

# Herramientas de mapeo objeto-relacional

# Herramientas de mapeo objeto-relacional

## Objetivos de la unidad

- Conocer el concepto de mapeo objeto-relacional
- Ejemplo de aplicación: Hibernate
  - Instalar y configurar la herramienta ORM
  - Definir los ficheros de mapeo
  - Aplicar mecanismos de persistencia

Documentación y descargas: <http://hibernate.org/orm/>

# Herramientas de mapeo objeto-relacional

## Concepto de mapeo

- El mapeo objeto-relacional (más conocido con sus siglas [ORM](#)) es una técnica de programación que permite convertir datos entre el sistema de tipos utilizado en un lenguaje de programación y el utilizado en una base de datos relacional.



# Herramientas de mapeo objeto-relacional

## Concepto de mapeo

- El modelo relacional trata con relaciones y conjuntos de datos.
- El paradigma OO trata con clases de objetos, objetos, atributos, métodos y asociaciones entre objetos.
- Un mapeo objeto-relacional (ORM) tiene como misión evitar estas diferencias y facilitar la tarea del programador.

# Herramientas de mapeo objeto-relacional

## Características

- Ventajas

- Rapidez en el desarrollo
- Abstracción de la base de datos
- Reutilización
- Mantenimiento del código

- Desventajas

- Tiempo / curva de aprendizaje
- Aplicaciones algo más lentas



# Herramientas de mapeo objeto-relacional

## Estructura de ficheros

- Dos puntos / aspectos principales
  - Las clases Java que representan los objetos persistentes (los que tienen correspondencia con las tablas de la base de datos relacional)
  - El fichero de mapeo (.hm.xml) que indica la correspondencia entre los atributos de una clase y los campos de la tabla con la que está asociado

# Herramientas de mapeo objeto-relacional

## Estructura de clases

- Clases POJO
  - Constructor público sin argumentos
  - *Getters* y *Setters* de todos los atributos que queramos hacer persistentes

# EJEMPLO DE FICHERO DE MAPEO

```
<?xml version="1.0"?>
<!DOCTYPE ...

<hibernate-mapping>
    <class name="modelos.Director" table="directores">
        <id name="idDirector" type="int">
            <column name="IDDIRECTOR" />
            <generator class="increment" />
        </id>

        <property name="nombre" type="string">
            <column name="NOMBREDIRECTOR" length="30"
                   not-null="true" />
        </property>
        <property name="apellidos" type="string">
            <column name="APELLIDOSDIRECTOR" length="30"
                   not-null="true" />
        </property>
    </class>
</hibernate-mapping>
```

```
<?xml version="1.0"?>  
<!DOCTYPE ...
```

Clase JAVA  
(y paquete)

```
<hibernate-mapping>  
  <class name="modelos.Director" table="directores">  
    <id name="idDirector" type="int">  
      <column name="IDDIRECTOR" />  
      <generator class="increment" />  
    </id>
```

Tabla BBDD

```
    <property name="nombre" type="string">  
      <column name="NOMBREDIRECTOR" length="30"  
             not-null="true" />  
    </property>  
    <property name="apellidos" type="string">  
      <column name="APELLIDOSDIRECTOR" length="30"  
             not-null="true" />  
    </property>  
  </class>
```

Nombre del atributo en la clase

Nombre campo BBDD

```
<?xml version="1.0"?>
<!DOCTYPE ...

<hibernate-mapping>
    <class name="modelos.Director" table="directores">
        <id name="idDirector" type="int">
            <column name="IDDIRECTOR" />
            <generator class="increment" />
        </id>
        <property name="nombre" type="string">
            <column name="NOMBREDIRECTOR" length="30"
                   not-null="true" />
        </property>
        <property name="apellidos" type="string">
            <column name="APELLIDOSDIRECTOR" length="30"
                   not-null="true" />
        </property>
    </class>
</hibernate-mapping>
```

Capo ID  
(Clave autoincremental)

# EJEMPLO DE FICHERO DE CONFIGURACIÓN

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE ...>
<hibernate-configuration>
    <session-factory>
        <mapping resource="modelos/Cine.hbm.xml" />
        <property name="hibernate.connection.driver_class">
            com.mysql.jdbc.Driver
        </property>
        <property name="hibernate.connection.password">
            dam
        </property>
        <property name="hibernate.connection.username">
            dam
        </property>
        <property name="hibernate.connection.url">
            jdbc:mysql://localhost:3306/CineHiber
        </property>
        ...
    </session-factory>
</hibernate-configuration>
```

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE ...>
<hibernate-configuration>
    <session-factory>
        <mapping resource="modelos/Cine.hbm.xml" />
        <property name="hibernate.connection.driver_class">
            com.mysql.jdbc.Driver
        </property>
        <property name="hibernate.connection.password">
            dam
        </property>
        <property name="hibernate.connection.username">
            dam
        </property>
        <property name="hibernate.connection.url">
            jdbc:mysql://localhost:3306/CineHiber
        </property>
        ...
    </session-factory>
</hibernate-configuration>
```

Localización de los  
ficheros de mapeo

Driver y datos de  
conexión con la BBDD  
(URL, User, Password)

