

Boletín

Ejercicio 1.- Crear un proyecto Maven llamado JPA_Hibernate_ejercicio1_24_jakarta.

- Se parte de una BD en MySQL llamada Ejercicio1 que esté vacía.
- Se crea un paquete entidades con las entidades Libro y Socio.
- Se añaden a dichas entidades las anotaciones correspondientes.
- A través de la interface EntityManager, probad cómo JPA maneja todas las operaciones CRUD sobre la entidad Libro.

Ejercicio 2.- Crear un proyecto Maven llamado JPA_Hibernate_ejercicio2_24_jakarta que permita la ejecución de un CRUD sobre una nueva BD ejercicio2 de Libros y Socios de la siguiente manera: Vamos a estudiar el patrón de diseño Facade.

Ejercicio 3.- Crear un proyecto Maven llamado JPA_Hibernate_ejercicio3_24_jakarta. Vamos a añadir asociaciones entre entidades y veremos cómo afecta al mapeo de la base de datos.

- 3.1 Crearemos una BD vacía ejercicio3.

3.2 Asociación unidireccional ONE to ONE sobre Empleado y Direccion

Crearemos dos POJOS Empleado y Direccion . Estableceremos una asociación entre sus entidades de tipo ONE to ONE Unidireccional. Una entidad Empleado contiene una referencia a una entidad Direccion, relación que ha sido declarada mediante la anotación @OneToOne. Empleado es el *dueño* de la relación de manera implícita, y por tanto cada entidad de este tipo contendrá por defecto una columna adicional en su tabla correspondiente de la base de datos (mediante la que podremos acceder al objeto Direccion). Esta columna es una *clave foránea* (foreign key).

3.3 Asociación bidireccional ONE to ONE sobre Empleado y PlazaParking:

- ambos lados de la relación deben estar anotados con @OneToOne, pero ahora uno de ellos debe indicar de manera explícita que la parte contraria es dueña de la relación.
- mappedBy en la anotación de asociación de la parte no-dueña

3.4 Asociación unidireccional ONE to MANY sobre Departamento y Empleado. (Un departamento tiene muchos empleados y un empleado solo puede pertenecer a un departamento.

3.5 Asociación bidireccional ONE to MANY – MANY to ONE sobre Departamento y Proyecto.

Ejercicio 4.- Crear un proyecto Maven llamado **JPA_Hibernate_ejercicio4_24_jakarta**. Vamos a añadir asociaciones entre entidades y veremos cómo afecta al mapeo de la base de datos.

- 3.1 Crearemos una BD vacía ejercicio4.

- 3.2 Vamos a añadir asociación **many to many** (en este caso bidireccional porque en ambas entidades figuran las colecciones) entre Socio y Libro de tal forma que al mapearlo se creará una tabla de base de datos PRESTAMOS.

Un Socio tiene una lista de libros prestados y desde un libro se accede a una lista de socios que lo han alquilado.

Ejercicio 5.- Crear un proyecto Maven llamado **JPA_Hibernate_ejercicio5_24_jakarta**. Vamos a añadir asociaciones entre entidades y veremos cómo afecta al mapeo de la base de datos.

- 1 Crearemos una BD llena ejercicio5 con el script de la biblioteca.

- 2 Generamos la entidades a partir de las tablas.