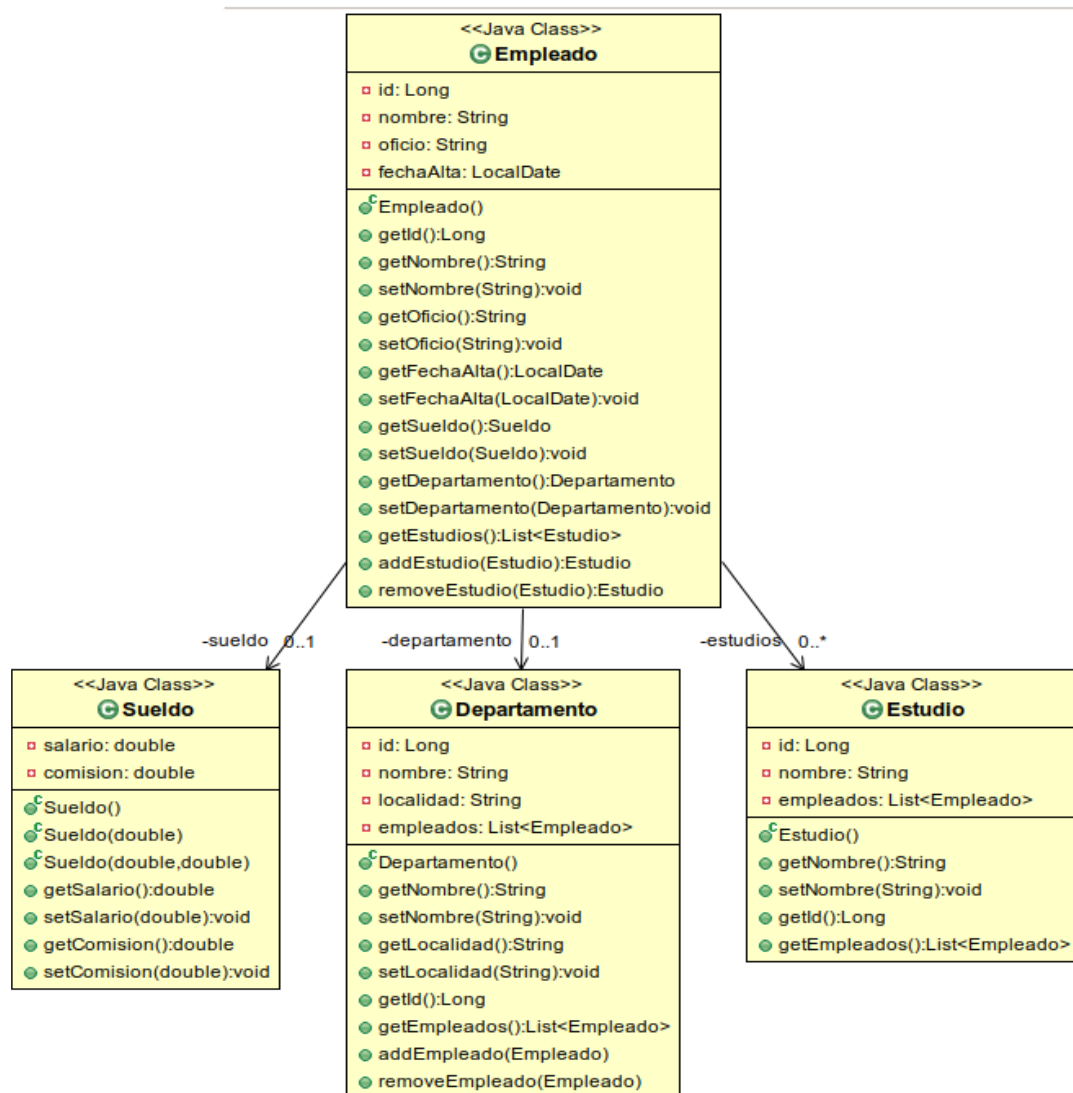


Hoja de ejercicios 2-04

JPA

EJERCICIOS

1.- Copia el proyecto anterior a otro llamado **Hoja2_04_1**. Ahora vamos a añadir una entidad *Estudio*. Un empleado podrá tener muchos estudios y un mismo estudio podrá haber sido realizado por muchos empleados. Haremos que esta relación **many-to-many** sea **bidireccional**.



Aunque no está incluido en el esquema, en *Estudio* debes incluir un atributo **codEstudio** de tipo String de 5 caracteres que se use como identificador natural. Usar la anotación **@NaturalId** para indicar que es un atributo que sirve para identificar (no habrá duplicados). Deben ser **NaturalId** los atributos que sirvan para identificar a los objetos en el mundo real.

Hoja de ejercicios 2-04

Para hacer algo funcionalmente igual, podemos anotar este atributo con **@Column(unique=true)** para que no admita repeticiones. Pero no es exactamente lo mismo, con **unique=true** establecemos la restricción a nivel de base de datos, con **NaturalId** se establece también a nivel de Hibernate. **NaturalId** proporciona muchas ventajas al trabajar con funciones nativas de Hibernate. **NaturalId** permite declarar de forma sencilla restricciones de unicidad sobre conjuntos de varios atributos.

En la clase App crea un estudio y establece que algunos empleados nuevos lo hayan cursado.

2.- Copia el proyecto anterior a **Hoja2_04_2**. Modificar ahora lo necesario para guardar también fecha de finalización del estudio para un empleado en concreto. Estaremos ante un caso de asociación **many-to-many con atributos extra**

Para ello será necesario crear las clases **EmpleadoEstudioId** y **EmpleadoEstudio**.

Con los estudios y empleados disponible desarrolla el programa para que pida un id de Estudio y un id de Empleado y, en caso de existir, añada el empleado al estudio (o viceversa).