

## UA05. ORM

### 5.4. Sesiones. Estados de un objeto

Para poder utilizar la persistencia en Hibernate es necesario definir un objeto **Session** utilizando la clase **SessionFactory**. La sesión corresponde con un objeto que representa una unidad de trabajo con la base de datos. La sesión nos permite representar el gestor de persistencia, ya que dispone de una API básica que nos permite cargar y guardar objetos.

La sesión está formada internamente por una cola de sentencias SQL que son necesarias ejecutar para poder sincronizar el estado de la sesión con la base de datos.

Asimismo, la **sesión** contiene una lista de objetos persistentes. Una sesión corresponde con el primer nivel de caché.

Si para realizar el acceso a datos, usamos Hibernate, la sesión nos permite definir el alcance de un contexto determinado. Para poder utilizar los mecanismos de persistencia de Hibernate se debe inicializar el entorno Hibernate y obtener un objeto Session utilizando la clase SessionFactory de Hibernate. Un objeto Session Hibernate representa una única unidad-de-trabajo para un almacén de datos dado y lo abre un ejemplar de SessionFactory. Se deben cerrar las sesiones cuando se haya completado todo el trabajo de una transacción.

Para obtener un ejemplar de SessionFactory realizaremos:

```
protected void setUp() throws Exception {  
    final StandardServiceRegistry registry = new StandardServiceRegistry  
        .configure() // por defecto: hibernate.cfg.xml  
        .build();  
    try {
```

```
sessionFactory = new MetadataSources( registry ).buildMetadat
}
catch (Exception e) {
    StandardServiceRegistryBuilder.destroy( registry );
}
}
```

Los estados en los que se puede encontrar un objetos son:

- **Transitorio (Transient)**: En este estado estará un objeto recién creado que no ha sido enlazado con el gestor de persistencia.
- **Persistente**: En este caso el objeto está enlazado con la sesión. Todos los cambios que se realicen serán persistentes.
- **Disociado (Detached)**: En este caso nos encontramos con un objeto persistente que sigue en memoria después de que termine la sesión. En este caso existe en Java y en la base de datos.
- **Borrado (Removed)**: En este caso el objeto está marcado para ser borrado de la base de datos. Existe en la aplicación Java y se borrará de la base de datos al terminar la sesión.

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