authorization Headers (7) Body Scripts Settings Code Cookies

Query Params

Key	Value
-----	-------

Body Cookies Headers (5) Test Results Status: 200 OK Time: 84 ms Size: 166 B

Pretty Raw Preview JSON

1

\*Get para comprobar que se han eliminado