

CURSO DE HIBERNATE 5





## HIBERNATE



### **HIBERNATE**

Más que un ORM. Comparativa con otros productos. JPA. Mayen, Módulos



### (4)**ENTIDADES**

Definición del modelo del dominio. Entidades y ciclo de vida. XML y anotaciones. Tipos de datos.





Persistencia, desfase objecto-relacional, ORM. Productos y estándares



### (3)**PRIMER PROYECTO**

Hibernate.cfg.xml, EntityManager y persistence.xml



### **ASOCIACIONES**

ManyToOne, OneToMany, OneToOne, ManyToMany



## HIBERNATE



### COLECCIONES

Mapeo de colecciones. Tipos (list, set, map). Colecciones ordenadas (sorted vs. ordered).



### CONTEXTODE PERSISTENCIA

Almacenamiento. recuperación y borrado de entidades.







Campos calculados, herencia.



### **GENERACION DEL ESQUEMA**

Customización del proceso de generación del esquema.



### (10)TRANSACCIONES

Control de concurrencia. Patrones y antipatrones.





#### (12) ENVERS

Introducción a la auditoria de entidades.





(11) CONSULTAS HPQL VS JPQL



Consultas con parámetros, Anotaciones. SQL nativo

## 1. PRERREQUISITOS



## ENTORNO (IDE)

STS nos permite realizar las mismas tareas que Eclipse (Neon) pero tiene además soporte *nativo* para Spring.



Forman parte de Jboss Tools, y añaden funcionalidades Hibernate a Eclipse/STS



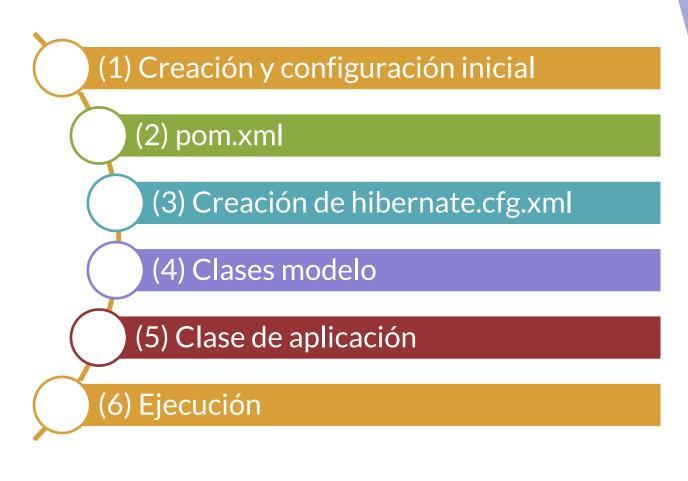


## **MYSQL**

Si bien ya no es software libre, su versión *community* es uno de los RDBMS más usados del mercado. Además, el gestor Mysql Workbench nos facilitará algunas tareas si no somos expertos en SQL.

# 2. PASOS PARA CREAR EL NUEVO PROYECTO

¡A por ellos!



### (1) Creación y configuración inicial

- Creamos un nuevo proyecto Maven
- ► Escogemos el arquetipo *quickstart*
- Definimos datos iniciales (groupId, artifact y package)
- ► Cambiamos la versión de java de 1.5 a 1.8

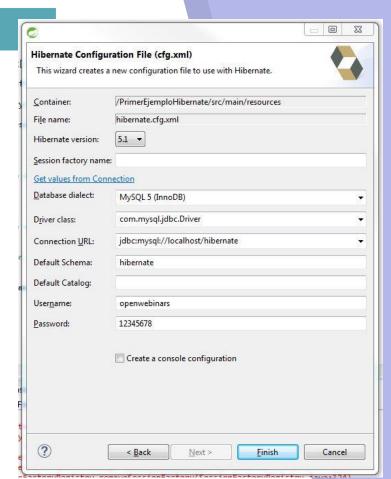
## (2) pom.xml

```
<dependencies>
 <!-- Otras dependencias, como la de jUnit... -->
     <dependency>
         <groupId>org.hibernate
         <artifactId>hibernate-core</artifactId>
         <version>5.2.9.Final
     </dependency>
     <dependency>
         <groupId>mysql</groupId>
         <artifactId>mysql-connector-java</artifactId>
         <version>5.1.41</version>
     </dependency>
 </dependencies>
```



### (3) Creación de hibernate.cfg.xml

- Creamos un Source Folder con ruta /src/main/resources.
- New > Other ... > Hibernate > Hibernate Configuration File
- Rellenamos los datos que correspondan.



## (3) Creación de hibernate.cfg.xml

► En la pestaña Hibernate establecemos las propiedades Show sql, Format sql y Hbm2ddl Auto.

## (4) Clases modelo

- ► Anotamos la clase con @Entity
- Anotamos la propiedad que será clave primaria con @Id
- Anotamos el resto de columnas con
   @Column

## (5) Clase de aplicación

- ► Inicializamos un SessionFactory con la configuración definida anteriormente.
- ► Abrimos una sesión.
- Creamos nuevas instancias.
- Iniciamos una transacción.
- Persistimos los datos.
- Commiteamos la transacción.
- Cerramos los objetos.