



# **CURSO DE HIBERNATE 5**

  
**OpenWebinars**



# HIBERNATE

## (2) HIBERNATE



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



## (4) ENTIDADES

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



## (1) INTRODUCCION

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



## (3) PRIMER PROYECTO

Hibernate.cfg.xml,  
EntityManager y  
persistence.xml



## (5) ASOCIACIONES

ManyToOne, OneToMany,  
OneToOne, ManyToMany



# HIBERNATE

(7)

## COLECCIONES



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

(9)

## CONTEXTO DE PERSISTENCIA



Almacenamiento,  
recuperación y borrado  
de entidades.



(6)

## ELEMENTOS AVANZADOS

Campos calculados,  
herencia.



(8)

## GENERACION DEL ESQUEMA

Customización del  
proceso de  
generación del  
esquema.



(10)

## TRANSACCIONES

Control de concurrencia.  
Patrones y antipatrones.



# HIBERNATE



## (11) CONSULTAS HQL VS JPQL

Consultas con  
parámetros,  
Anotaciones. SQL nativo

## (12) ENVERS

Introducción a la  
auditoria de entidades.





1.

# CONFIGURACIÓN

# CONFIGURACIÓN

```
<dependency>  
  <groupId>org.hibernate</groupId>  
  <artifactId>hibernate-envers</artifactId>  
  <version>5.2.9.Final</version>  
</dependency>
```

Para cada entidad que queremos auditar, tenemos que incluir la anotación @Audited

# OPCIONES DE CONFIGURACIÓN

- ▶ `org.hibernate.envers.audit_table_prefix`
- ▶ `org.hibernate.envers.audit_table_suffix`
- ▶ `org.hibernate.envers.revision_field_name`
- ▶ `org.hibernate.envers.revision_type_field_name`
- ▶ `org.hibernate.envers.audit_strategy`
- ▶ `org.hibernate.envers.audit_strategy_validity_store_revent_timestamp`



2.

UTILIZACIÓN



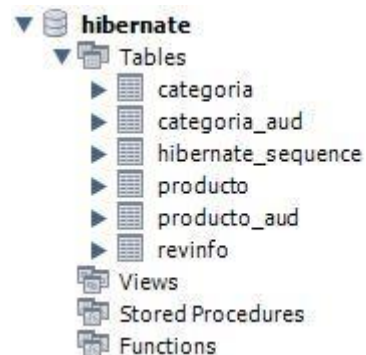
# UTILIZACIÓN

```
@Entity
@Audited
public class Categoria implements Serializable {

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

    //Resto de atributos o métodos

}
```



# UTILIZACIÓN

```
Categoria[] categorias = {
    new Categoria("Ordenadores y portátiles"),
    new Categoria("Smartphones y telefonía"),
    new Categoria("Imagen y sonido")
};

addCategorias(Arrays.asList(categorias));

public static void addCategorias(List<Categoria> list) {
    em.getTransaction().begin();
    for (Categoria c : list)
        em.persist(c);
    em.getTransaction().commit();
}
```

id	nombre
1	Ordenadores v portátiles
2	Smartphones v telefonía
3	Imagen v sonido
NULL	NULL

id	REV	REVTYPE	REVENID	nombre
1	1	0	NULL	Ordenadores v portátiles
2	1	0	NULL	Smartphones v telefonía
3	1	0	NULL	Imagen v sonido
NULL	NULL	NULL	NULL	NULL

id	REV	REVTYPE	REVENID	descripcion	nombre	pvp	categoria_id
4	2	0	5	Apple MacoBook Pro Retina Displav Intel i7/16G...	Macbook Pro	2058	1
4	5	1	6	Apple MacoBook Pro Retina Displav Intel i7/16G...	Macbook Pro	1749.3	1
4	6	1	7	Apple MacoBook Pro Retina Displav Intel i7/16G...	Macbook Pro	2186.625	1
4	7	1	8	Apple MacoBook Pro Retina Displav Intel i7/16G...	Macbook Pro	1967.9625	1
4	8	1	9	Apple MacoBook Pro Retina Displav Intel i7/16G...	Macbook Pro	2322.19575	1
4	9	2	NULL	NULL	NULL	NULL	NULL
5	2	0	5	Asus D540SA-XX621 Intel Celeron N3060/4GB/5...	Asus D540SA-XX621	229	1
5	5	1	6	Asus D540SA-XX621 Intel Celeron N3060/4GB/5...	Asus D540SA-XX621	194.65	1
5	6	1	7	Asus D540SA-XX621 Intel Celeron N3060/4GB/5...	Asus D540SA-XX621	243.3125	1
5	7	1	8	Asus D540SA-XX621 Intel Celeron N3060/4GB/5...	Asus D540SA-XX621	218.98125000000002	1
5	8	1	9	Asus D540SA-XX621 Intel Celeron N3060/4GB/5...	Asus D540SA-XX621	258.397875	1
5	9	2	NULL	NULL	NULL	NULL	NULL
6	2	0	5	Ho Soectre 13-V10 1NS Intel Core i7-7500U/8GB...	Ho Soectre 13-V10...	1499	1
6	5	1	6	Ho Soectre 13-V10 1NS Intel Core i7-7500U/8GB...	Ho Soectre 13-V10...	1274.1499999999999	1
6	6	1	7	Ho Soectre 13-V10 1NS Intel Core i7-7500U/8GB...	Ho Soectre 13-V10...	1592.6874999999998	1
6	7	1	8	Ho Soectre 13-V10 1NS Intel Core i7-7500U/8GB...	Ho Soectre 13-V10...	1433.4187499999998	1
6	8	1	9	Ho Soectre 13-V10 1NS Intel Core i7-7500U/8GB...	Ho Soectre 13-V10...	1691.4341249999998	1
6	9	2	NULL	NULL	NULL	NULL	NULL
7	3	0	5	Apple iPhone 7 128 GB Roio Edición Limitada	Iphone 7 Roio	869	2
7	5	1	6	Apple iPhone 7 128 GB Roio Edición Limitada	Iphone 7 Roio	738.65	2
7	6	1	7	Apple iPhone 7 128 GB Roio Edición Limitada	Iphone 7 Roio	923.3125	2
7	7	1	8	Apple iPhone 7 128 GB Roio Edición Limitada	Iphone 7 Roio	830.98125	2
7	8	1	9	Apple iPhone 7 128 GB Roio Edición Limitada	Iphone 7 Roio	980.557875	2
7	9	2	NULL	NULL	NULL	NULL	NULL
8	3	0	5	Samsuna Galaxv J5 2016 Nearo Dual Libre	Samsuna Galaxv J5	189	2
8	5	1	6	Samsuna Galaxv J5 2016 Nearo Dual Libre	Samsuna Galaxv J5	160.65	2
8	6	1	7	Samsuna Galaxv J5 2016 Nearo Dual Libre	Samsuna Galaxv J5	200.8125	2
8	7	1	8	Samsuna Galaxv J5 2016 Nearo Dual Libre	Samsuna Galaxv J5	180.73125000000002	2
8	8	1	9	Samsuna Galaxv J5 2016 Nearo Dual Libre	Samsuna Galaxv J5	213.262875	2
8	9	2	NULL	NULL	NULL	NULL	NULL
9	3	0	5	Ba X5 Plus 32GB Nearo	Ba X5 Plus	289	2
9	5	1	6	Ba X5 Plus 32GB Nearo	Ba X5 Plus	245.65	2
9	6	1	7	Ba X5 Plus 32GB Nearo	Ba X5 Plus	307.0625	2
9	7	1	8	Ba X5 Plus 32GB Nearo	Ba X5 Plus	276.35625	2
9	8	1	9	Ba X5 Plus 32GB Nearo	Ba X5 Plus	326.100375	2
9	9	2	NULL	NULL	NULL	NULL	NULL



3.

## CONSULTA DE VALORES AUDITADOS

# CONSULTA

```
AuditReader reader = AuditReaderFactory.get(entityManager);
```

```
AuditQuery query = reader.createQuery().forRevisionsOfEntity(Producto.class, false, true);
```

```
List<Object[]> results = query.getResultList();
```

```
for (Object[] obj : results) {
```

```
    Producto p = (Producto) obj[0];
```

```
    DefaultRevisionEntity dre = (DefaultRevisionEntity) obj[1];
```

```
    RevisionType revType = (RevisionType) obj[2];
```

```
    System.out.println(String.format("(Id: %d) - Producto: %s (%.2f €) (%s el %s)", dre.getId(), p.getNombre(), p.getPvp(),  
        revType.toString(), dre.getRevisionDate()));
```

```
}
```