CURSO DE HIBERNATE 5

OpenWebinars

HIBERNATE

(1) INTRODUCCION

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

(2) HIBERNATE

Más que un ORM. Comparativa con otros productos. JPA. Maven. Módulos

(3) PRIMER PROYECTO

Hibernate.cfg.xml, EntityManager y persistence.xml

(4) ENTIDADES

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

(5) ASOCIACIONES

ManyToOne, OneToMany, OneToOne, ManyToMany

HIBERNATE

(6) ELEMENTOS AVANZADOS

Campos calculados, herencia.

(7) COLECCIONES

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

(8) GENERACION DEL ESQUEMA

Customización del proceso de generación del esquema.

(9) CONTEXTO DE PERSISTENCIA

Almacenamiento, recuperación y borrado de entidades.

(10) TRANSACCIONES

Control de concurrencia. Patrones y antipatrones.

HIBERNATE

(12)
ENVERS

Introducción a la
auditoria de entidades.

(11)
CONSULTAS
HPQL VS JPQL

Consultas con
parámetros,
Anotaciones. SQL nativo

1.

GENERACIÓN AUTOMÁTICA DEL ESQUEMA

GENERACIÓN AUTOMÁTICA

- Definimos las clases entidad, asociaciones, clases *embedd*, …
- Hibernate se encarga de generar el esquema en la base de datos.

Solo deberíamos utilizar esta funcionalidad para desarrollo y testing. En producción es mas adecuado el uso de scripts sql.

GENERACIÓN AUTOMÁTICA (HBM2DDL AUTO)

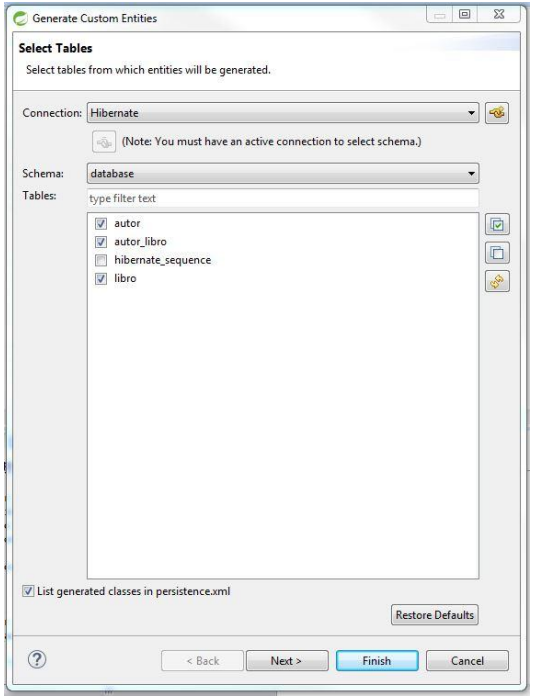
- **none**: valor por defecto. No realiza acciones.
- **create-only**: proceso de creación de la base de datos.
- **drop**: realiza el borrado de la base de datos.
- **create**: borra la base de datos y después la crea.
- **create-drop**: borra, crea y borra al cerrar el *SessionFactory/EntityManagerFactory*.
- **validate**: valida el esquema de la base de datos.
- **update**: actualiza la base de datos con los cambios necesarios.

2.

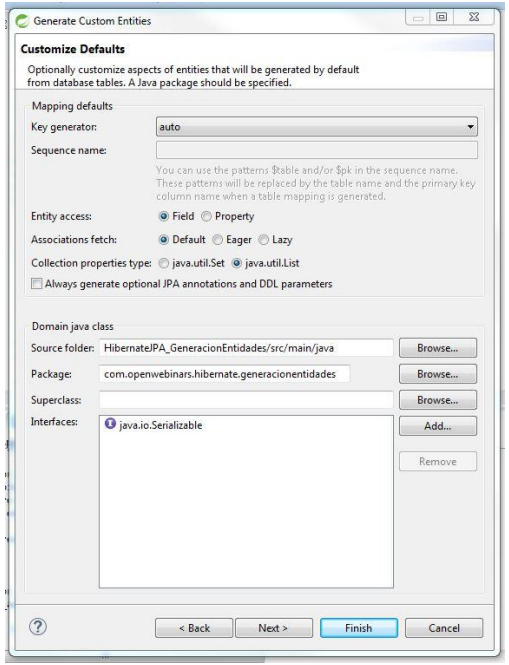
GENERACIÓN DE LAS ENTIDADES A PARTIR DEL ESQUEMA

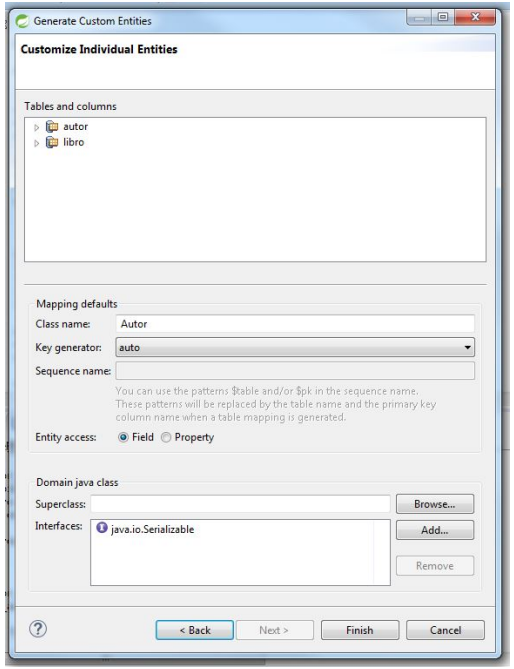
GENERACIÓN AUTOMÁTICA DE ENTIDADES

- Funcionalidad de Eclipse/STS.
- Escanea la base de datos y genera las entidades, asociaciones, embedd, …
- Selección de tablas
- Permite la selección de algunos parámetros.

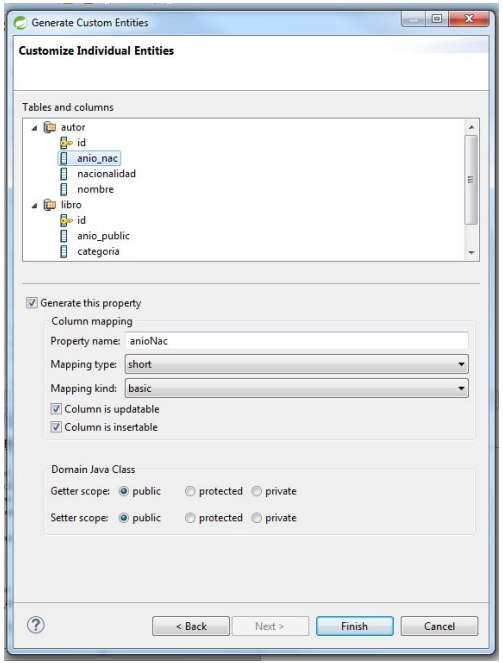


GENERACIÓN AUTOMÁTICA DE ENTIDADES





GENERACIÓN AUTOMÁTICA DE ENTIDADES




```java
@Entity
@NamedQuery(name="Autor.findAll", query="SELECT a FROM Autor a")
public class Autor implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    private long id;

    @Column(name="anio_nac")
    private short anioNac;

    private String nacionalidad;

    private String nombre;

    //bi-directional many-to-many association to Libro
    @ManyToMany
    @JoinTable(
        name="autor_libro"
        , joinColumns={
            @JoinColumn(name="id_autor")
            }
        , inverseJoinColumns={
            @JoinColumn(name="id_libro")
            }
        )
    private List<Libro> libros;
```