

## Ejercicio 13 – Relaciones OneToOne y OneToMany

En esta etapa del aprendizaje en el desarrollo de aplicaciones Java con Spring Boot, nos adentraremos en el fascinante mundo de las relaciones entre entidades utilizando **Spring Data JPA**. Hasta ahora, hemos trabajado con operaciones básicas sobre una única tabla, pero en esta práctica, daremos un paso más allá al explorar las relaciones OneToOne y OneToMany.

### OneToOne

La relación **OneToOne** establece una conexión directa entre dos entidades, indicando que un objeto de una entidad está relacionado exactamente con un objeto de la otra entidad, y viceversa. Esta relación suele ser útil cuando queremos vincular dos entidades de manera exclusiva.

Imagina, por ejemplo, una entidad Persona y una entidad Dirección. Cada persona tiene una única dirección y viceversa. Implementar esta relación proporciona una estructura más coherente y fácil de mantener en comparación con la duplicación de información.

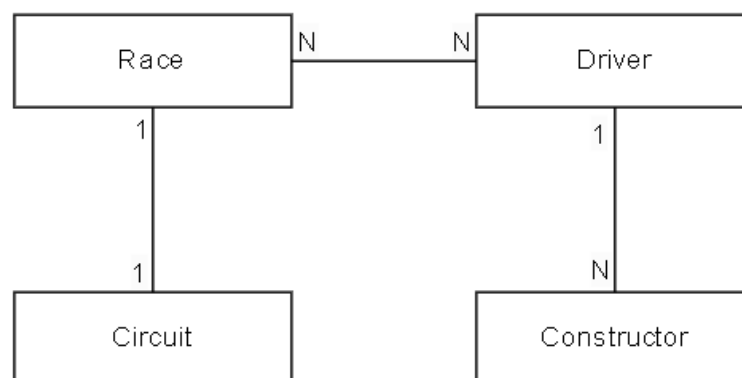
### OneToMany

La relación **OneToMany**, como su nombre sugiere, implica que un objeto de una entidad puede estar relacionado con varios objetos de otra entidad.

Supongamos una entidad Departamento y una entidad Empleado. Un departamento puede tener varios empleados, pero cada empleado solo pertenece a un departamento.

### Modelo Entidad-Relación de la práctica

El modelo con el que estamos trabajando en estas prácticas se puede simplificar con el siguiente diagrama:



Seguramente echas de menos la entidad Result. Recuerda que en los casos de relación con cardinalidad N:N, se crea una nueva tabla donde se almacenan las claves primarias de las dos entidades que participan en la relación. En nuestro caso, la cardinalidad N:N entre Race y Driver produce la entidad Result.

Las claves primarias de las entidades que participan en la relación pueden ser claves primarias de la nueva tabla, aunque también puede resolverse añadiendo un nuevo identificador autoincremental en lugar de la clave compuesta por las primarias de las participantes. De nuevo, este último es nuestro caso en la entidad Result.

### Parte 1: Creación y mapeo de entidades

Desarrolla las clases necesarias para mapear las entidades. Olvida por el momento las relaciones entre ellas. No olvides implementar el constructor vacío, getters y setters (o usa Lombok).

Realiza el mapeo de las entidades contra la base de datos mediante anotaciones.

### Parte 2: Relaciones entre entidades

Crea las relaciones OneToOne y OneToMany entre las entidades. Trabajaremos la relación ManyToMany en la siguiente práctica.

Comprueba con la API que tienes realizada de la práctica anterior que los **Drivers** contienen información de su **Constructor**.

### Parte 3: Operaciones CRUD

Añade un repositorio para **Constructor** y comprueba que se realizan el resto de operaciones CRUD sobre un **Driver** (y su **Constructor**).

Añade ahora un servicio para operar con **Race** y **Circuit**. Tendrás que definir también repositorios para estas entidades.