

ALEJANDRO SAINZ SAINZ

PARTE 1

EQUIPOS

```
// Notacion
@Entity
@Table(name="equipos")
public class Equipo {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private Long id;

    // Por comodidad uso los mismos nombres que en sql, en snake_case, que
    // es de java, pero asi me aseguro de que los nombres coincidan con los
    // tablas
    @Column(nullable = false)
    private String nombre_equipo;

    @Column(nullable=false)
    private String sede;

    // Genero constructor vacio y getters y setters
    public Equipo() {
    }

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }
}
```

```

// Endpoint principal
@RequestMapping("/equipos")
public class EquiposController {

    @Autowired
    private PrimaryService servicio;

    // Hago primero el getMapping para todos los equipos
    @GetMapping("/todos")
    public List<Equipo> listarEquipos(){
        return servicio.obtenerEquipos();
    }

    // Metodo Post para añadir un equipo
    @PostMapping
    public void addEquipo(@RequestBody Equipo nuevoEquipo){
        servicio.addEquipo(nuevoEquipo);
    }

    // Metodo para poder modificar el nombre de un equipo
    // Quizá podía pasar el nombre desde el cuerpo de la petición pero no me
    // arriesgo, prefiero con @RequestParam. Creo que más fácil asegurar
    @PutMapping("/actualizarnombre/{uuid}")
    public void modificarRegistro( @PathVariable Long id, @RequestParam String
    nombre){
        servicio.actualizarNombre(id, nombre);
    }

    // Creo el método de eliminar un equipo
    @DeleteMapping("/eliminarEquipo/{id}")
    public void eliminarEquipo(@PathVariable Long id){
        servicio.eliminarEquipo(id);
    }
}

```

El controller de equipos, no se si lo haré todo en uno o haré varios.

```

package com.api.futbol.repository;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.api.futbol.entity.Equipo;

// Lo primero la notación
@Repository
// A diferencia de otras veces como el UUID es de tipo String aquí tiene que
// ser de tipo String. Por ahora, al principio no me preocupo de las relaciones.
// Esta va a ser mi primera prueba
public interface EquiposInterface extends JpaRepository<Equipo, Long> {
}

```

Por ahora no he añadido nada del manytoone, va solo esto hasta el momento

```

// Me aseguro de la notación
@Service
public class PrimaryService {

    @Autowired
    private EquiposInterface equipInter;

    // Método que llama al repositorio para obtener todos los equipos
    public List<Equipo> obtenerEquipos() {
        return equipInter.findAll();
    }

    // Método para añadir
    public void addEquipo(Equipo nuevoEquipo){
        equipInter.save(nuevoEquipo);
    }

    // Método para actualizar
    public void actualizarNombre(Long id, String nombre){
        // Declaro una instancia de equipo y retorno su valor mediante findById
        Equipo equipo = equipInter.findById(id).orElse(null);
        // Modifico el nombre del equipo mediante el setter
        equipo.setNombre_equipo(nombre);

        equipInter.save(equipo);
    }

    // Método del service para eliminar.
    public void eliminarEquipo(Long id) {
        equipInter.deleteById(id);
    }
}

```

El servicio que tengo creado por ahora.

The screenshot shows the Postman interface with the 'equipos' collection selected. On the left, there's a sidebar with 'New HTTP Request' and 'Globals'. On the right, the 'New Environment' tab is open, showing a table with two variables:

Variable	Type	Initial value	Current value
base	default	http://localhost:8080	http://localhost:8080
endpoint	default	/equipos	/equipos

Primer Enviroment en Postman para equipos.

GET | [base] | [endpoint] /todos

Params Authorization Headers (7) Body Scripts Settings

Query Params

Key	Value
Key	Value

Body Cookies Headers (8) Test Results

Pretty Raw Preview JSON ↻

Status: 200 OK Time: 273 ms Size: 255 B

1

Primer Get En vacio devuelve array y 200 ok

POST | [base] | [endpoint] /add

Params Authorization Headers (9) Body Scripts Settings

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

```
[{"id": 1, "nombre_equipo": "los suicidados", "sede": "alguna cuneta por ahí"}]
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Text ↻

Status: 200 OK Time: 65 ms Size: 212 B

1

Tuve que hacer algún cambio, ahora tiene el post endpoint propio. Pero funciona al final.

GET | [base] | [endpoint] /todos

Params Authorization Headers (7) Body Scripts Settings

Query Params

Key	Value
Key	Value

Body Cookies Headers (8) Test Results

Pretty Raw Preview JSON ↻

```
[
  {
    "id": 1,
    "nombre_equipo": "los Suicidados",
    "sede": "alguna cuneta por ahí"
  },
  {
    "id": 2,
    "nombre_equipo": "Las Alimañas",
    "sede": "El Cubil"
  },
  {
    "id": 3,
    "nombre_equipo": "Los Perezosos",
    "sede": "El barranco"
  }
]
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE

e1_0.id,
e1_0.nombre_equipo,
e1_0.sede
from
equipos e1_0

a: Ready

Meto algún equipo más. Hago un getAll para ver cambios. Voy a cambiar ahora el nombre de Los perezosos.

The screenshot shows the Postman interface with a PUT request to the endpoint `({base})/{endpoint}/actualizanombre/{?nombre=Los Ladrones}`. The 'Params' tab is selected, showing a query parameter `nombre` with the value `Los Ladrones`. The 'Test Results' section shows a status of 200 OK with a response time of 76 ms and a size of 212 B. The response body is empty, indicated by the number '1'.

Parece que va bien

The screenshot shows the Postman interface with a GET request to the endpoint `({base})/{endpoint}/todos`. The 'Params' tab is selected, showing no parameters. The 'Test Results' section shows a status of 200 OK with a response time of 17 ms and a size of 448 B. The response body is a JSON array containing two objects representing teams:

```
9   "nombre_equipo": "Las Alimañas",
10  "sede": "El Cubil"
11 },
12 {
13   "id": 3,
14   "nombre_equipo": "Los Ladrones",
15   "sede": "El barranco"
16 }
17 ]
```

Vemos el cambio.

The screenshot shows the Postman interface for a DELETE request to the endpoint {{base}}/{{endpoint}}/eliminarEquipo/3. The response status is 200 OK, time 31 ms, size 212 B. The body of the response is empty, containing only the number 1.

```
DELETE {{base}}{{endpoint}}/eliminarEquipo/3
```

Status: 200 OK Time: 31 ms Size: 212 B

1

Eliminamos el equipo con id 3.

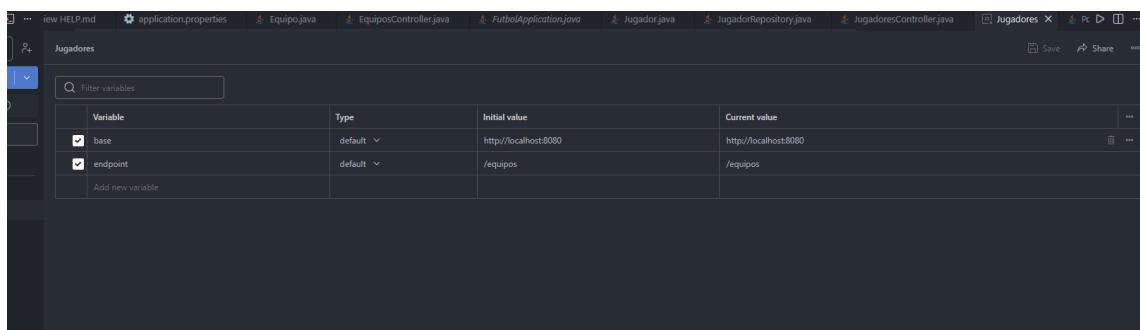
The screenshot shows the Postman interface for a GET request to the endpoint {{base}}/{{endpoint}}/todos. The response status is 200 OK, time 14 ms, size 387 B. The body of the response is a JSON array containing two team objects.

```
GET {{base}}{{endpoint}}/todos
```

```
[{"id": 1, "nombre_equipo": "los Suicidados", "sede": "alguna cuneta por ah\u00ed"}, {"id": 2, "nombre_equipo": "Las Altima\u00f1as", "sede": "El Cubil"}]
```

Status: 200 OK Time: 14 ms Size: 387 B

Listo



Enviroment de jugadores

A screenshot of the Postman API client. The URL is set to 'http://localhost:8080/jugadores/todos'. The 'Params' tab is selected, showing a single query parameter 'Key' with a value of 'Value'. The 'Body' tab shows JSON response data: '1 []'. The status bar at the bottom right indicates 'Status: 200 OK Time: 285 ms Size: 255 B'. Other tabs include 'Cookies', 'Headers (8)', 'Test Results', and 'Pretty'.

Gracias por la ayuda, el primer get all. A ver que fallos da a partir de ahora.

ARBITROS

Postman screenshot showing a GET request to `/Arbitros/GetAll`. The response status is 200 OK, Time: 237 ms, Size: 166 B. The response body is empty, indicated by a single character '1'.

El get básico va. Por ahora no devuelve nada pero va.

Postman screenshot showing a POST request to `/Arbitros/AddArbitros`. The response status is 200 OK, Time: 114 ms, Size: 123 B. The response body contains a JSON array with one element:

```
[{"id": "23577b31-ac43-477a-8dc2-2fdacf6d5fb8", "nombre": "Jose Luis", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "PRINCIPAL"}, {"id": "86e01735-8898-4210-ad69-1c2da47b3ef", "nombre": "Pepe", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "PRINCIPAL"}, {"id": "5770804c-2df6-4e8e-b3cb-8244628f9481", "nombre": "Antonio", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "PRINCIPAL"}, {"id": "b1f8ff6e-02d7-4886-abec-87bd617c76f", "nombre": "Fernando", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "PRINCIPAL"}, {"id": "c9c46c6-c799-4f2e-a74a-94e50f495c88", "nombre": "Antonio", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "ASISTENTE"}, {"id": "ce8d6f0a-7ba3-4ead-a86c-88ed50e15bb3", "nombre": "Fernando", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "ASISTENTE"}, {"id": "e33da22c-d9ae-4f5c-bd78-687bcd726d72", "nombre": "Paco", "apellido1": "Pereira", "apellido2": "Antunez", "rol": "ASISTENTE"}]
```

Insertamos un árbitro.

Postman screenshot showing a POST request to `/Arbitros/AddArbitros`. The response status is 200 OK, Time: 88 ms, Size: 1.03 KB. The response body is empty, indicated by a single character '1'.

Terminal window showing Hibernate SQL insert statements:

```
Hibernate: insert into arbitros (apellido1, apellido2, nombre, rol, id) values (?, ?, ?, ?, ?)
Hibernate: select
```

Meto unos cuantos para poder ahora probar el filtro.

The screenshot shows the Postman interface with a dark theme. In the left sidebar, there's a tree view with categories like 'Peticiones Arbitros', 'Peticiones Equipos', and 'Peticiones Jugadores'. The main area shows a 'New HTTP Request' panel with a 'GET' method selected. The URL is `[[base]]/[[endpoint]]/futbol/PeticionesArbitros/GetByFiltro`. Under 'Query Params', there is a table with one row: 'Key' is 'rol' and 'Value' is 'ASISTENTE'. Below the table, there are tabs for 'Body', 'Cookies', 'Headers (5)', and 'Test Results'. The 'Test Results' tab is active, showing a status of 'Status: 200 OK Time: 27 ms Size: 544 B'. The JSON response body is displayed in a code block:

```
7   },
8   {
9     "rol": "ASISTENTE"
10    },
11    {
12      "id": "ce0dd60a-7ba1-4ead-a86c-88be050e350b",
13      "nombre": "Fernando",
14      "apellidos": "Ferreira",
15      "apellidos2": "Antunes",
16      "rol": "ASISTENTE"
17    },
18    {
19      "id": "e33da22c-d9aa-4f5c-bd78-687bcfd726d72",
20      "rol": "Referee"
```

Buscar rol por asistente. Añadimos un param en postman y parece que esto lo he hecho bien.

The screenshot shows the Postman interface with a dark theme. In the left sidebar, there's a tree view with categories like 'Peticiones Arbitros', 'Peticiones Equipos', and 'Peticiones Jugadores'. The main area shows a 'New HTTP Request' panel with a 'DELETE' method selected. The URL is `[[base]]/[[endpoint]]/borrar/27577b31-ac43-477a-8dc2-9fcacfd5fb8`. Under 'Query Params', there is a table with one row: 'Key' is 'Key' and 'Value' is 'Value'. Below the table, there are tabs for 'Body', 'Cookies', 'Headers (4)', and 'Test Results'. The 'Test Results' tab is active, showing a status of 'Status: 200 OK Time: 26 ms Size: 123 B'. The response body is a single digit '1'.

Si lo mando así parece que va. Tendré que comprobar si le veo en la lista.

```
GET | {{base}}/{{endpoint}}/todos
Params Authorization Headers (7) Body Scripts Settings
Body Cookies Headers (5) Test Results
Pretty Raw Preview JSON ▾
1 [
2   {
3     "id": "86e01735-8898-4210-ad69-1c2da47b3ee",
4     "nombre": "Pepe",
5     "apellido1": "Pereira",
6     "apellido2": "Antunez",
7     "rol": "PRINCIPAL"
8   },
9   {
10    "id": "8770b94c-2df6-4e06-b3cb-8244628fb481",
11    "nombre": "Antonio",
12    "apellido1": "Pereira",
13    "apellido2": "Antunez",
14    "rol": "PRINCIPAL"
15  },
16  {
17    "id": "b1f0f6e0-02d7-4086-abec-87bdf17cc76f",
18    "nombre": "Fernando",
19    "apellido1": "Pereira",
20    "apellido2": "Antunez",
21    "rol": "PRINCIPAL"
22  },
23  {
24    "id": "ca4c46c6-e799-4f2e-a24a-94e5df405c80",
25    "nombre": "Antonio",
26    "apellido1": "Pereira",
27    "apellido2": "Antunez",
28    "rol": "ASISTENTE"
29  },
30  {
31    "id": "e8e6f69a-7ba3-4ead-a86c-88ed50e15bb3",
32    "nombre": "Fernando",
33    "apellido1": "Pereira",
34  }
]

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE

arbitros al_0

Había cogido el primero, y ahora no está. En principio supongo que bien.