

Práctica 6 HLC

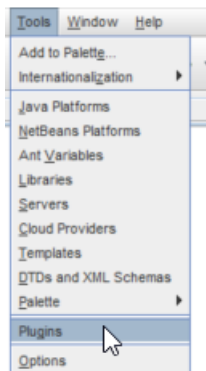
HLC

Configuración de Hibernate

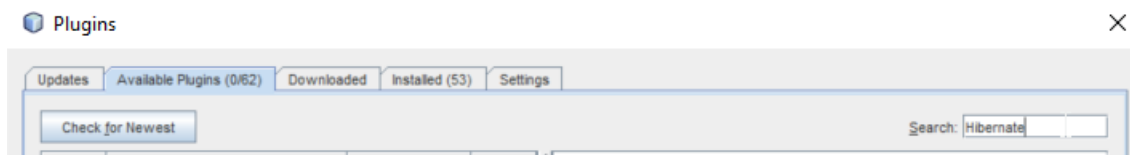
Preparación

Antes de configurar Hibernate, tenemos que tener los plugins necesarios. Para ello es necesario tener instalado [NetBeans 8.2 con JDK 8u111](#) (recomendado usar torrent, es más rápido que la descarga normal).

Una vez dentro de NetBeans, vamos a ir al menú de los plugins:



Y en la pestaña Available Plugins, vamos a buscar Hibernate.



Si no aparece buscando, es probable que ya esté instalado, y haya que actualizarlo en la pestaña Updates, en cuyo caso lo actualizamos.

Creación de un proyecto Java Web con Hibernate

Configuración de la Base de Datos

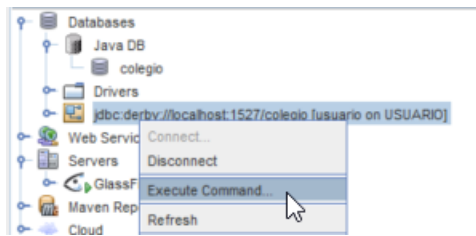
Primero tenemos que crear la base de datos.

La base de datos es Derby, y el archivo para la inserción de datos se encuentra en `src/java/Data`.

En Services, crearemos una nueva base de datos con las siguientes credenciales:

- Nombre: colegio
- Usuario: usuario
- Contraseña: usuario

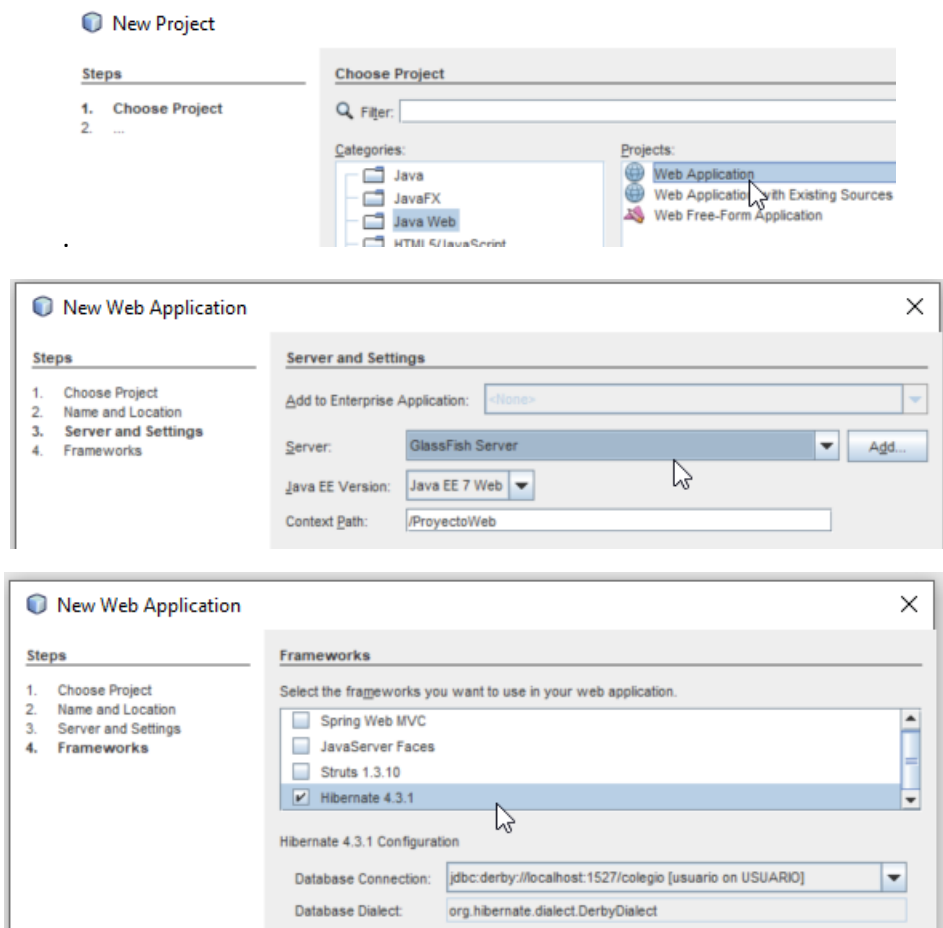
Nos conectamos a la BD usando las credenciales. Una vez conectados, seleccionamos el driver y hacemos clic en Execute Command. Se nos abrirá una pestaña en el editor, ahí podremos ejecutar el código SQL de src/java/Data/dbase.sql.



Creación y configuración del proyecto

Creamos un nuevo proyecto Java Web > Web Application.

En el Wizard para la creación del proyecto, debemos comprobar que como server se usará GlassFish Server, y en los frameworks hemos elegido Hibernate 4.3.1

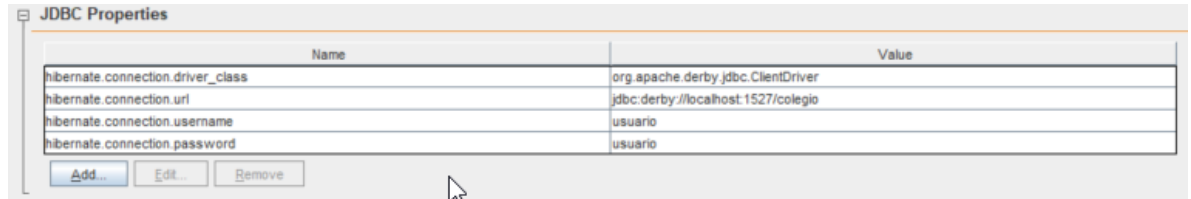


Configuración de archivos para Hibernate

Hay que crear tres paquetes: Dato, Persistence, y Model. El archivo hibernate.cfg.xml lo moveremos a Persistence.

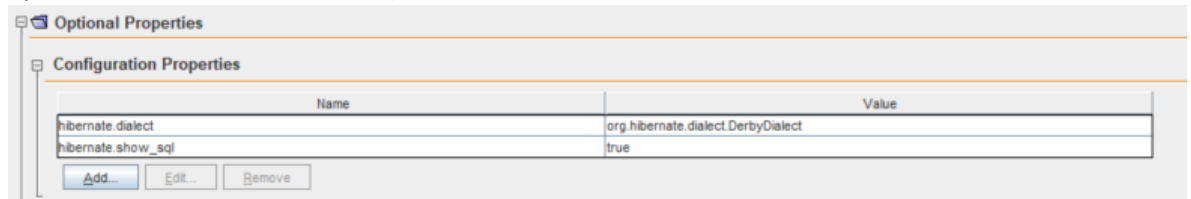
Ahora vamos a editar ese archivo, así que lo abriremos. Hay que realizar las siguientes comprobaciones:

- En JDBC properties deben estar estos campos (si no aparecieran, se añaden con Add):



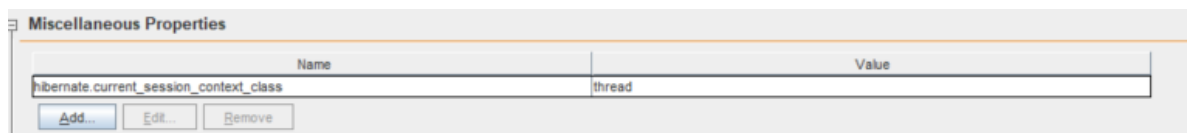
Name	Value
hibernate.connection.driver_class	org.apache.derby.jdbc.ClientDriver
hibernate.connection.url	jdbc:derby://localhost:1527/colegio
hibernate.connection.username	usuario
hibernate.connection.password	usuario

- En Optional Properties > Configuration Properties deben aparecer estos campos (si no aparecieran, se añaden con Add):



Name	Value
hibernate.dialect	org.hibernate.dialect.DerbyDialect
hibernate.show_sql	true

- En Optional Properties > Miscellaneous Properties debe aparecer este campo, si no, se añade:



Name	Value
hibernate.current_session_context_class	thread

Ahora vamos a crear los siguientes archivos en Persistence:

- New > Other > Filtramos buscando Hibernate util > Lo creamos sin cambiar nada. Una vez creado vamos a modificar la línea en la que se asigna sessionFactory, añadiendo como parámetro al método .configure() la siguiente cadena "Persistence/hibernate.cfg.xml".
- New > Other > Filtramos buscando Hibernate Reverse Engineernig Wizard. Cuando estemos configurando este nuevo archivo, haremos clic en Add All cuando aparezca este boton.
- En el paquete Model haremos lo siguiente: clic derecho > New > Other > Buscamos Hibernate Mapping Files and POJOs from Database, y en el Wizard para la configuración, comprobaremos que JDK 5 Language Features está marcado, y lo creamos tal cual.

Por último, creamos las clases Dao, que son las que se usarán para interactuar con la BD. Crearemos una interfaz por cada tabla de la BD, y una clase para implementar cada interfaz.