

UA05. ORM

5.6.2. Uno a uno (bidireccional)

Esta es muy similar a la anterior, pero en éste caso la relación entre las clases **Alumnado** y **Direccion** va a ser bidireccional.

```
@Entity
@Table(name="Alumnado")
public class Alumnado implements Serializable {
    @Id
    @Column(name="IdAlum")
    private int idAlum;

    @Column(name="nombre")
    private String nombre;

    @Column(name="ape1")
    private String ape1;

    @Column(name="ape2")
    private String ape2;

    @OneToOne(cascade=CascadeType.ALL)
    @PrimaryKeyJoinColumn
    private Direccion direccion;
```

```
@Entity
@Table(name="Direccion")
public class Direccion implements Serializable {
    @Id
    @Column(name = "idAlum")
    private int idAlum;

    @Column(name="calle")
    private String calle;

    @Column(name="numero")
    private int numero;

    @Column(name="poblacion")
    private String poblacion;

    @Column(name="provincia")
    private String provincia;

    @OneToOne(cascade=CascadeType.ALL)
    @PrimaryKeyJoinColumn
    private Alumnado alumnado;
```

Para realizar una prueba, usaremos el siguiente código:

```
Direccion direccion1=new Direccion(3, "Calle de la sarten", 23, "Mani
Alumnado alumnado1=new Alumnado(3, "Sergio", "Mateo", "Ramis");
alumnado1.setDireccion(direccion1);
direccion1.setAlumnado(alumnado1);
Direccion direccion2=new Direccion(4, "Calle Luis lamarca", 45, "Torr
Alumnado alumnado2=new Alumnado(4, "Paco", "Moreno", "Díaz");
alumnado2.setDireccion(direccion2);
direccion2.setAlumnado(alumnado2);
```

El ejemplo incluye dos casos:

- Crear un objeto **direccion1** y otro **alumnado1**, crear las relaciones y finalmente en guardar el objeto **alumnado1**.
- Crear un objeto **direccion2** y otro **alumnado2**, crear las relaciones y finalmente guardar el objeto **direccion2**.

En ambos casos el resultado aparente es el mismo, se guarda tanto el objeto **Direccion** como el objeto **Alumnado** al ser la relación bidireccional aunque realmente los 2 casos no son iguales.

En el primer caso, si persistimos un objeto **Alumnado** se inserta directamente dicho objeto en la base de datos por lo que no puede existir ya la fila pero sí que se permite que la **Direccion** ya exista, actualizándose en dicho caso.

Pero en el segundo caso si persistimos el objeto **Direccion** lo que ocurre es lo contrario, no podrá existir la fila de la dirección pero sí podrá existir la fila del **Alumnado**.

Obra publicada con ~~Licencia Creative Commons Reconocimiento No comercial Compartir igual~~
4.0 <<http://creativecommons.org/licenses/by-nc-sa/4.0/>>