CRUD con Doctrine ORM

Fernando Díaz-Alonso Dorado

CRUD

¿Qué es?

Cuando hablamos de CRUD en la Gestión de una Base de Datos estamos hablando de las funcionalidades:

- **Create** → Insertar nuevos registros en la BB.DD.
- **Read** → Leer lo datos de los registros que tiene la BB.DD.
- **Update** → Actualizar los datos de los registros que existen en la BB.DD.
- **Delete** → Borrar los datos de algún registro o todos los existentes en la BB.DD.

Estas funcionalidades son esenciales para poder gestionar y hacer uso de cualquier BB.DD. Por tanto, en Doctrine también tenemos métodos para poder hacer CRUD.

Persistencia de datos

Persist

Como ya introdujimos, Doctrine es una capa que conecta nuestra aplicación con la BB.DD. y los datos, no son actualizados inmediatamente.

Para que los datos sean actualizados debemos indicarle a Doctrine que debe en primera instancia guardar esos datos.

Para ello, tenemos el método del EntityManager **PERSIST**. Que se encarga de guardar en memoria los cambio que se han realizado en la entidad que le indiquemos.

EntityManager->persist($entidad)

- A partir de ese momento la entidad será administrada por un EntityManager.
- Podemos tener en estado tantas entidades como sea necesario.
- Pensemos que es como un **buffer de persistencia**.

Persistencia de datos

Flush

Persist guarda los datos, pero no realiza el INSERT (SQL) a la BB.DD. Hasta que se realiza la llamada al método del EntityManager **FLUSH** que realiza la sincronización con la BB.DD.

EntityManager->flush()

- Realmente, Doctrine va a retrasar la mayoría de los ***comandos SQL*** hasta que se invoque éste método.
- Flush emitirá todas las ***sentencias SQL necesarias*** para sincronizar sus objetos con la base de datos.
- De esta manera la operación es ***más eficiente*** y en una única y corta transacción.
- Además flush se encarga de mantener la ***integridad referencial***.

CRUD

Inserción

Los pasos a seguir son:

1. Crear un nuevo objeto a partir de la entidad.
2. Se insertan los valores de los atributos del objeto a través de los setters.
3. Se envía el objetos al buffer de persistencia.

EntityManager→persist($entidad)

4. Se inserta un nuevo registro en la BB.DD. a partir del objeto con flush() desde el buffer de persistencia.

EntityManager→flush()

CRUD

Inserción II

1. Creamos un nuevo objeto desde la entidad Users.
2. Usamos los setters para insertar la nueva información.
3. Enviamos al buffer de persistencia para guardar los datos.
4. Y lanzamos la sincronización con la BB.DD.

Es importante indicar que el entityManager sobre el se realiza el flush debe ser el mismo que se ha usado para hacer el persist.

CRUD

Lectura

1. Ya vimos los métodos para poder acceder a los datos es necesario obtener el repositorio de la entidad desde el ***EntitnyManager***.
2. Mediante los métodos predefinidos de los respositorios de las entidades al extender del ***EntityRepository.***
3. Con el método ***findAll()*** obtenemos todos los registros de la entidad Users.
4. Con el método ***find()*** obtenemos el registro correspondiente a la id.

```php
$userRepository = $entityManager->getRepository(Users::class);
$users = $userRepository->findAll();
$user = $userRepository->find($id);
```

CRUD

Actualización

Los pasos a seguir son:

1. Obtenemos el objeto a partir del repositorio de la entidad, registro en la BBDD.
2. Se modifican, actualizan, los valores de los atributos del objeto a través de los setters.
3. Se envía el objetos al buffer de persistencia.

EntityManager→persist($entidad)

4. Se inserta un nuevo registro en la BB.DD. a partir del objeto con flush() desde el buffer de persistencia.

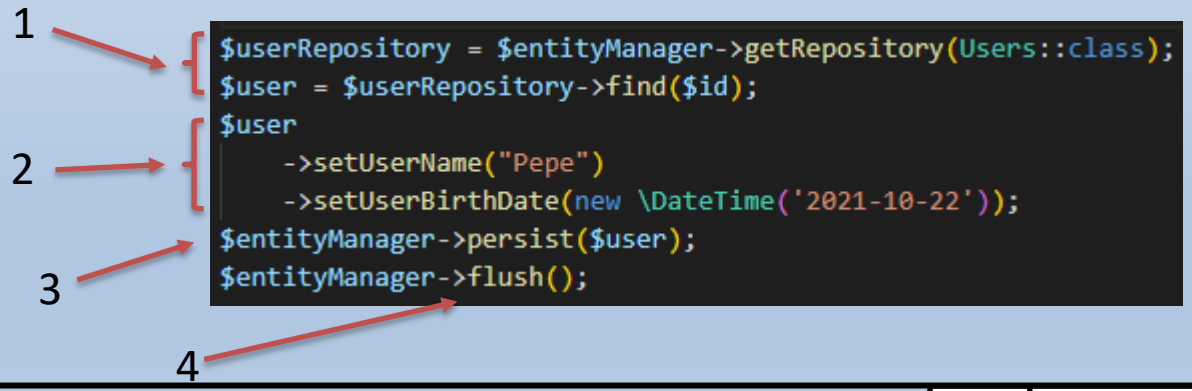
EntityManager→flush()

CRUD

Actualización II

1. Obtenemos el objeto desde el repositorio de la entidad Users.
2. Usamos los setters para modificar la nueva información, que contiene.
3. Enviamos al buffer de persistencia para guardar los datos.
4. Y lanzamos la sincronización con la BB.DD.

Es importante indicar que el entityManager sobre el se realiza el flush debe ser el mismo que se ha usado para hacer el persist.



CRUD

Borrado

Los pasos a seguir son:

1. Obtenemos el objeto a partir del repositorio de la entidad, registro en la BBDD.
2. Se elimina el objeto a través del método remove.

EntityManager→remove($entidad)

3. Se inserta un nuevo registro en la BB.DD. a partir del objeto con flush() desde el buffer de persistencia.

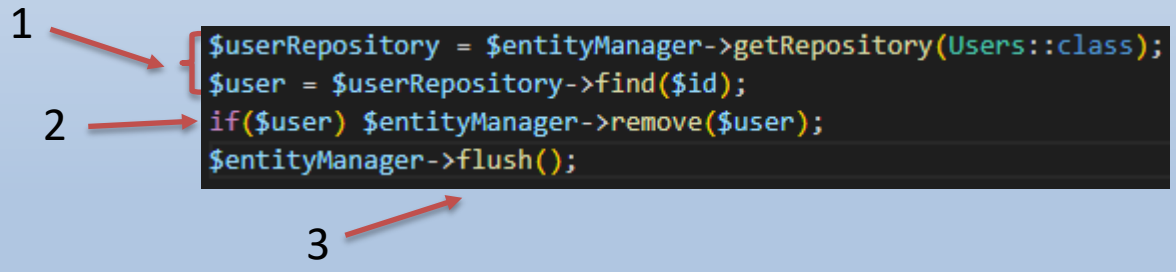
EntityManager→flush()

CRUD

Borrado II

1. Obtenemos el objeto desde el repositorio de la entidad Users.
2. Eliminamos la información que contiene el objeto user, mediante el método **remove()** que a su vez también lo envía al buffer de persistencia para guardar los datos.
3. Y lanzamos la sincronización con la BB.DD.

Es importante indicar que el entityManager sobre el se realiza el flush debe ser el mismo que se ha usado para hacer el persist.



Doctrine

Bibliografía

- https://www.doctrine-project.org/projects/doctrine-orm/en/2.13/reference/working-with-objects.html#working-with-objects
- https://diego.com.es/symfony-y-doctrine