

# Mantenimiento y Gestión del Cambio

en Sistemas Software

Grado en Ingeniería Informática

**Práctica 3 Herramientas de Mapeo**

**OBJETIVOS**

* Realizar un programa que acceda a la BD para realizar operaciones de consulta y modificación utilizando la herramienta *Hibernate*
* Configurar *Hibernate* para que realice el mapeo en la capa de persistencia
* Utilizar las clases mapeadas para llevar a cabo sentencias de actualización y consulta

Para realizar esta práctica necesitarás cargar el esquema ***esquemaCasosPoliciales.sql*** disponible en la página web de la asignatura.

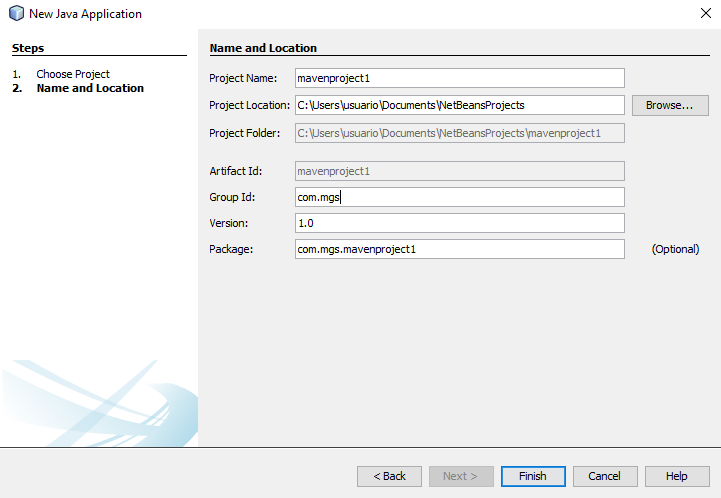
## Ejercicio 1

* Realizar el mapeo de las clases del esquema CasosPoliciales, y, utilizando la herramienta *Hibernate*.

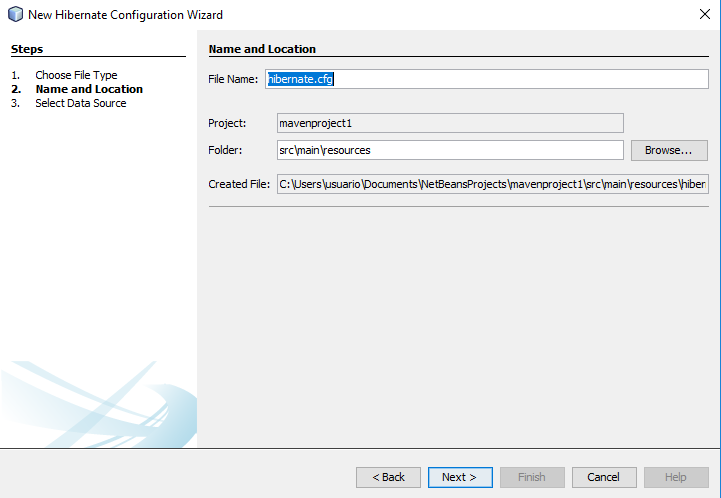
1.Crear el esquema en mysql a partir del fichero esquemaCasosPoliciales.sql



2. Crear un proyecto maven en Netbeans.



3. new/hibernate configuration wizard.

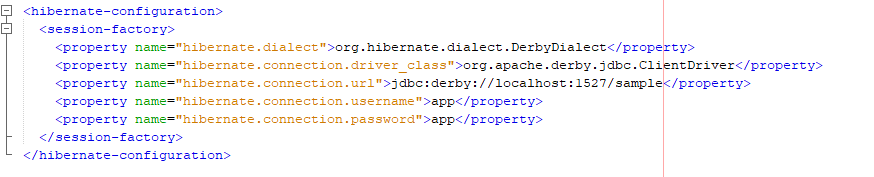


4. new database conection

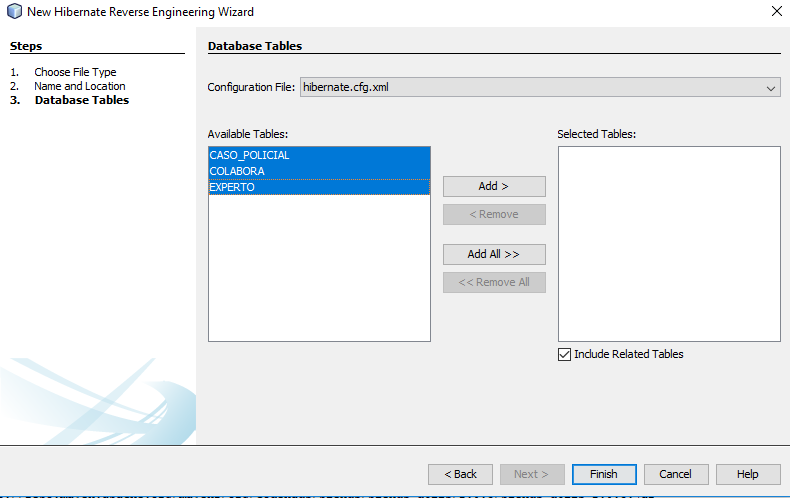
5. configuracion. Recordad cambiar el nombre de la bd por casospoliciales. El password de root lo dejamos vacío.

Hemos usado la BD de ejemplo de Netbeans, podemos usarla como jdbc:derby://localhost:1527/sample [app on APP]

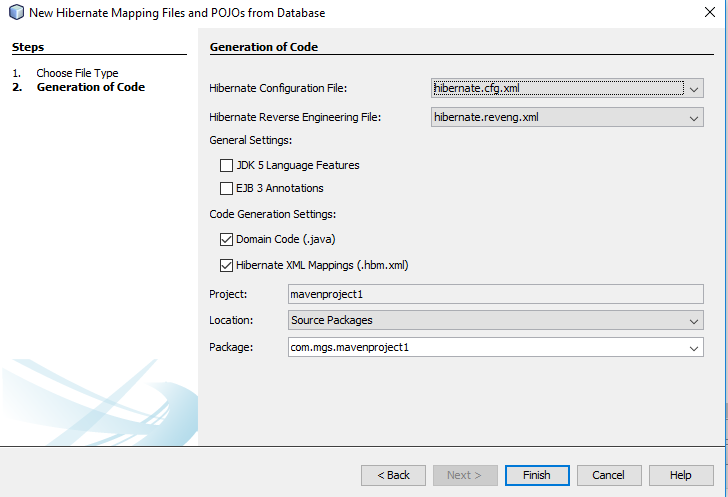
6. Si el fichero de configuracion xml nos sale en modo diseño, con el boton derecho, editors, source y nos saldrá en modo texto.



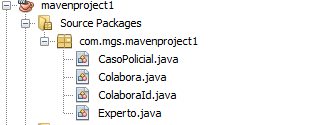
7. Ingeniería inversa.



8. Obtener POJOs.



9. Ya tenemos la BD mapeada.



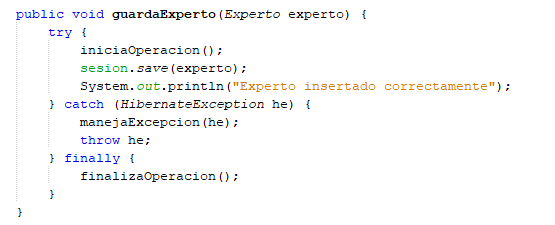
## Ejercicio 2

Utilizar las clases creadas en el Ejercicio 1 y las facilidades de *Hibernate*

para implementar los siguientes métodos:

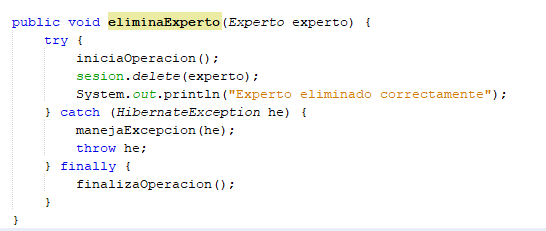
* Guardar un experto en la base de datos

public void guardaExperto(Experto experto)



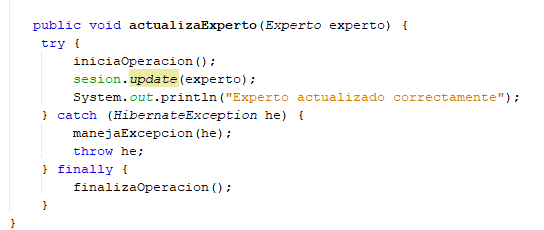
* Eliminar un experto de la base de datos

public void eliminaExperto(Experto experto)



* Actualizar un experto existente en la base de datos

public void actualizaExperto(Experto experto)



* Mostrar los datos de un experto

public Experto obtenExperto(String idExperto)

Todos los métodos anteriores deben formar parte de la nueva clase manejaExperto. Para nuestro control interno de versiones, crearemos un TAG llamado “ejercicio 1” y lo subiremos a un repositorio en nuestra organización MGCSS en GitHub.

Se debe realizar una batería de pruebas para todos métodos de este ejercicio.

## Ejercicio 2

Del mismo modo, añadir los siguientes métodos a manejaExperto

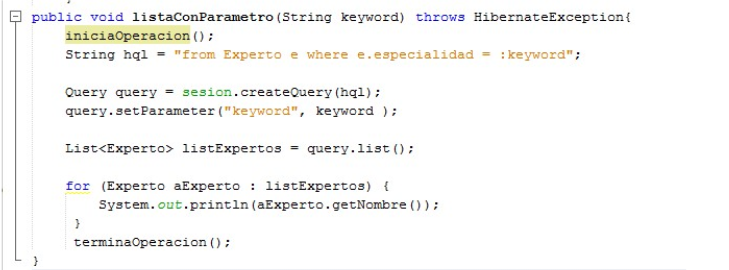
* Mostrar todos los nombres y especialidades de cada uno de los expertos

de la tabla Experto

public void obtenNombresyEspecialidad()

* Mostrar todos los expertos cuya especialidad sea la pasada por parámetro

public void listaConParametro(String keyword)



* Listar los nombres de todos los expertos y de todos los casos en los que hayan participado

public void obtenCasos()

Para nuestro control interno de versiones, crearemos un TAG llamado “ejercicio 2” y lo subiremos a un repositorio en nuestra organización MGCSS en GitHub

## Ejercicio 3

* Comenta, brevemente, cuales son, en tu opinión, las principales ventajas e inconvenientes de utilizar herramientas como *Hibernate* para llevar a cabo el mapeo de una base de datos.
* ¿Qué problemas nos encontramos al realizar la batería de pruebas contra la base de datos? ¿Se te ocurre alguna forma de solucionarlo?