



Trabajo final

Martín García Chagüezá

Juan José Orozco López

▼ *Lineamientos*

https://prod-files-secure.s3.us-west-2.amazonaws.com/46472d32-9223-434f-9051-a8297ec5dbaf/386a2562-e852-4fcf-b63e-e353ae12f154/Trabajo_Final_2024_02.docx.pdf

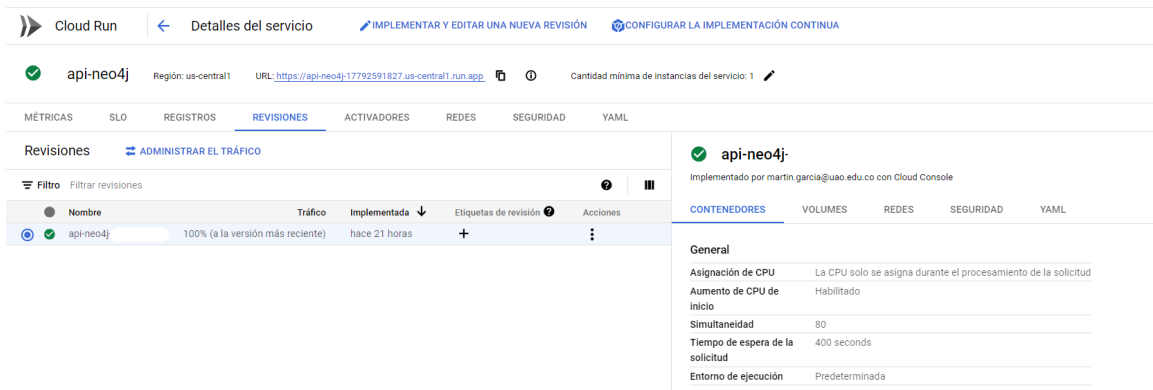
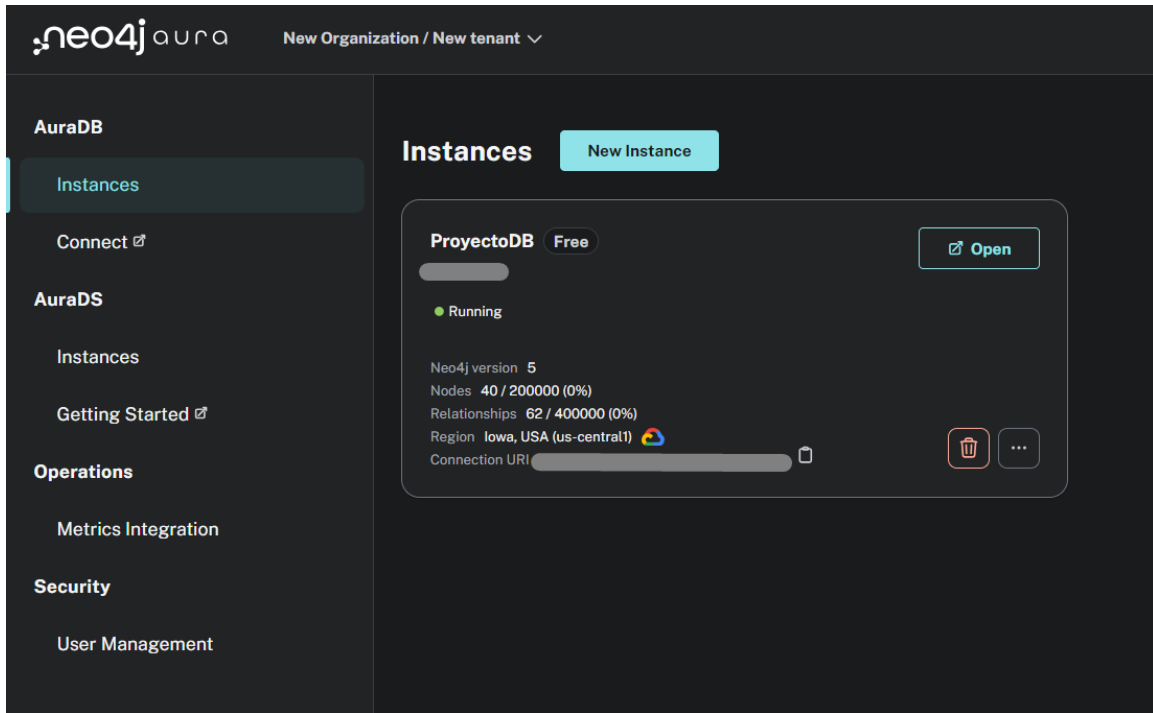
▼ *Repositorio de Github*

<https://github.com/mitgar14/almacenamiento-final>

Introducción

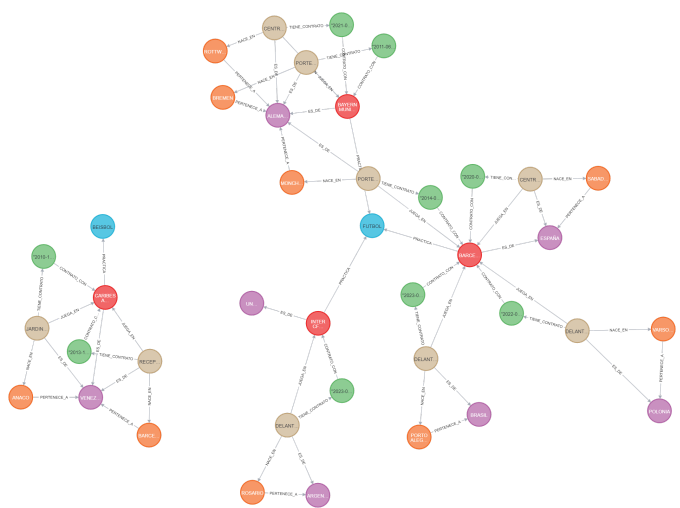
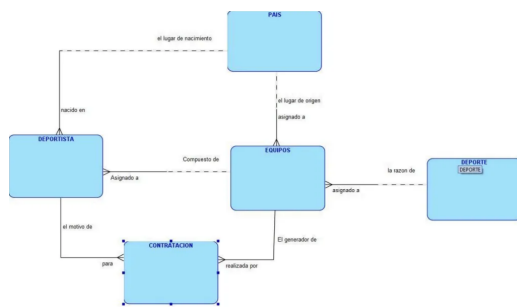
En este trabajo realizamos la construcción de un modelo basado en nodos usando Neo4j que busca representar a *deportistas*, *equipos* y *contratos* mediante relaciones definidas.

Para obtener la información de esta base de datos se procedió a la creación de una API vinculada a nuestra instancia de **Neo4j AuraDB** (*base de datos en la nube*) que, mediante Docker, fue desplegada en el servicio de Google Cloud **Cloud Run** (*servicio serverless*).



Modelo realizado en Neo4j

El modelo que se buscaba implementar en la base de datos tenía que obedecer al MER que se muestra. Siguiendo los requerimientos, nuestro producto fue el siguiente:



País

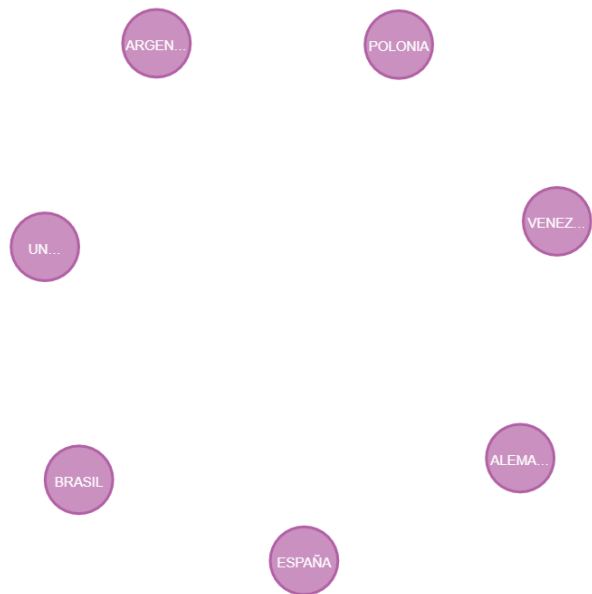
En este caso se asegura la creación de nodos únicos con la etiqueta **País**. Cada comando establece un nodo representando un país específico mediante la propiedad **nombre**. En resumen:

- **MERGE**: Crea el nodo si no existe o lo encuentra si ya está presente.
- **:País**: Etiqueta que clasifica el nodo como un país.
- **{nombre: "..."}** : Define la propiedad **nombre** del nodo con el valor correspondiente al país.

```

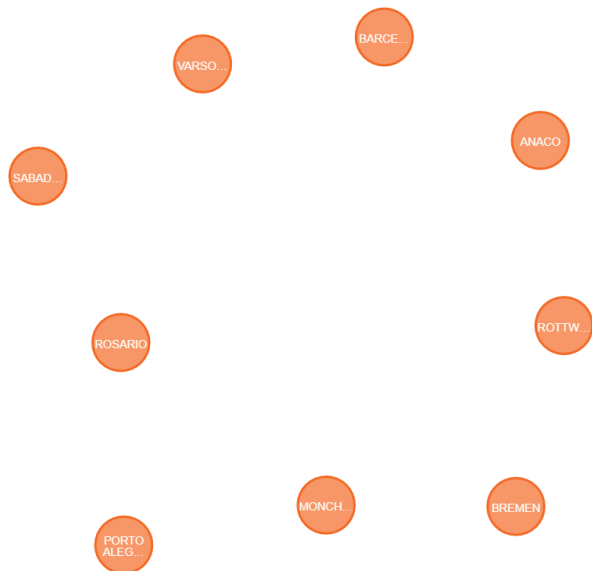
MERGE (:Pais {nombre: "ESPAÑA
MERGE (:Pais {nombre: "ALEMAN
MERGE (:Pais {nombre: "VENEZU
MERGE (:Pais {nombre: "POLONI

```



Ciudad → relacionado con *Países*

En esta sección se crean nodos de tipo **Ciudad** y se relacionan con el nodo correspondiente del país. A continuación, se explica cada parte de los comandos tomando como ejemplo a *Alemania*:



Encontrar el nodo del *País*

```
MATCH (paisAlemania:Pais {nombre: "ALEMANIA"})
```

- **MATCH** : Busca nodos existentes que cumplan con el patrón especificado.
- **(paisAlemania:Pais {nombre: "ALEMANIA"})** : Encuentra el nodo etiquetado como **Pais** con la propiedad **nombre** igual a "ALEMANIA" y lo asigna a la variable **paisAlemania** para su uso posterior en las relaciones.

Crear *ciudades* y relacionarlas con el *país*

```
MERGE (c1:Ciudad {nombre: "MONCHENGLADBACH"}) - [:PERTENECE_A] ->(paisAlemania)
MERGE (c9:Ciudad {nombre: "BREMEN"}) - [:PERTENECE_A] ->(paisAlemania)
MERGE (c13:Ciudad {nombre: "ROTTWEIL"}) - [:PERTENECE_A] ->(paisAlemania)
```

- **MERGE** : Asegura que cada nodo de **Ciudad** con el nombre especificado exista una única vez y crea la relación **PERTENECE_A** con **paisAlemania** si no existe.
- **(:Ciudad {nombre: "..."})** : Define un nodo con la etiqueta **Ciudad** y una propiedad **nombre** con el valor correspondiente a la ciudad.
- **[:PERTENECE_A]->** : Establece una relación dirigida de la ciudad al país, indicando que la ciudad **pertenece a** ese país.
- **Variables (c1, c9, c13)** : Asignan nombres únicos a cada nodo de ciudad para facilitar su referencia, aunque en este contexto específico no se reutilizan posteriormente.

Deporte

Se crean nodos para representar diferentes *deportes* utilizando los siguientes comandos:

```
MERGE (:Deporte {nombre: "FUTBOL"});
MERGE (:Deporte {nombre: "BEISBOL"});
```

Estos comandos generan nodos etiquetados como `Deporte`, cada uno identificado de manera única por la propiedad `nombre`.

La etiqueta `:Deporte` clasifica cada nodo específicamente como un deporte, facilitando su identificación y posterior relación con otras entidades dentro del grafo, como equipos.

La propiedad `nombre` actúa como un identificador distintivo para cada deporte, permitiendo diferenciar claramente entre ellos y asegurar una representación precisa de las diversas disciplinas deportivas en la base de datos.



Equipo → relacionado con *Deportes y Países*

Se procede a la creación de nodos que representan diferentes equipos deportivos y a establecer sus relaciones con los deportes y los países correspondientes. A continuación, se presentan los comandos utilizados:

```
// Crear Equipos
MERGE (:Equipo {nombre: "BARCELON
A"});
MERGE (:Equipo {nombre: "BAYERN MUN
ICH"});
MERGE (:Equipo {nombre: "CARIBES DE
ANZOATEGUI"});

// Relacionar equipos con deportes
y países

// Barcelona
MATCH (eBarcelona:Equipo {nombre:
"BARCELONA"}),
```



```

(dFutbol:Deporte {nombre: "FUTBOL"}),
(paisEspaña:Pais {nombre: "ESPAÑA"})
MERGE (eBarcelona)-[:PRACTICA]->(dFutbol)
MERGE (eBarcelona)-[:ES_DE]->(paisEspaña);

```

Creación de Equipos

Se crean nodos etiquetados como `Equipo` para "BARCELONA", "BAYERN MUNICH" y "CARIBES DE ANZOATEGUI" utilizando la sentencia `MERGE`. Cada nodo se identifica de manera única por la propiedad `nombre`.

Establecimiento de relaciones

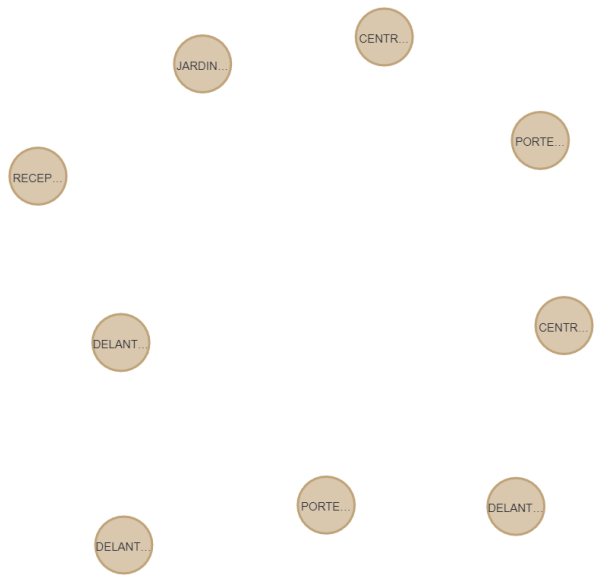
Mediante la sentencia `MATCH`, se localizan los nodos correspondientes al equipo "BARCELONA", al deporte "FUTBOL" y al país "ESPAÑA".

- Se establecen dos relaciones:
 - `PRACTICA`: Conecta el equipo "BARCELONA" con el deporte "FUTBOL", indicando la práctica de dicho deporte por parte del equipo.
 - `ES_DE`: Conecta el equipo "BARCELONA" con el país "ESPAÑA", indicando la pertenencia geográfica del equipo.

Deportista → relacionado con Ciudad, País, Equipo y Contrato

Se aborda la creación de nodos que representan a los **jugadores** deportivos y sus relaciones con las *ciudades, países y equipos* correspondientes.

A continuación, se presentan los comandos utilizados para este propósito:



```
MATCH (c:Ciudad {nombre: "VARSOVIA"})-[:PERTENECE_A]->(pais:Pais {nombre: "POLONIA"}),
      (eBarcelona:Equipo {nombre: "BARCELONA"})
CREATE (d:Deportista {
  nombre: "ROBERT LEWANDOWSKI",
  posicion: "DELANTERO",
  dorsal: 9,
  sexo: "MASCULINO"
})
CREATE (d)-[:NACE_EN {fecha_nacimiento: date("1988-08-21")}]>(c)
CREATE (d)-[:ES_DE]->(pais)
CREATE (d)-[:JUEGA_EN {fecha_vinculacion: date("2022-07-19")}]>(eBarcelona)
```

Localización de entidades existentes

Se utiliza **MATCH** para encontrar el nodo de la ciudad "VARSOVIA" que pertenece al país "POLONIA" y el nodo del equipo "BARCELONA". Esto establece las entidades con las que el jugador estará relacionado.

Creación del nodo *Deportista*

Se crea un nodo etiquetado como `Deportista` con propiedades específicas como `nombre`, `posicion`, `dorsal` y `sexo`, representando al jugador Robert Lewandowski.

Establecimiento de relaciones

- `NACE_EN`: Conecta al deportista con la ciudad "VARSOVIA", indicando su lugar de nacimiento y registrando la fecha de nacimiento.
- `ES_DE`: Asocia al deportista con el país "POLONIA", especificando su nacionalidad.
- `JUEGA_EN`: Vincula al deportista con el equipo "BARCELONA", indicando el inicio de su vinculación con el equipo a través de la fecha correspondiente.

Estos comandos permiten estructurar la información relativa a los jugadores dentro del grafo, estableciendo conexiones claras entre los deportistas, sus lugares de origen y los equipos para los que juegan.

Contrato → relacionado con *Equipo*

Ahora se procede a la creación de contratos que vinculan a los deportistas con los equipos deportivos.

A continuación, se presentan los comandos utilizados para esta sección:

```
CREATE (contrato:Contrato {  
  fecha_inicio: date("2022-07-19"),  
  fecha_fin: date("2026-06-30"),
```

```
        valor_contrato: 50000000
    })
    CREATE (d)-[:TIENE_CONTRATO]->(contrato)
    CREATE (contrato)-[:CONTRATO_CON]->(eBarcelona);
```

Creación del Nodo Contrato:

Se crea un nodo etiquetado como `Contrato` con las propiedades `fecha_inicio`, `fecha_fin` y `valor_contrato`, que representan los términos del acuerdo entre el deportista y el equipo.

Establecimiento de Relaciones:

- `TIENE_CONTRATO`: Conecta al deportista (`d`) con el contrato, indicando que el deportista posee dicho contrato.
- `CONTRATO_CON`: Vincula el contrato con el equipo "BARCELONA", especificando que el contrato está asociado con este equipo.

API

La API creada para este trabajo se puede acceder (*hasta @14 de diciembre de 2024*) por medio del siguiente link → <https://api-neo4j-17792591827.us-central1.run.app>

Los endpoints para poder acceder a ella son los siguientes:

▼ Equipos

▼ **GET** - Todos los equipos → `/api/equipos/`

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

- Method:** GET
- URL:** `{{base_url}}/api/equipos/`
- Status:** 200 OK
- Response Time:** 843 ms
- Response Size:** 750 B
- Response Format:** JSON
- Response Body (Pretty):**

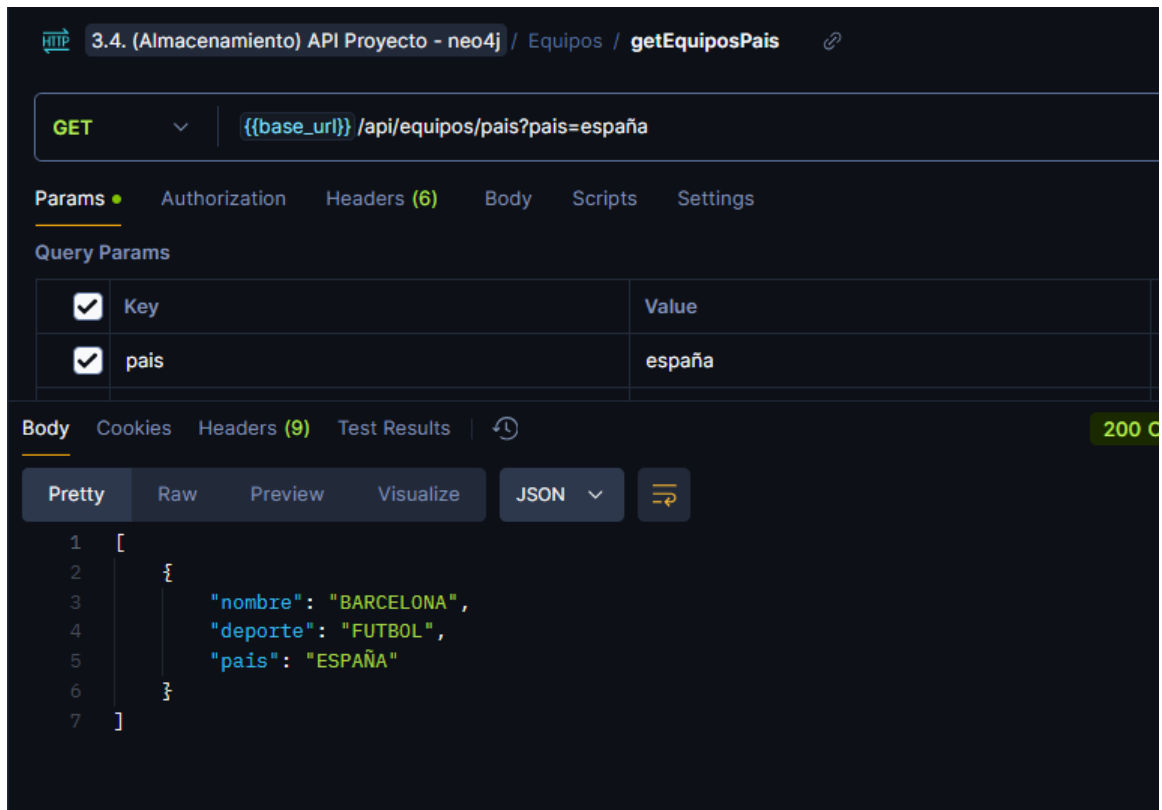
```
[
  {
    "id": 278,
    "nombre": "BARCELONA",
    "pais": "ESPAÑA",
    "deporte": "FUTBOL",
    "num_deportistas": 4
  },
  {
    "id": 279,
    "nombre": "BAYERN MUNICH",
    "pais": "ALEMANIA",
    "deporte": "FUTBOL",
    "num_deportistas": 2
  },
  {
    "id": 280,
    "nombre": "CARIBES DE ANZOATEGUI",
    "pais": "VENEZUELA",
    "deporte": "BEISBOL",
    "num_deportistas": 2
  },
  {
    "id": 316,
    "nombre": "INTER MIAMI CF",
    "pais": "ESTADOS UNIDOS",
    "deporte": "FUTBOL",
    "num_deportistas": 1
  }
]
```

▼ **GET** - Todos los equipos de un país → `/api/equipos/pais?pais=<pais>`

Query

Clave → pais

Valor → el país a buscar



▼ GET - Todos los equipos de un deporte →

`/api/equipos/deporte?deporte=<deporte>`

Query

Clave → deporte

Valor → el deporte a buscar

3.4. (Almacenamiento) API Proyecto - neo4j / Equipos / **getEquiposDeporte**

GET `{{base_url}}/api/equipos/deporte?deporte=beisbol`

Params • Authorization Headers (6) Body Scripts Settings

Query Params

<input checked="" type="checkbox"/>	Key	Value
<input checked="" type="checkbox"/>	deporte	beisbol
	Key	Value

Body Cookies Headers (9) Test Results | 200

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "nombre": "CARIBES DE ANZOATEGUI",
4     "pais": "VENEZUELA",
5     "deporte": "BEISBOL"
6   }
7 ]
```

▼ **GET - Equipo por su ID** → `/api/equipos/:id`

Parámetros

id → el ID del equipo

HTTP 3.4. (Almacenamiento) API Proyecto - neo4j / Equipos / getEquipoid

GET {{base_url}}/api/equipos/278

Params Authorization Headers (6) Body Scripts Settings

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (9) Test Results 200 OK • 564 ms • 430 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 278,
3   "nombre": "BARCELONA",
4   "pais": "ESPAÑA",
5   "deporte": "FUTBOL"
6 }
```

▼ POST - Crear equipo → /api/equipos/



Consideraciones

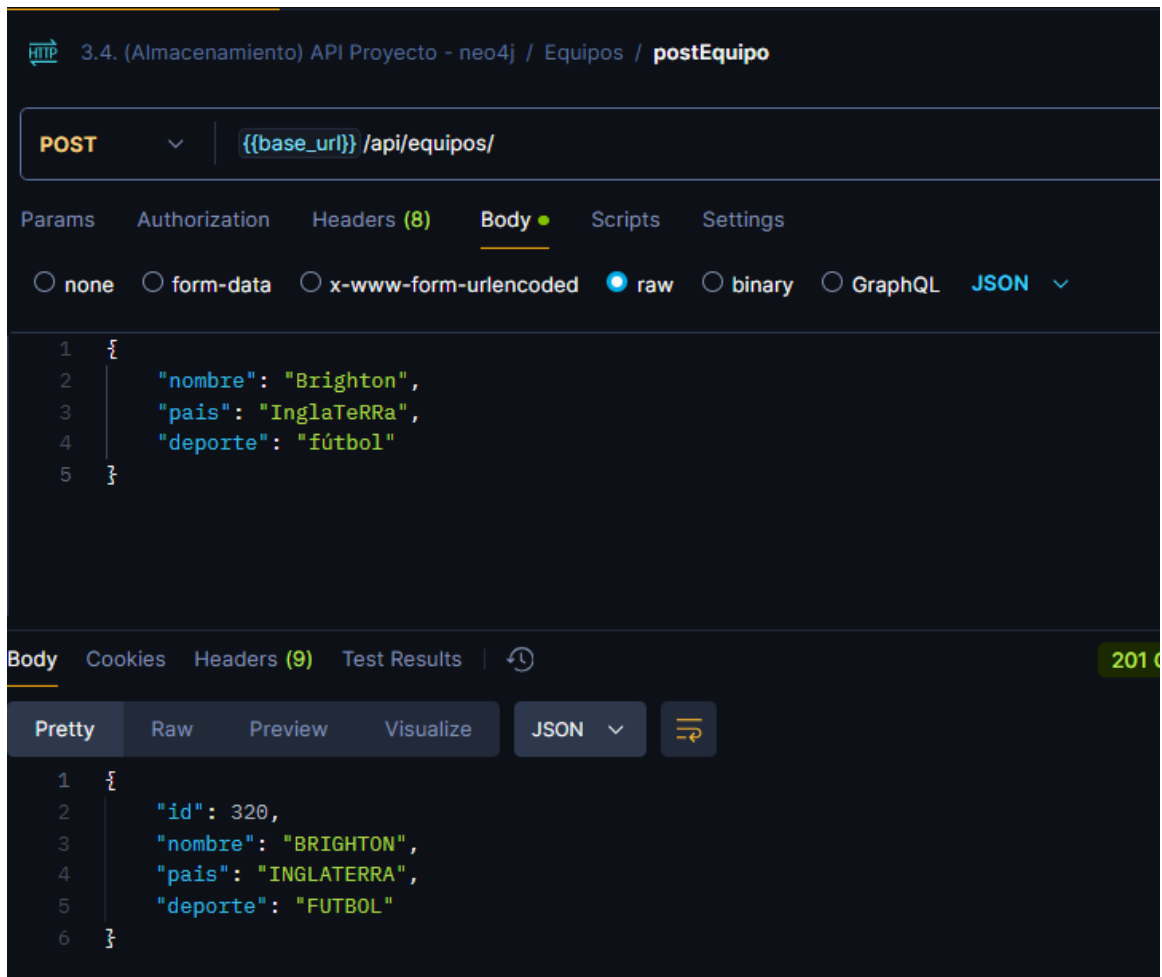
- El formato del body debe ser en JSON, tal cual como se observa en la imagen.
- No hay un estándar para escribir los valores en ninguno de los POSTs, lo importante es que luego se mantengan los mismos valores → *que no sea que de Estados Unidos pasen a escribir USA o EE.UU.*
- Este POST se encarga de **crear un nodo para el país** (y sus respectivas relaciones), a pesar de que este no exista.
 - Sin embargo, no se encarga de crear un nodo para el *deporte*; los requerimientos de la actividad no lo hace necesario. **Solamente se crea la relación entre estos dos.**

Body

nombre → nombre del equipo

pais → país actual del equipo

deporte → deporte que ejerce el equipo, principalmente



▼ **PUT** - Modificar equipo por su ID → `/api/equipos/:id`



Consideraciones

- El formato del *body* debe ser en JSON, tal cual como se observa en la imagen.
- No hay un estándar para escribir los valores en ninguno de los *PUTs*, lo importante es que luego se mantengan los mismos valores → *que no sea que de Estados Unidos pasen a escribir USA o EE.UU.*
- No hay restricciones en la cantidad de campos que se pueden manejar en el *body*: de las 3 claves, se pueden modificar tranquilamente solo 1 o 2 de ellas.
 - Sin embargo, no se recomienda modificar el campo de país ni de deporte dado que no hay un manejo para la actualización de los nodos al que "*pais*" y "*deporte*" se encuentran vinculados.

Parámetros

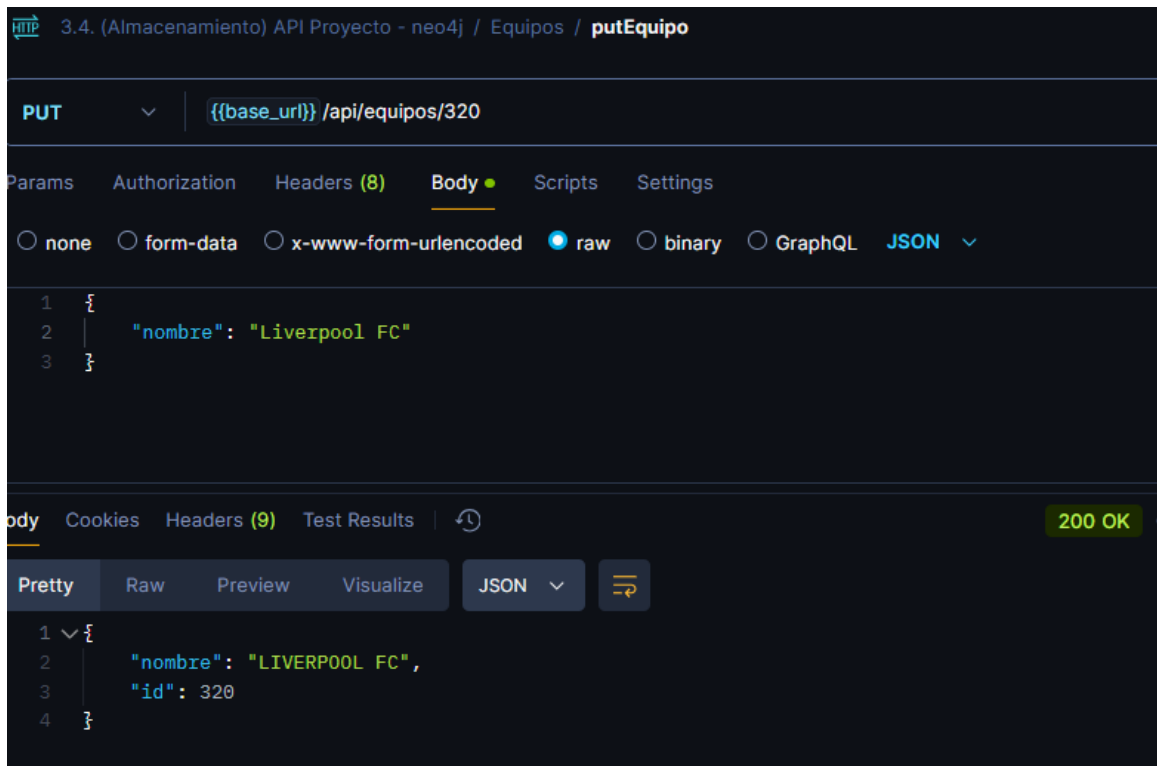
id → el ID del equipo a modificar

Body (cada campo es opcional)

nombre → nombre del equipo

pais (*bajo precaución*) → país actual del equipo

deporte (*bajo precaución*) → deporte que ejerce el equipo, principalmente



▼ **DELETE** - Eliminar equipo por su ID → `/api/equipos/:id`

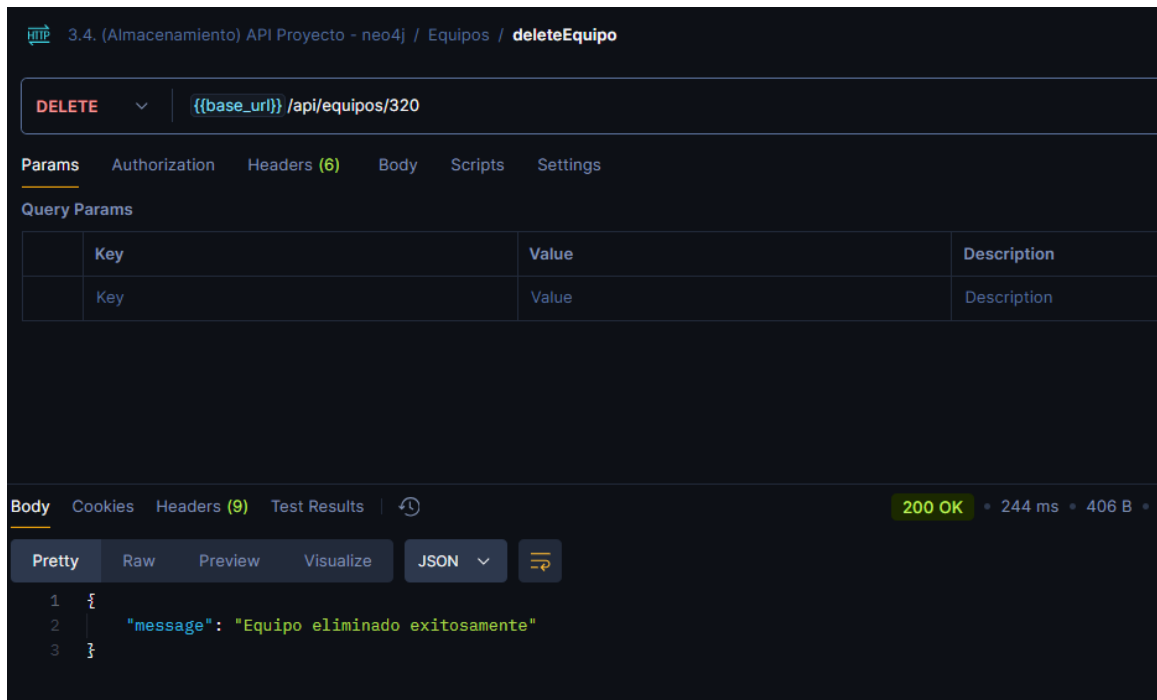


Consideraciones

- No se puede eliminar un equipo si tiene **deportistas** o **contratos** asociados.
- Esto también eliminara las relaciones directas que tenga el *equipo* con otros nodos, mas no eliminará estos.


Parámetros

id → el ID del equipo a eliminar



▼ *Deportistas*

▼ *GET* - Todos los deportistas → `/api/deportistas/`



The screenshot shows a REST client interface with a GET request to `{{base_url}}/api/deportistas/`. The response is a JSON array of two player objects, each containing personal details, birthplace information, and contract details.

```
1  [
2    {
3      "id": 281,
4      "posicion": "PORTERO",
5      "dorsal": 1,
6      "sexo": "MASCULINO",
7      "nombre": "MARC-ANDRE TER STEGEN",
8      "ciudadNacimiento": {
9        "id": 269,
10       "nombre": "MONCHENGLADBACH",
11       "pais": "ALEMANIA"
12     },
13     "contratos": {
14       "total_contratos": 1,
15       "resumen": {
16         "activos": 1,
17         "antiguos": 0,
18         "equipos": [
19           {
20             "nombre": "BARCELONA",
21             "pais": "ESPAÑA",
22             "estado": "Equipo actual (2014-2025)"
23           }
24         ]
25       }
26     }
27   },
28   {
29     "id": 283,
30     "posicion": "DELANTERO",
31     "dorsal": 9,
32     "sexo": "MASCULINO",
33     "nombre": "ROBERT LEWANDOWSKI",
34     "ciudadNacimiento": {
35       "id": 274,
36       "nombre": "VARSOVIA",
37       "pais": "POLONIA"
38     },
39     "contratos": {
40       "total_contratos": 1,
41       "resumen": {
42         "activos": 1,
43         "antiguos": 0,
44         "equipos": [
45           {
46             "nombre": "BARCELONA",
47             "pais": "ESPAÑA",
48             "estado": "Equipo actual (2022-2026)"
49           }
50         ]
51       }
52     }
53   }
54 ]
```

▼ GET - Todos los deportistas de un equipo por su ID →

`/api/deportistas/equipo/:equipoID`

Parámetros

equipoID → el ID del equipo en donde queremos buscar los jugadores
(ejemplo de la imagen, el Barcelona)

HTTP 3.4. (Almacenamiento) API Proyecto - neo4j / Deportistas / getDeportistasEquipo

GET {{base_url}} /api/deportistas/equipo/278

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK

Pretty Raw Preview Visualize JSON

```
1 {
2   "equipoID": 278,
3   "deportistas": [
4     {
5       "posicion": "PORTERO",
6       "dorsal": 1,
7       "sexo": "MASCULINO",
8       "nombre": "MARC-ANDRE TER STEGEN",
9       "id": 281
10    },
11    {
12      "posicion": "DELANTERO",
13      "dorsal": 9,
14      "sexo": "MASCULINO",
15      "nombre": "ROBERT LEWANDOWSKI",
16      "id": 283
17    },
18    {
19      "posicion": "CENTROCAMPISTA OFENSIVO",
20      "dorsal": 8,
21      "sexo": "MASCULINO",
22      "nombre": "PEDRI",
23      "id": 285
24    },
25    {
26      "posicion": "DELANTERO",
27      "dorsal": 10,
28      "sexo": "MASCULINO",
29      "nombre": "RAPHAEL DIAS BELLOLI",
30      "id": 314
31    }
32  ]
33 }
```

▼ GET - Deportista(s) por coincidencia de nombre →

`/api/deportistas/buscar?nombre=<nombre>`

Query

nombre → el nombre (o *la coincidencia*) del deportista que queremos buscar

3.4. (Almacenamiento) API Proyecto - neo4j / Deportistas / **getDeportistasNombre**

GET `{{base_url}} /api/deportistas/buscar?nombre=messi`

Params • Authorization Headers (6) Body Scripts Settings

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description
<input checked="" type="checkbox"/>	nombre	messi	

Body Cookies Headers (9) Test Results **200 OK** • 403 ms • 695 B

Pretty Raw Preview Visualize JSON

```

1  [
2    {
3      "id": 317,
4      "posicion": "DELANTERO",
5      "dorsal": 10,
6      "sexo": "MASCULINO",
7      "nombre": "LIONEL MESSI",
8      "ciudadNacimiento": {
9        "id": 296,
10       "nombre": "ROSARIO",
11       "pais": "ARGENTINA"
12     },
13     "contratos": {
14       "total_contratos": 1,
15       "resumen": {
16         "activos": 1,
17         "antiguos": 0,
18         "equipos": [
19           {
20             "nombre": "INTER MIAMI CF",
21             "pais": "ESTADOS UNIDOS",
22             "estado": "Equipo actual (2023-2025)"
23           }
24         ]
25       }
26     }
27   }

```

▼ **GET - Deportista por su ID** → `/api/deportistas/:id`

Parámetros

id → el ID del deportista que se quiere buscar

3.4. (Almacenamiento) API Proyecto - neo4j / Deportistas / getDeportistasID

GET {{base_url}} /api/deportistas/281

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 281,
3   "posicion": "PORTERO",
4   "dorsal": 1,
5   "sexo": "MASCULINO",
6   "nombre": "MARC-ANDRE TER STEGEN",
7   "ciudadNacimiento": {
8     "id": 269,
9     "nombre": "MONCHENGLADBACH",
10    "pais": "ALEMANIA"
11  },
12  "contratos": {
13    "total_contratos": 1,
14    "resumen": {
15      "activos": 1,
16      "antiguos": 0,
17      "equipos": [
18        {
19          "nombre": "BARCELONA",
20          "pais": "ESPAÑA",
21          "estado": "Equipo actual (2014-2025)"
22        }
23      ]
24    }
25  }
26 }
```

▼ **POST** - Crear deportista → `/api/deportistas/`



Consideraciones

- El formato del body debe ser en JSON, tal cual como se observa en la imagen.
- No hay un estándar para escribir los valores en ninguno de los POSTs, lo importante es que luego se mantengan los mismos valores → *que no sea que de Estados Unidos pasen a escribir USA o EE.UU.*
- Este POST se encarga de **crear un nodo para el país y la ciudad** (*y sus respectivas relaciones*), a pesar de que estas no existan.
 - La relación con *fecha_nacimiento* también se incluye aquí.

Body

nombre → nombre del deportista

dorsal → número que lleva en la camiseta del club (*no va entre comillas*)

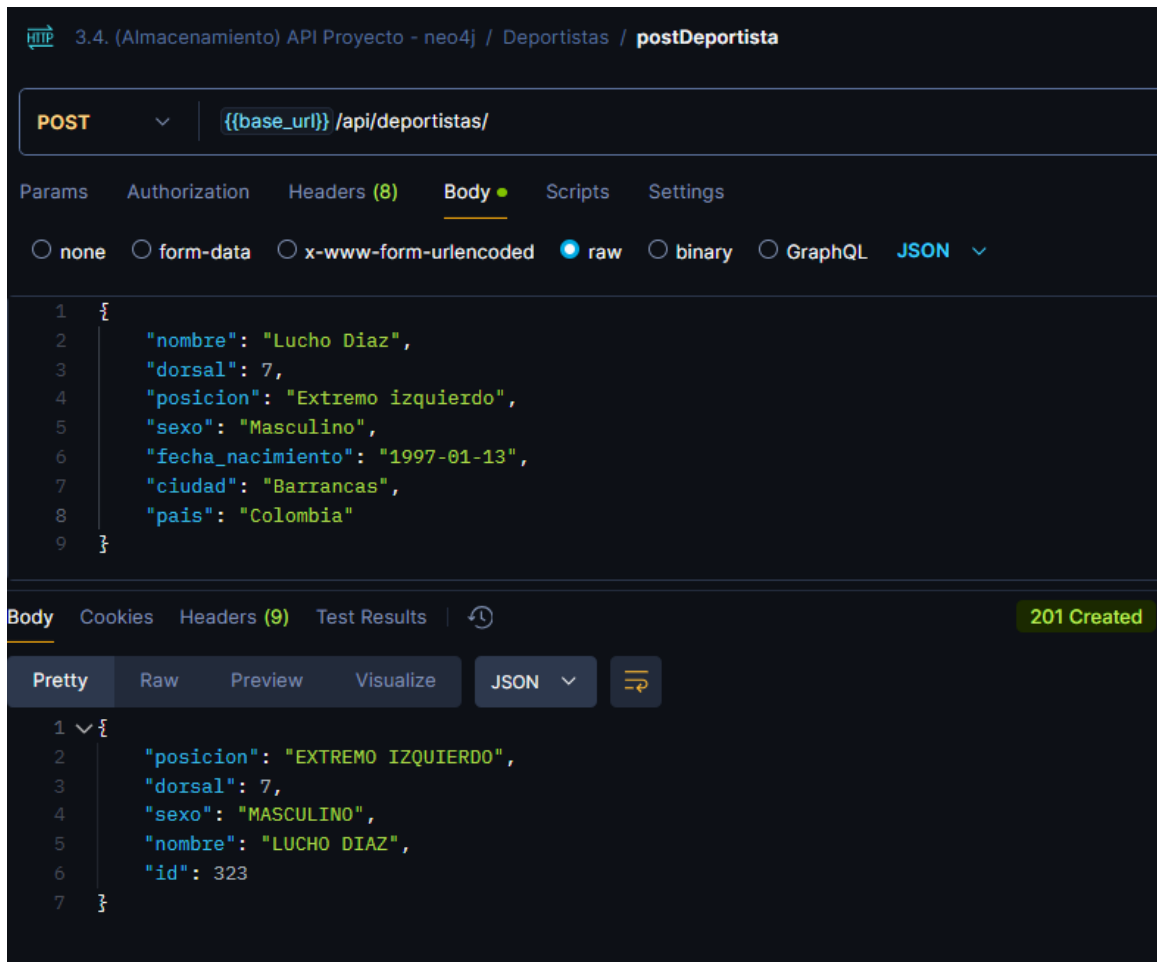
posicion → posición en la que juega el deportista

sexo → sexo del deportista (*Masculino, Femenino, Otro*)

fecha_nacimiento → fecha de nacimiento del deportista (*en formato YYYY-MM-DD*)

ciudad → ciudad de nacimiento del deportista

pais → país de nacimiento del deportista (*vinculado a la **ciudad***)



▼ **PUT** - Modificar deportista por su ID → `/api/deportistas/:id`



Consideraciones

- El formato del *body* debe ser en JSON, tal cual como se observa en la imagen.
- No hay un estándar para escribir los valores en ninguno de los *PUTs*, lo importante es que luego se mantengan los mismos valores → *que no sea que de Estados Unidos pasen a escribir USA o EE.UU.*
- No hay restricciones en la cantidad de campos que se pueden manejar en el *body*: de las 7 claves, se pueden modificar tranquilamente solo 1 o 2 de ellas.
 - Sin embargo, no se recomienda modificar el campo de *país, ciudad o fecha de nacimiento* dado que no hay un manejo para la actualización de los nodos al que "*pais*", "*ciudad*" y "*fecha_nacimiento*" se encuentran vinculados.

Parámetros

id → el ID del deportista a modificar

Body (cada campo es opcional)

nombre → nombre del deportista

dorsal → número que lleva en la camiseta del club (*no va entre comillas*)

posicion → posición en la que juega el deportista

sexo → sexo del deportista (*Masculino, Femenino, Otro*)

fecha_nacimiento (*bajo precaución*) → fecha de nacimiento del deportista (*en formato YYYY-MM-DD*)

ciudad (*bajo precaución*) → ciudad de nacimiento del deportista

pais (*bajo precaución*) → país de nacimiento del deportista (*vinculado a la ciudad*)

3.4. (Almacenamiento) API Proyecto - neo4j / Deportistas / putDeportista

PUT `{{base_url}} /api/deportistas/323`

Params Authorization Headers (8) Body Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL ☒ JSON

```
1 {
2   "nombre": "Lucho Frias",
3   "posicion": "Extremo"
4 }
```

Body Cookies Headers (9) Test Results 200 OK

Pretty Raw Preview Visualize JSON

```
1 {
2   "posicion": "EXTREMO",
3   "dorsal": 7,
4   "sexo": "MASCULINO",
5   "nombre": "LUCHO FRIAS"
6 }
```

▼ **DELETE** - Eliminar deportista por su ID → `/api/deportistas/:id`



Consideraciones

- No se puede eliminar un deportista si tiene **contratos** asociados.
- Esto también eliminara las relaciones directas que tenga el *deportista* con otros nodos, mas no eliminará estos.

Parámetros

id → el ID del deportista a eliminar

HTTP 3.4. (Almacenamiento) API Proyecto - neo4j / Deportistas / **deleteDeportista**

DELETE `{{base_url}}/api/deportistas/323`

Params Authorization Headers (6) Body Scripts Settings

Query Params

	Key	Value	Description
	Key	Value	Description

Body Cookies Headers (9) Test Results 200 OK • 362 ms

Pretty Raw Preview Visualize JSON

```
1 {  
2   "message": "Deportista eliminado exitosamente"  
3 }
```

▼ Contratos

▼ GET - Todos los contratos → `/api/contrataciones/`

3.4. (Almacenamiento) API Proyecto - neo4j / Contrataciones / **getContrataciones**

GET {{base_url}} /api/contrataciones/

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 451 ms • 2.95 KB

Pretty Raw Preview Visualize JSON

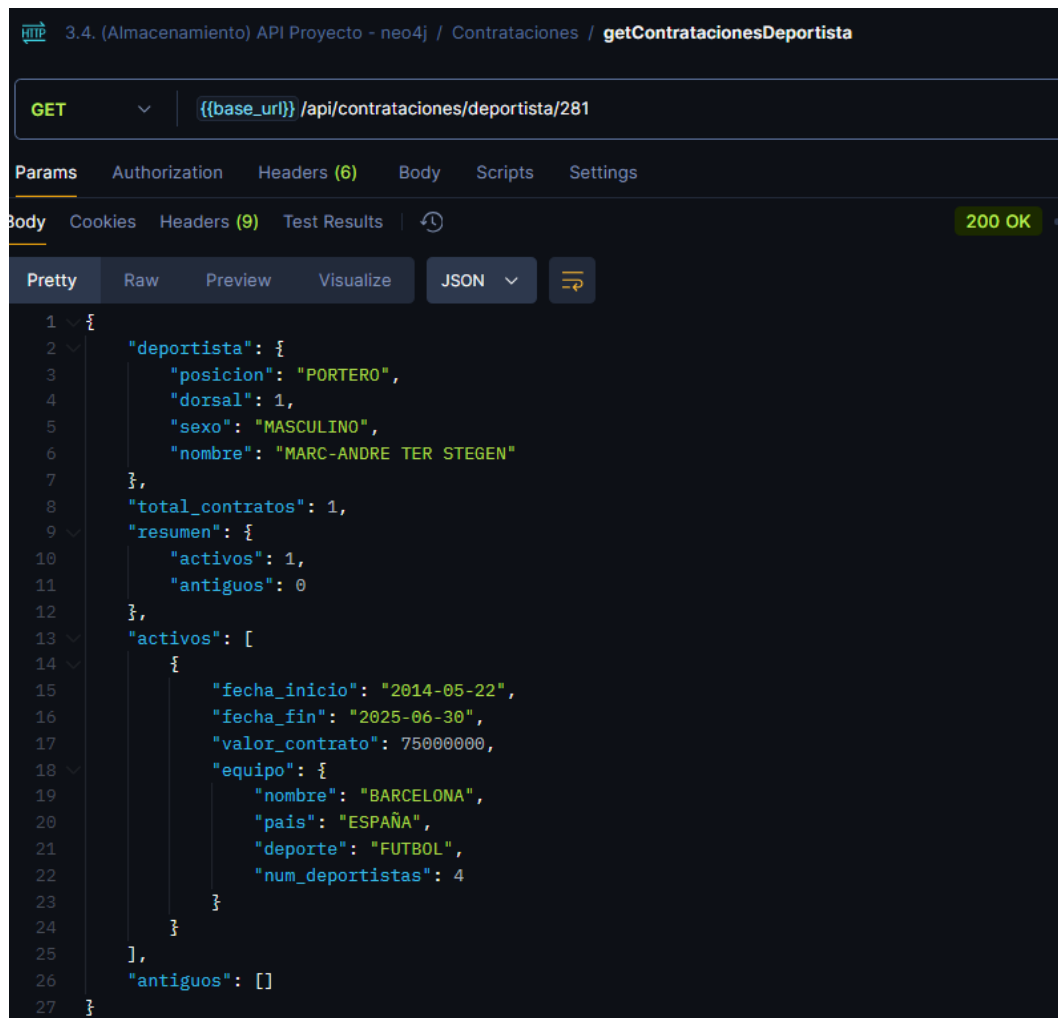
```
1 [
2   {
3     "deportista": {
4       "posicion": "PORTERO",
5       "dorsal": 1,
6       "sexo": "MASCULINO",
7       "nombre": "MARC-ANDRE TER STEGEN"
8     },
9     "equipo": {
10      "nombre": "BARCELONA",
11      "pais": "ESPAÑA",
12      "deporte": "FUTBOL",
13      "num_deportistas": 4
14    },
15    "contrato": {
16      "id": 282,
17      "fecha_inicio": "2014-05-22",
18      "fecha_fin": "2025-06-30",
19      "valor_contrato": 75000000
20    }
21  },
22  {
23    "deportista": {
24      "posicion": "DELANTERO",
25      "dorsal": 9,
26      "sexo": "MASCULINO",
27      "nombre": "ROBERT LEWANDOWSKI"
28    },
29    "equipo": {
30      "nombre": "BARCELONA",
31      "pais": "ESPAÑA",
32      "deporte": "FUTBOL",
```

▼ GET - Contrato(s) de un deportista por su ID →

/api/contrataciones/deportista/:deportistaID

Parámetros

deportistaID → el ID del deportista del que queremos buscar su(s) contrato(s)



3.4. (Almacenamiento) API Proyecto - neo4j / Contrataciones / getContratacionesDeportista

GET {{base_url}} /api/contrataciones/deportista/281

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK

Pretty Raw Preview Visualize JSON

```
1 {
2   "deportista": {
3     "posicion": "PORTERO",
4     "dorsal": 1,
5     "sexo": "MASCULINO",
6     "nombre": "MARC-ANDRE TER STEGEN"
7   },
8   "total_contratos": 1,
9   "resumen": {
10    "activos": 1,
11    "antiguos": 0
12  },
13  "activos": [
14    {
15      "fecha_inicio": "2014-05-22",
16      "fecha_fin": "2025-06-30",
17      "valor_contrato": 75000000,
18      "equipo": {
19        "nombre": "BARCELONA",
20        "pais": "ESPAÑA",
21        "deporte": "FUTBOL",
22        "num_deportistas": 4
23      }
24    }
25  ],
26  "antiguos": []
27 }
```

▼ GET - Contrato(s) de un equipo por su ID →

`/api/contrataciones/equipo/:equipoID`

Parámetros

equipoID → el ID del equipo del que queremos buscar su(s) contrato(s)

3.4. (Almacenamiento) API Proyecto - neo4j / Contrataciones / **getContratacionesEquipo**

GET `{{base_url}} /api/contrataciones/equipo/278`

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 626 ms • 1.45 KB

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "deportista": {
4       "posicion": "PORTERO",
5       "dorsal": 1,
6       "sexo": "MASCULINO",
7       "nombre": "MARC-ANDRE TER STEGEN",
8       "pais": "ALEMANIA"
9     },
10    "contrato": {
11      "fecha_inicio": "2014-05-22",
12      "fecha_fin": "2025-06-30",
13      "valor_contrato": 75000000
14    },
15    "equipo": {
16      "nombre": "BARCELONA",
17      "pais": "ESPAÑA",
18      "deporte": "FUTBOL"
19    }
20  },
21  {
22    "deportista": {
23      "posicion": "DELANTERO",
24      "dorsal": 9,
25      "sexo": "MASCULINO",
26      "nombre": "ROBERT LEWANDOWSKI",
27      "pais": "POLONIA"
28    },
29    "contrato": {
30      "fecha_inicio": "2022-07-19",
31      "fecha_fin": "2026-06-30",
32      "valor_contrato": 50000000
33    }
34  }
35 ]
```

▼ **POST** - Crear contrato → `/api/contrataciones/`



Consideraciones

- El formato del body debe ser en JSON, tal cual como se observa en la imagen.
- No hay un estándar para escribir los valores en ninguno de los POSTs, lo importante es que luego se mantengan los mismos valores → *que no sea que de Estados Unidos pasen a escribir USA o EE.UU.*
- Este POST se encarga de crear las relaciones entre *Deportista* y *Contrato* (**TIENE_CONTRATO**), *Equipo* y *Contrato* (**CONTRATO_CON**), y *Deportista* y *Equipo* (**JUEGA_EN** , *que presenta carencias debido a que no se crea junto a **fecha_vinculación***)

Body

deportistaID → el ID del deportista que va a ser contratado

equipoID → el ID del equipo que contrata al deportista

fecha_inicio → la fecha que fue contratado el deportista (*en formato YYYY-MM-DD*)

fecha_fin → la fecha en que el primer contrato termina (*en formato YYYY-MM-DD*)

valor_contrato → por cuánto fue contratado el deportista

HTTP 3.4. (Almacenamiento) API Proyecto - neo4j / Contrataciones / **postContratacion**

POST `{{base_url}}/api/contrataciones/`

Params Authorization Headers (8) **Body** Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** ▾

```
1 {
2   "deportistaID": 323,
3   "equipoID": 320,
4   "fecha_inicio": "2023-07-15",
5   "fecha_fin": "2025-12-31",
6   "valor_contrato": 25000000
7 }
```

Body Cookies Headers (9) Test Results **201 Created** • 699 ms

Pretty Raw Preview Visualize **JSON** ▾

```
1 {
2   "id": 324,
3   "deportista": {
4     "posicion": "EXTREMO",
5     "dorsal": 7,
6     "sexo": "MASCULINO",
7     "nombre": "LUCHO FRIAS"
8   },
9   "equipo": {
10    "nombre": "LIVERPOOL FC"
11  },
12  "contrato": {
13    "id": 324,
14    "fecha_inicio": "2023-07-15",
15    "fecha_fin": "2025-12-31",
16    "valor_contrato": 25000000
17  }
18 }
```

▼ **PUT** - Modificar contrato por su ID → `/api/contrataciones/:id`



Consideraciones

- El formato del *body* debe ser en JSON, tal cual como se observa en la imagen.
- No hay un estándar para escribir los valores en ninguno de los *PUTs*, lo importante es que luego se mantengan los mismos valores → *que no sea que de Estados Unidos pasen a escribir USA o EE.UU.*
- No hay restricciones en la cantidad de campos que se pueden manejar en el *body*: de las 3 claves, se pueden modificar tranquilamente solo 1 o 2 de ellas.

Parámetros

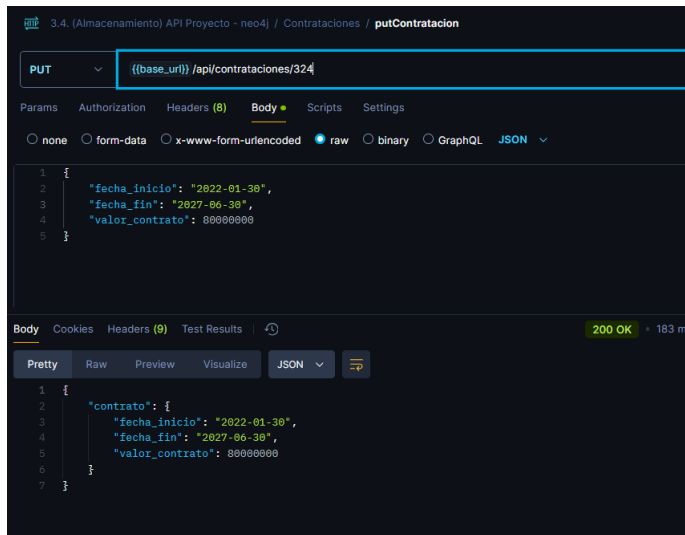
id → el ID del contrato que queremos modificar

Body (cada campo es opcional)

fecha_inicio → la fecha que fue contratado el deportista (*en formato YYYY-MM-DD*)

fecha_fin → la fecha en que el primer contrato termina (*en formato YYYY-MM-DD*)

valor_contrato → por cuánto fue contratado el deportista



```
3.4. (Almacenamiento) API Proyecto - neo4j / Contrataciones / putContratacion

PUT {{base_url}}/api/contrataciones/324

Params Authorization Headers (8) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 {
2   "fecha_inicio": "2022-01-30",
3   "fecha_fin": "2027-06-30",
4   "valor_contrato": 80000000
5 }

Body Cookies Headers (9) Test Results 200 OK 183 ms

Pretty Raw Preview Visualize JSON

1 {
2   "contrato": {
3     "fecha_inicio": "2022-01-30",
4     "fecha_fin": "2027-06-30",
5     "valor_contrato": 80000000
6   }
7 }
```



```
"deportista": {
  "posicion": "EXTREMO",
  "dorsal": 7,
  "sexo": "MASCULINO",
  "nombre": "LUCHO FRIAS"
},
"total_contratos": 1,
"resumen": {
  "activos": 1,
  "antiguos": 0
},
"activos": [
  {
    "fecha_inicio": "2022-01-30",
    "fecha_fin": "2027-06-30",
    "valor_contrato": 80000000,
    "equipo": {
      "nombre": "LIVERPOOL FC",
      "pais": "INGLATERRA",
      "deporte": "FUTBOL",
      "num_deportistas": 1
    }
  }
],
"antiguos": []
}
```

▼ DELETE - Eliminar contrato por sus ID →

`/api/contrataciones/:id`

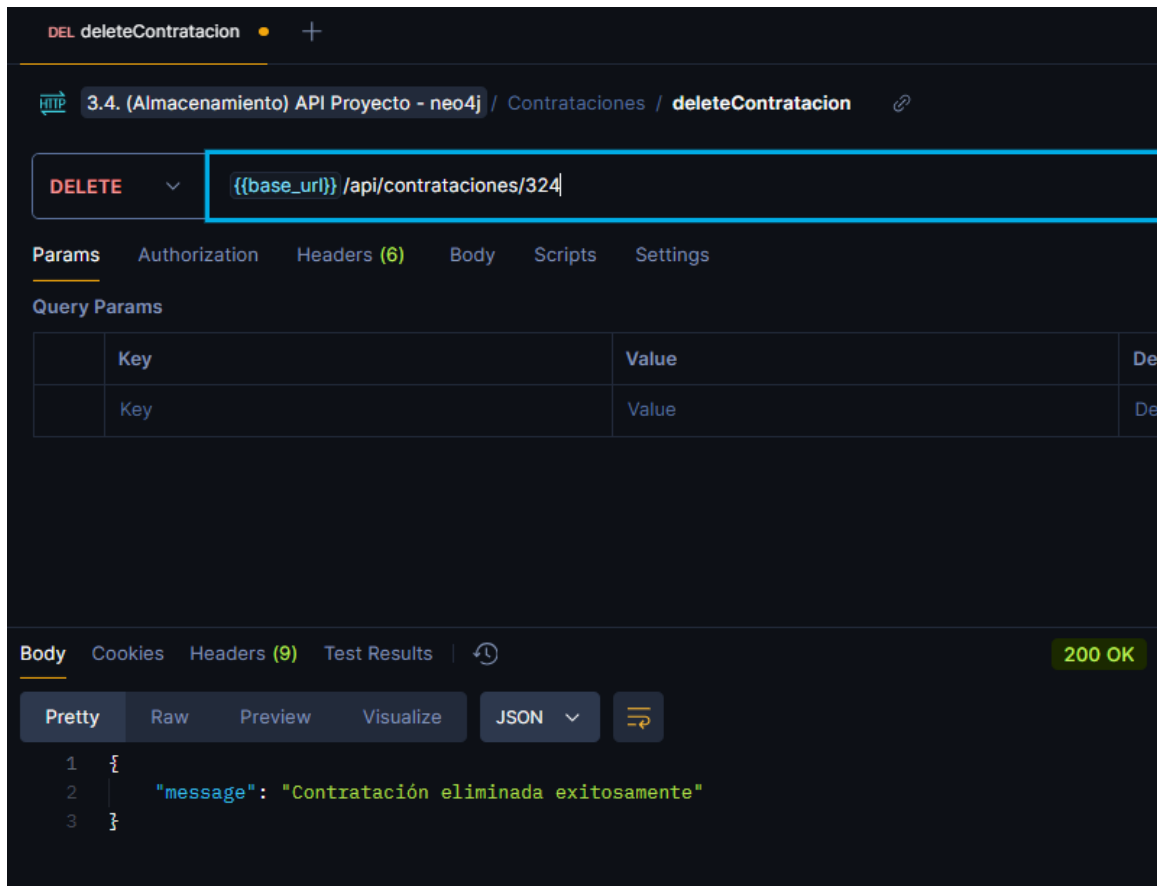


Consideraciones

- Esto también eliminara las relaciones directas que tenga el *contrato* con otros nodos, mas no eliminará estos.

Parámetros

id → el ID del contrato que va a ser eliminado



▼ Consultas del profe

▼ Deportistas con contratos a partir de una fecha específica

→ `/api/consultas/contratos-desde?fecha=<fecha>`

Query

fecha → la fecha desde cuando se quieren ver los contratos (*en formato YYYY-MM-DD*)

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Profe / **Deportistas con contratos a partir de una fecha específica**

GET `{{base_url}} /api/consultas/contratos-desde?fecha=2020-01-01`

Params • Authorization Headers (6) Body Scripts Settings

Query Params

<input checked="" type="checkbox"/> Key	Value	Description
<input checked="" type="checkbox"/> fecha	2020-01-01	
Key	Value	Description

Body Cookies Headers (9) Test Results 200 OK • 2.34 s

Pretty Raw Preview Visualize JSON ↕

```

1  [
2    {
3      "deportista": {
4        "posicion": "CENTROCAMPISTA OFENSIVO",
5        "dorsal": 8,
6        "sexo": "MASCULINO",
7        "nombre": "PEDRI"
8      },
9      "contrato": {
10       "fecha_inicio": "2020-07-01",
11       "fecha_fin": "2025-06-30",
12       "valor_contrato": 20000000
13     },
14     "equipo": {
15       "nombre": "BARCELONA",
16       "pais": "ESPAÑA",
17       "deporte": "FUTBOL"
18     }
19   },
20   {
21     "deportista": {
22       "posicion": "CENTROCAMPISTA DEFENSIVO",
23       "dorsal": 4

```

▼ **Deportistas masculinos en equipos de fútbol de España →**

`/api/consultas/masculinos-futbol-espana`

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Profe / **Deportistas masculinos en equipos de fútbol de España**

GET {{base_url}} /api/consultas/masculinos-futbol-espana

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 163 ms • 967 B

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "deportista": {
4       "posicion": "PORTERO",
5       "dorsal": 1,
6       "sexo": "MASCULINO",
7       "nombre": "MARC-ANDRE TER STEGEN"
8     },
9     "equipo": {
10      "nombre": "BARCELONA",
11      "pais": "ESPAÑA"
12    }
13  },
14  {
15    "deportista": {
16      "posicion": "DELANTERO",
17      "dorsal": 9,
18      "sexo": "MASCULINO",
19      "nombre": "ROBERT LEWANDOWSKI"
20    },
21    "equipo": {
22      "nombre": "BARCELONA",
23      "pais": "ESPAÑA"
24    }
25  },
26 ]
```

▼ **Deportistas españoles en equipos españoles →**

/api/consultas/deportistas-espana-equipos-espana

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Profe / Deportistas españoles en equipos españoles

GET {{base_url}} /api/consultas/deportistas-espana-equipos-espana

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 256 ms

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "deportista": {
4       "posicion": "CENTROCAMPISTA OFENSIVO",
5       "dorsal": 8,
6       "sexo": "MASCULINO",
7       "nombre": "PEDRI",
8       "ciudad_nacimiento": {
9         "nombre": "SABADELL",
10        "pais": "ESPAÑA"
11      }
12    },
13    "equipo": {
14      "nombre": "BARCELONA",
15      "pais": "ESPAÑA"
16    }
17  }
18 ]
```

▼ Deportistas con contratos superiores a 1,000,000 →

/api/consultas/contratos-altos

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Profe / **Deportistas con contratos superiores a 1,000,000**

GET `{{base_url}}/api/consultas/contratos-altos`

Params Authorization Headers (6) Body Scripts Settings

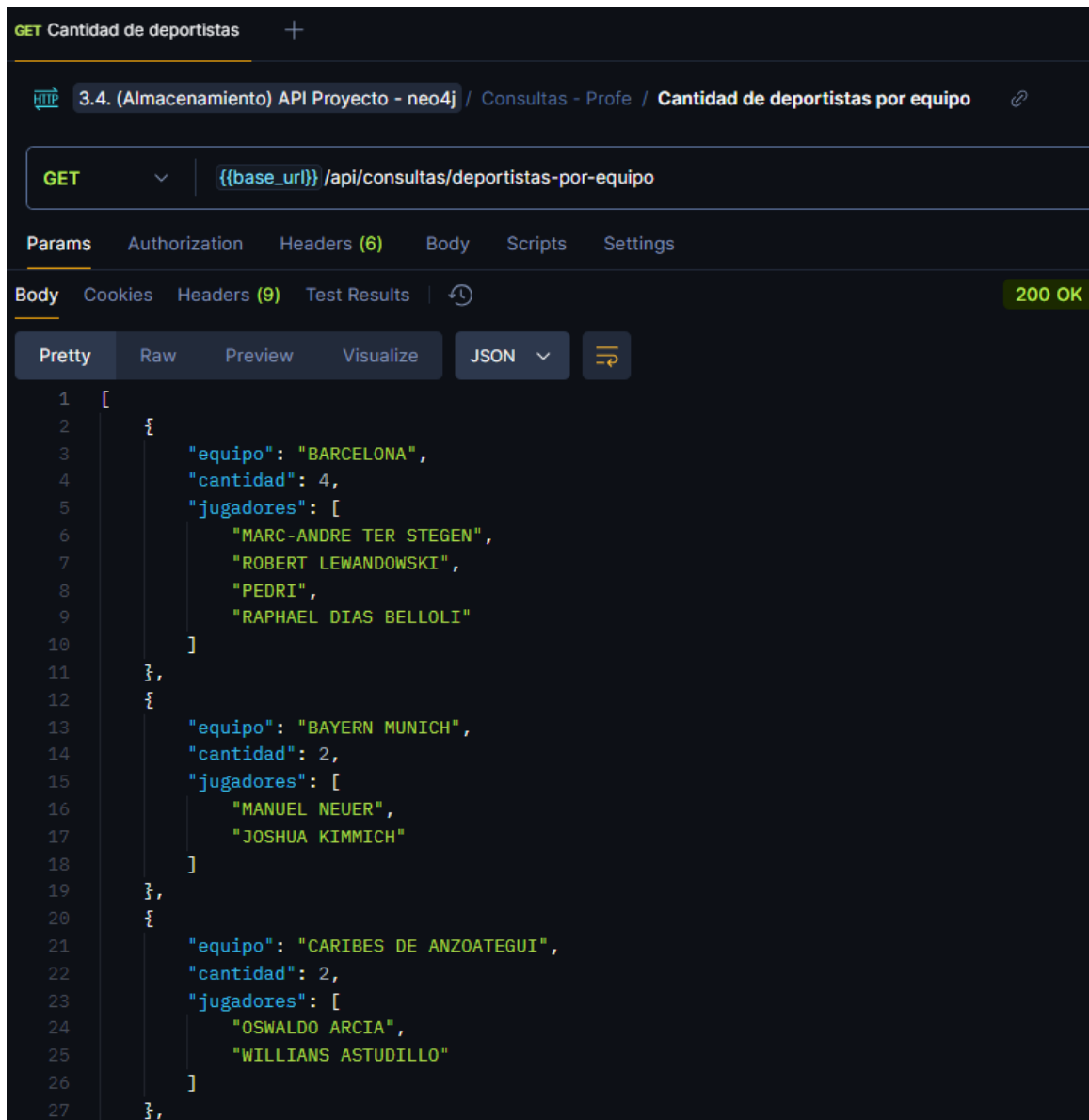
Body Cookies Headers (9) Test Results 200 OK • 228

Pretty Raw Preview Visualize JSON

```
1  [
2    {
3      "deportista": {
4        "posicion": "PORTERO",
5        "dorsal": 1,
6        "sexo": "MASCULINO",
7        "nombre": "MARC-ANDRE TER STEGEN"
8      },
9      "contrato": {
10       "fecha_inicio": "2014-05-22",
11       "fecha_fin": "2025-06-30",
12       "valor_contrato": 75000000
13     },
14     "equipo": {
15       "nombre": "BARCELONA",
16       "pais": "ESPAÑA",
17       "deporte": "FUTBOL"
18     }
19   },
20   {
21     "deportista": {
22       "posicion": "DELANTERO",
23       "dorsal": 9,
24       "sexo": "MASCULINO",
25       "nombre": "ROBERT LEWANDOWSKI"
26     },
27     "contrato": {
28       "fecha_inicio": "2022-07-19",
29       "fecha_fin": "2026-06-30",
30       "valor_contrato": 50000000
31     },
32   }
```

▼ Cantidad de deportistas por equipo →

`/api/consultas/deportistas-por-equipo`



GET Cantidad de deportistas

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Profe / Cantidad de deportistas por equipo

GET {{base_url}} /api/consultas/deportistas-por-equipo

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK

Pretty Raw Preview Visualize JSON

```
[
  {
    "equipo": "BARCELONA",
    "cantidad": 4,
    "jugadores": [
      "MARC-ANDRE TER STEGEN",
      "ROBERT LEWANDOWSKI",
      "PEDRI",
      "RAPHAEL DIAS BELLOLI"
    ]
  },
  {
    "equipo": "BAYERN MUNICH",
    "cantidad": 2,
    "jugadores": [
      "MANUEL NEUER",
      "JOSHUA KIMMICH"
    ]
  },
  {
    "equipo": "CARIBES DE ANZOATEGUI",
    "cantidad": 2,
    "jugadores": [
      "OSWALDO ARCIA",
      "WILLIAMS ASTUDILLO"
    ]
  }
]
```

▼ Consultas del grupo

▼ Nacionalidad de los deportistas en determinado equipo →

`/api/consultas/nacionalidad-deportistas-por-equipo`

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Grupo / Nacionalidad de los deportistas en determinado equipo

GET {{base_url}} /api/consultas/nacionalidad-deportistas-por-equipo

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 731 ms • 1.4 KB

Pretty Raw Preview Visualize JSON ↻

```
1 [
2   {
3     "nacionalidad": "ALEMANIA",
4     "cantidad": 3,
5     "jugadores": {
6       "MARC-ANDRE TER STEGEN": {
7         "equipo": "BARCELONA",
8         "deporte": "FUTBOL",
9         "pais": "ESPAÑA"
10      },
11      "MANUEL NEUER": {
12        "equipo": "BAYERN MUNICH",
13        "deporte": "FUTBOL",
14        "pais": "ALEMANIA"
15      },
16      "JOSHUA KIMMICH": {
17        "equipo": "BAYERN MUNICH",
18        "deporte": "FUTBOL",
19        "pais": "ALEMANIA"
20      }
21    }
22  },
23  {
24    "nacionalidad": "ARGENTINA",
25    "cantidad": 1,
26    "jugadores": {
27      "LIONEL MESSI": {
28        "equipo": "INTER MIAMI CF",
29        "deporte": "FUTBOL",
30        "pais": "ESTADOS UNIDOS"
31      }
32    }
33  }
34 ]
```

▼ Deportistas por cada deporte → `/api/consultas/deportistas-por-deporte-minimo-1`

HTTP 3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Grupo / **Deportistas por cada deporte**

GET `{{base_url}} /api/consultas/deportistas-por-deporte-minimo-1`

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 402 ms • 433 B •

Pretty Raw Preview Visualize JSON ↗

```
1 [
2   {
3     "deporte": "FUTBOL",
4     "cantidad": 7
5   },
6   {
7     "deporte": "BEISBOL",
8     "cantidad": 2
9   }
10 ]
```

▼ Contratos que terminan en los próximos 6 meses →

`/api/consultas/contratos-terminan-en-6-meses`

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Grupo / Contratos que terminan en los próximos 6 meses

GET {{base_url}} /api/consultas/contratos-terminan-en-6-meses

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 451 ms • 967 B • Save

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "deportista": {
4       "posicion": "JARDINERO",
5       "dorsal": 9,
6       "sexo": "MASCULINO",
7       "nombre": "OSWALDO ARCIA"
8     },
9     "contrato": {
10      "fecha_inicio": "2010-10-12",
11      "fecha_fin": "2024-12-31",
12      "valor_contrato": 15000000,
13      "leFaltan": "1 mes, 10 días"
14    },
15    "equipo": {
16      "nombre": "CARIBES DE ANZOATEGUI",
17      "pais": "VENEZUELA",
18      "deporte": "BEISBOL"
19    }
20  },
21  {
22    "deportista": {
23      "posicion": "RECEPTOR",
24      "dorsal": 37,
25      "sexo": "MASCULINO",
26      "nombre": "WILLIAMS ASTUDILLO"
27    },
28    "contrato": {
29      "fecha_inicio": "2013-10-15",
30      "fecha_fin": "2024-12-31",
31      "valor_contrato": 12000000,
32      "leFaltan": "1 mes, 10 días"
33    }
34  }
35 ]
```

▼ Equipos con al menos un contrato activo →

`/api/consultas/equipos-contratos-activos`

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Grupo / Equipos con al menos un contrato activo

GET {{base_url}} /api/consultas/equipos-contratos-activos

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 382 ms • 1.22 KB

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "equipo": {
4       "nombre": "BARCELONA"
5     },
6     "contratosActivos": 4,
7     "jugadores": [
8       {
9         "nombre": "MARC-ANDRE TER STEGEN",
10        "pais": "ALEMANIA",
11        "posicion": "PORTERO"
12      },
13      {
14        "nombre": "ROBERT LEWANDOWSKI",
15        "pais": "POLONIA",
16        "posicion": "DELANTERO"
17      },
18      {
19        "nombre": "PEDRI",
20        "pais": "ESPAÑA",
21        "posicion": "CENTROCAMPISTA OFENSIVO"
22      },
23      {
24        "nombre": "RAPHAEL DIAS BELLOLI",
25        "pais": "BRASIL",
26        "posicion": "DELANTERO"
27      }
28    ]
29  },
30 ]
```

▼ Deportistas con contratos de duración superior a 3 años →
`/api/consultas/deportistas-contratos-largos`

GET Deportistas con conti x +

3.4. (Almacenamiento) API Proyecto - neo4j / Consultas - Grupo / Deportistas con contratos de duración superior a 3 años

GET {{base_url}} /api/consultas/deportistas-contratos-largos

Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (9) Test Results 200 OK • 247 ms • 2.57 KB

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "deportista": {
4       "posicion": "PORTERO",
5       "dorsal": 1,
6       "sexo": "MASCULINO",
7       "nombre": "MARC-ANDRE TER STEGEN"
8     },
9     "contrato": {
10      "fecha_inicio": "2014-05-22",
11      "fecha_fin": "2025-06-30",
12      "valor_contrato": 75000000,
13      "leFaltan": "0 años, 7 meses, 9 días"
14    },
15    "equipo": {
16      "nombre": "BARCELONA",
17      "pais": "ESPAÑA"
18    }
19  },
20  {
21    "deportista": {
22      "posicion": "DELANTERO",
23      "dorsal": 9,
24      "sexo": "MASCULINO",
25      "nombre": "ROBERT LEWANDOWSKI"
26    },
27    "contrato": {
28      "fecha_inicio": "2022-07-19",
29      "fecha_fin": "2026-06-30",
30      "valor_contrato": 50000000,
31      "leFaltan": "1 año, 7 meses, 9 días"
32    }
33  }
34 ]
```