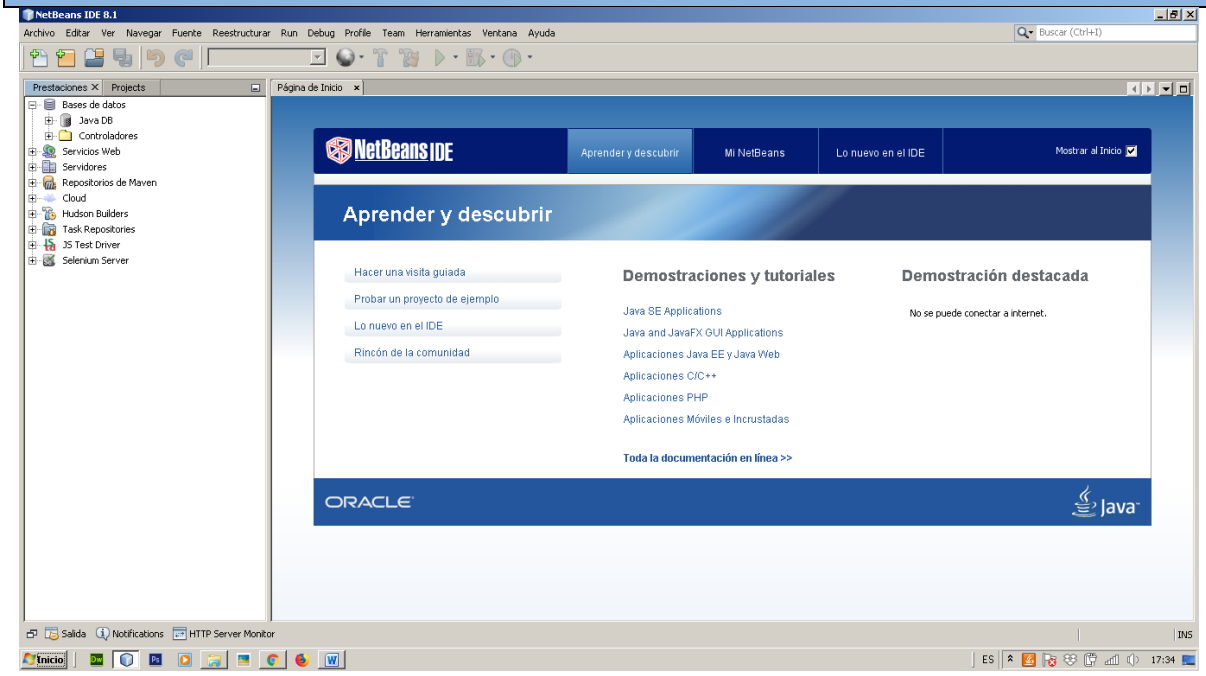
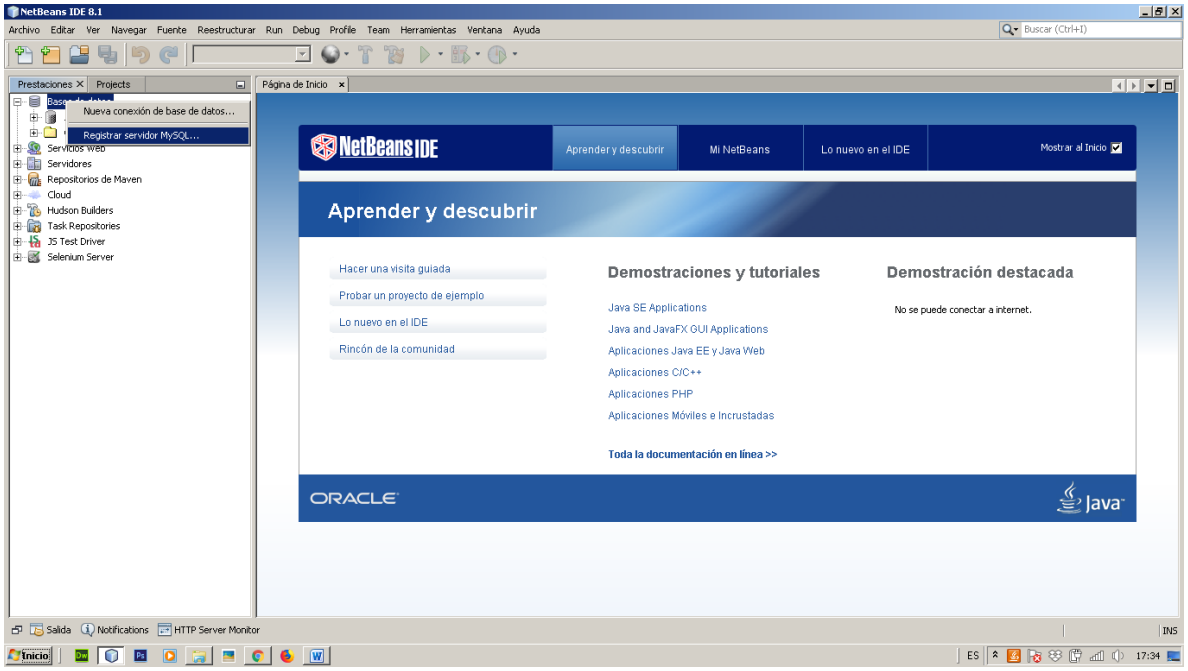
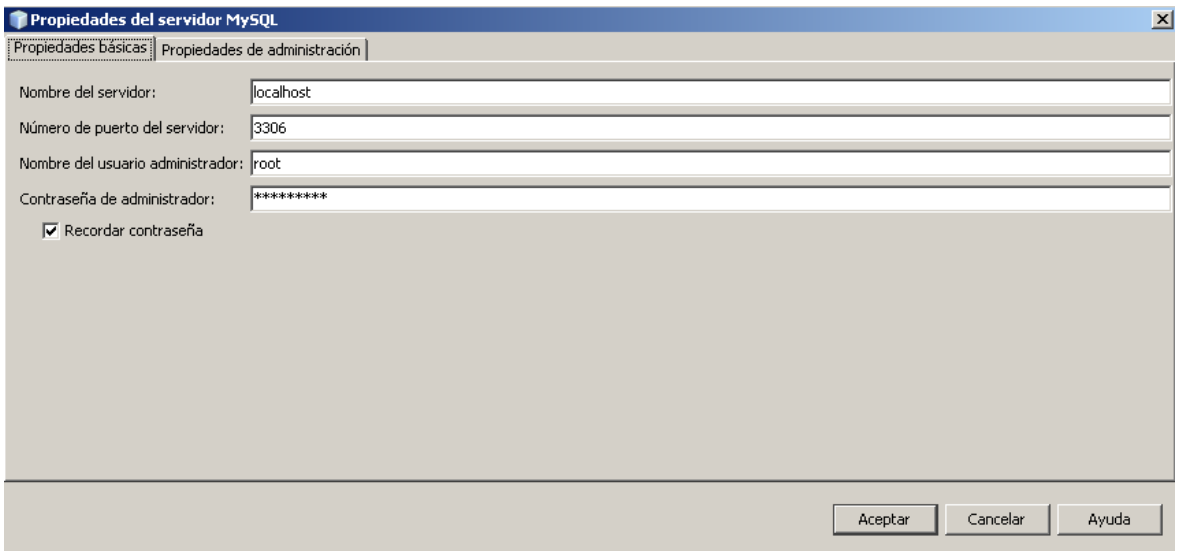
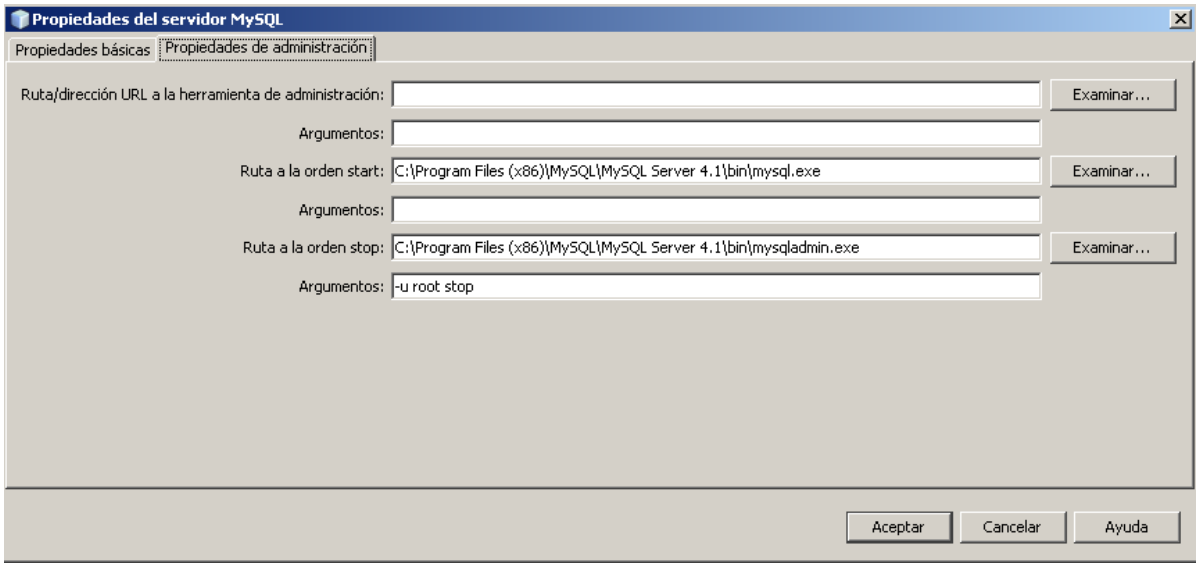
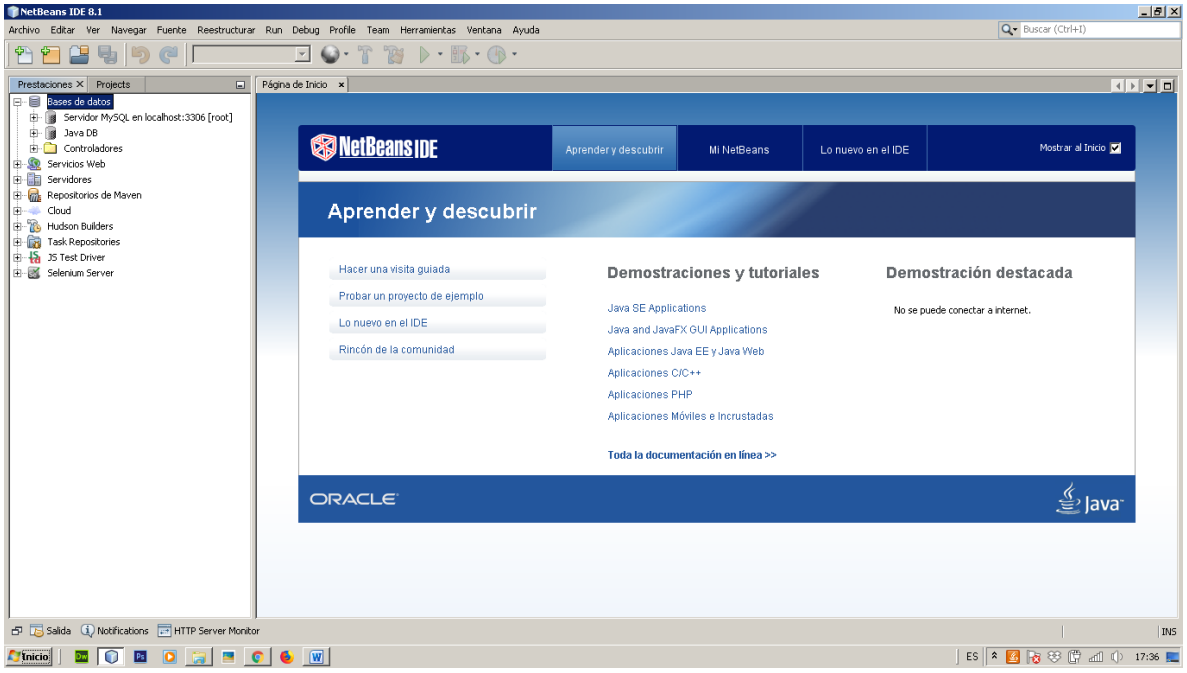
# Crear la conexión Mysql con Netsben

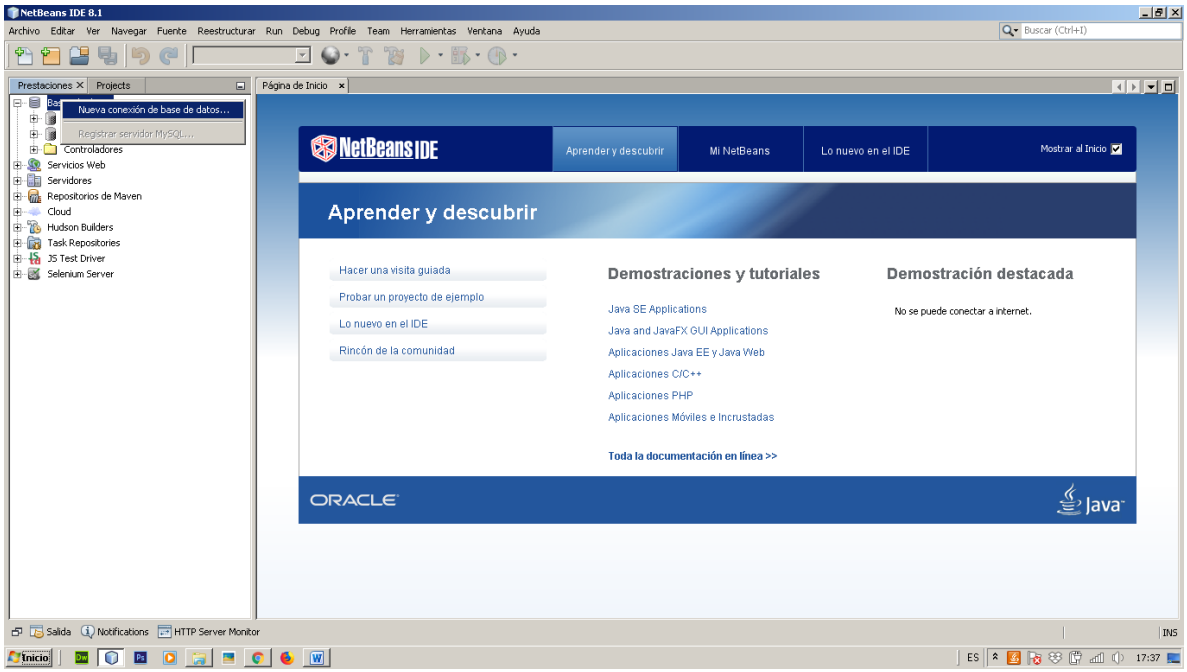


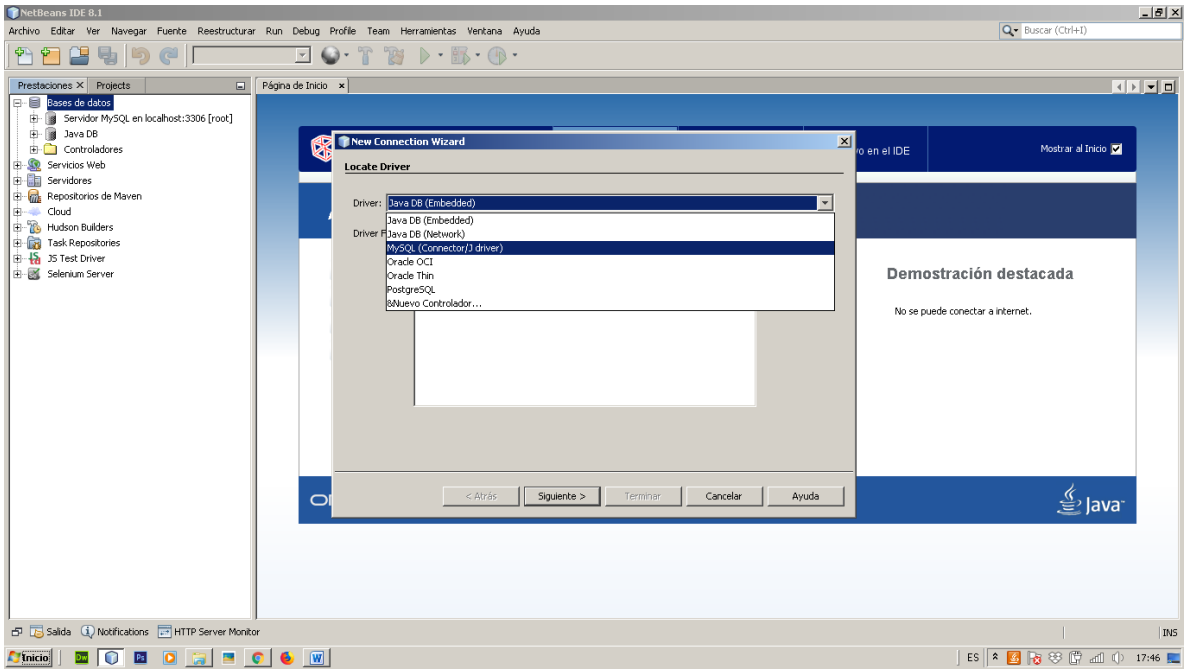



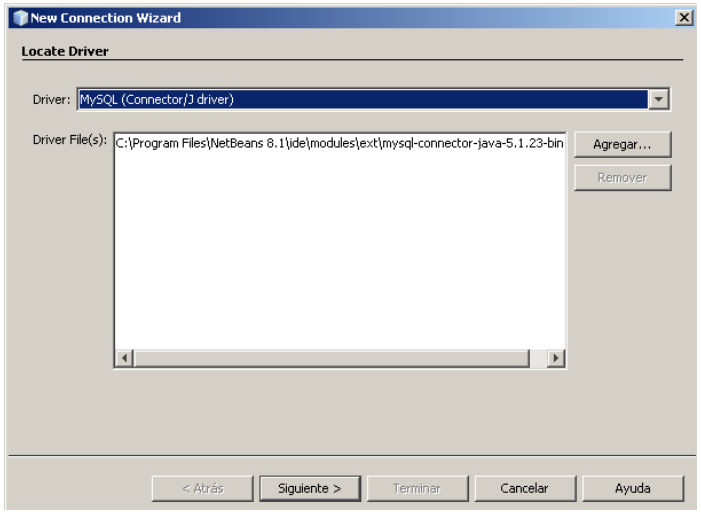



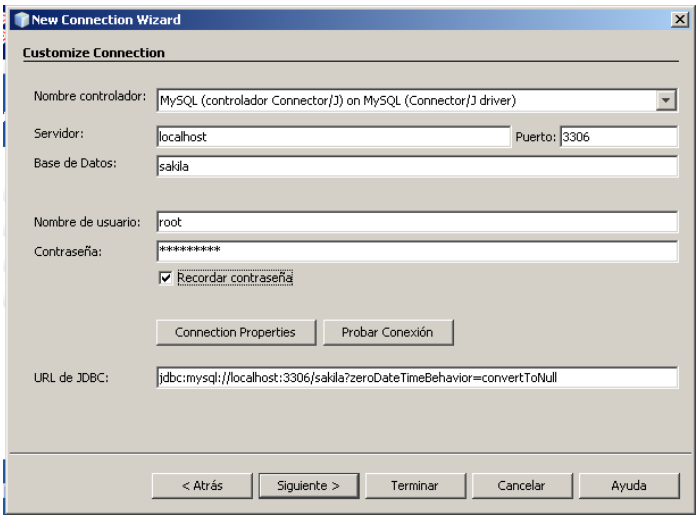

Propiedades del servidor MySQL
Propiedades básicas
Propiedades de administración
Ruta/dirección URL a la herramienta de administración:
Examinar...
Argumentos:
Ruta a la orden start:
C:\Program Files (x86)\MySQL\MySQL Server 4.1\bin\mysql.exe
Examinar...
Argumentos:
Ruta a la orden stop:
C:\Program Files (x86)\MySQL\MySQL Server 4.1\bin\mysqladmin.exe
Examinar...
Argumentos:
-u root stop
Aceptar
Cancelar
Ayuda

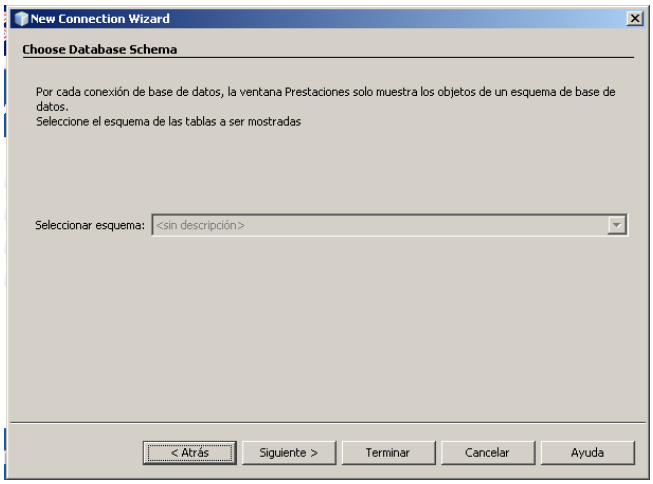

NetBeans IDE 8.1
Archivo Editar Ver Navegar Fuente Reestructurar Run Debug Profile Team Herramientas Ventana Ayuda
Buscar (Ctrl+I)
Prestaciones
Projects
Página de Inicio
Bases de datos
Servidor MySQL en localhost:3306 [root]
Java DB
Controladores
Servicios Web
Servidores
Repositorios de Maven
Cloud
Hudson Builders
Task Repositories
JS Test Driver
Selenium Server
NetBeans IDE
Aprender y descubrir
Mi NetBeans
Lo nuevo en el IDE
Mostrar al Inicio
Aprender y descubrir
Hacer una visita guiada
Probar un proyecto de ejemplo
Lo nuevo en el IDE
Rincón de la comunidad
Demostraciones y tutoriales
Java SE Applications
Java and JavaFX GUI Applications
Aplicaciones Java EE y Java Web
Aplicaciones C/C++
Aplicaciones PHP
Aplicaciones Móviles e Incrustadas
Toda la documentación en línea >>
Demostración destacada
No se puede conectar a internet.
ORACLE
Java
Salida
Notifications
HTTP Server Monitor
INS
Inicio
ES
17:36

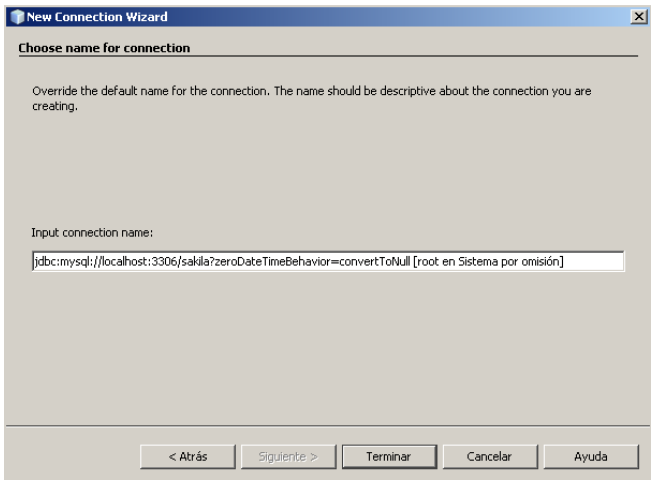

NetBeans IDE 8.1
Archivo Editar Ver Navegar Fuente Reestructurar Run Debug Profile Team Herramientas Ventana Ayuda
Buscar (Ctrl+I)
Prestaciones
Projects
Página de Inicio
Nueva conexión de base de datos...
Registrar servidor MySQL...
Controladores
Servicios Web
Servidores
Repositorios de Maven
Cloud
Hudson Builders
Task Repositories
JS Test Driver
Selenium Server
NetBeans IDE
Aprender y descubrir
Mi NetBeans
Lo nuevo en el IDE
Mostrar al Inicio
Aprender y descubrir
Hacer una visita guiada
Probar un proyecto de ejemplo
Lo nuevo en el IDE
Rincón de la comunidad
Demostraciones y tutoriales
Java SE Applications
Java and JavaFX GUI Applications
Aplicaciones Java EE y Java Web
Aplicaciones C/C++
Aplicaciones PHP
Aplicaciones Móviles e Incrustadas
Toda la documentación en línea >>
Demostración destacada
No se puede conectar a internet.
ORACLE
Java
Salida
Notifications
HTTP Server Monitor
INS
Inicio
ES
17:37

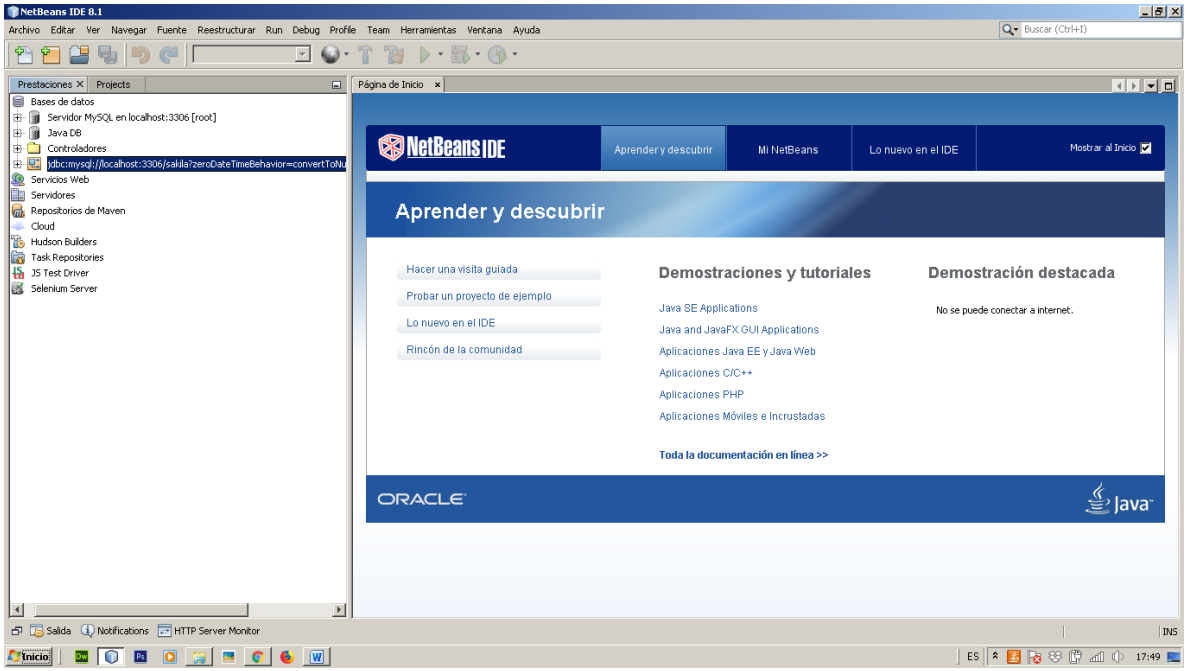

NetBeans IDE 8.1
Archivo Editar Ver Navegar Fuente Reestructurar Run Debug Profile Team Herramientas Ventana Ayuda
Prestaciones
Projects
Página de Inicio
Bases de datos
Servidor MySQL en localhost:3306 [root]
Java DB
Controladores
Servicios Web
Servidores
Repositorios de Maven
Cloud
Hudson Builders
Task Repositories
JS Test Driver
Selenium Server
New Connection Wizard
Locate Driver
Driver:
Java DB (Embedded)
Java DB (Network)
MySQL (Connector/J driver)
Oracle OCI
Oracle Thin
PostgreSQL
&Nuevo Controlador...
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda
Mostrar al Inicio
Demostración destacada
No se puede conectar a internet.
Java
Salida
Notifications
HTTP Server Monitor
INS
Inicio
ES
17:46


New Connection Wizard
Locate Driver
Driver: MySQL (Connector/J driver)
Driver File(s): C:\Program Files\NetBeans 8.1\ide\modules\ext\mysql-connector-java-5.1.23-bin
Agregar...
Remover
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda


New Connection Wizard
Customize Connection
Nombre controlador: MySQL (controlador Connector/J) on MySQL (Connector/J driver)
Servidor: localhost
Puerto: 3306
Base de Datos: sakila
Nombre de usuario: root
Contraseña: *********
Recordar contraseña
Connection Properties
Probar Conexión
URL de JDBC: jdbc:mysql://localhost:3306/sakila?zeroDateTimeBehavior=convertToNull
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda


New Connection Wizard
Choose Database Schema
Por cada conexión de base de datos, la ventana Prestaciones solo muestra los objetos de un esquema de base de datos.
Seleccione el esquema de las tablas a ser mostradas
Seleccionar esquema: <sin descripción>
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda

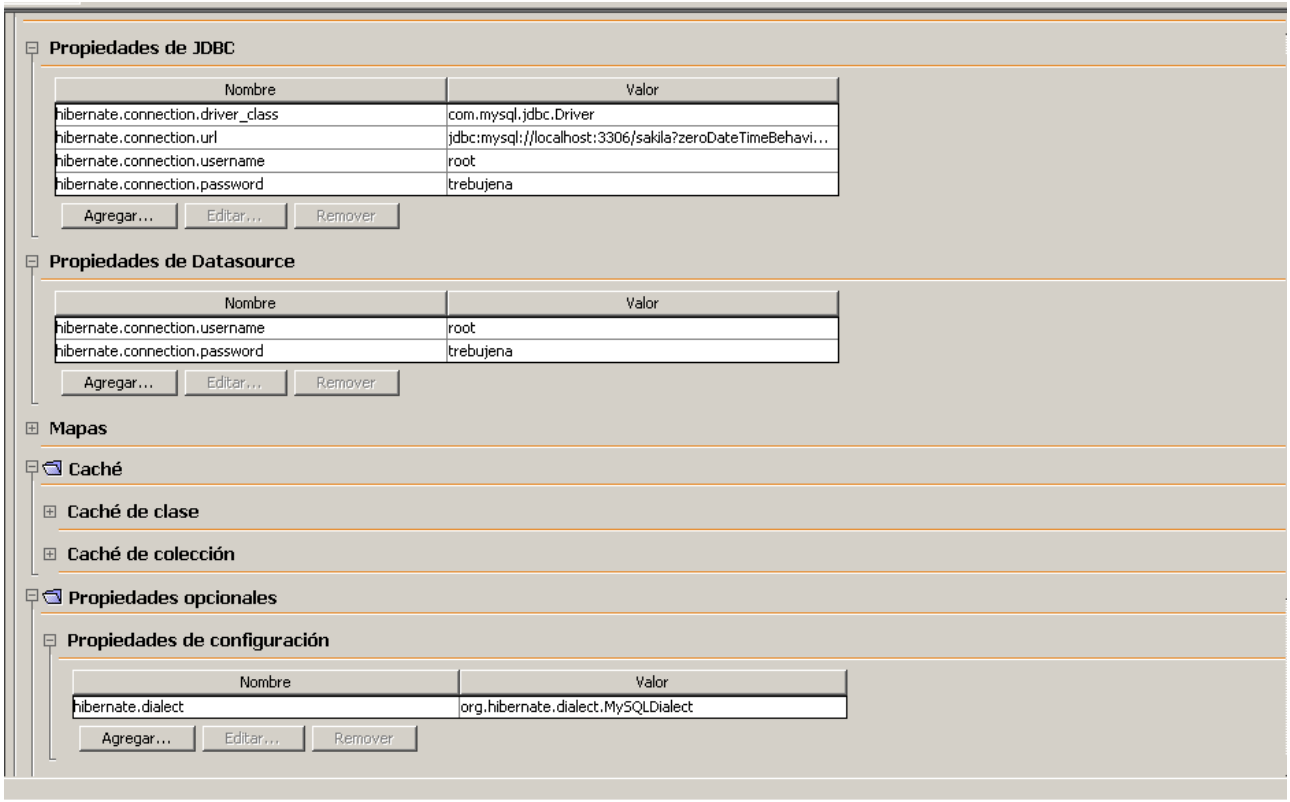

New Connection Wizard
Choose name for connection
Override the default name for the connection. The name should be descriptive about the connection you are creating.
Input connection name:
jdbc:mysql://localhost:3306/sakila?zeroDateTimeBehavior=convertToNull [root en Sistema por omisión]
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda


NetBeans IDE 8.1
Archivo Editar Ver Navegar Fuente Reestructurar Run Debug Profile Team Herramientas Ventana Ayuda
Buscar (Ctrl+I)
Prestaciones
Projects
Bases de datos
Servidor MySQL en localhost:3306 [root]
Java DB
Controladores
jdbc:mysql://localhost:3306/sakila?zeroDateTimeBehavior=convertToNu
Servicios Web
Servidores
Repositorios de Maven
Cloud
Hudson Builders
Task Repositories
JS Test Driver
Selenium Server
Página de Inicio
NetBeans IDE
Aprender y descubrir
Mi NetBeans
Lo nuevo en el IDE
Mostrar al Inicio
Aprender y descubrir
Hacer una visita guiada
Probar un proyecto de ejemplo
Lo nuevo en el IDE
Rincón de la comunidad
Demostraciones y tutoriales
Java SE Applications
Java and JavaFX GUI Applications
Aplicaciones Java EE y Java Web
Aplicaciones C/C++
Aplicaciones PHP
Aplicaciones Móviles e Incrustadas
Toda la documentación en línea >>
Demostración destacada
No se puede conectar a internet.
ORACLE
Java
Salida
Notifications
HTTP Server Monitor
INS
Inicio
ES
17:49

## Crear el proyecto de aplicación web

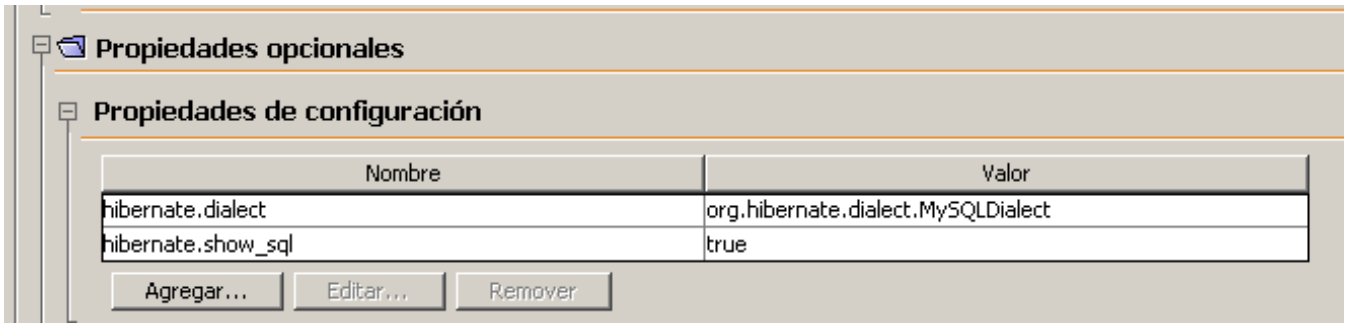
## Modificar el archivo de configuración de Hibernate hibernate.cfg.xml

Este archivo de configuración contiene información sobre la conexión de la base de datos.

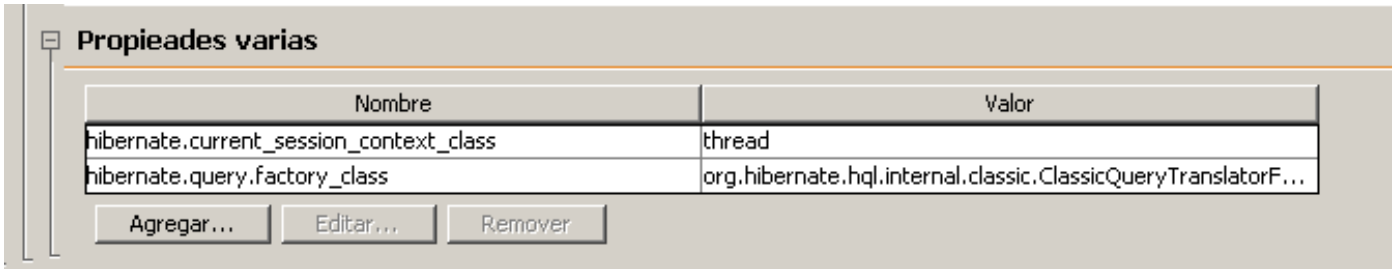



Debemos editar las Propiedades predeterminadas especificadas en hibernate.cfg.xml para habilitar el registro de depuración para declaraciones de SQL y para habilitar la administración del contexto de la sesión de Hibernate.

- seleccione la propiedad ***hibernate.show_sql*** y establecer el valor en ***verdadero***. Esto habilita el registro de depuración de las sentencias de SQL.




- Debemos expandir el nodo Propiedades varias y haga clic en Agregar:
  - ***hibernate.current_context_***class y establecer el valor a ***thread***
  - ***hibernate.query.factory_class*** y establecer el valor a ***org.hibernate.hql.internal.classic.ClassicqueryTranslatorFactory***



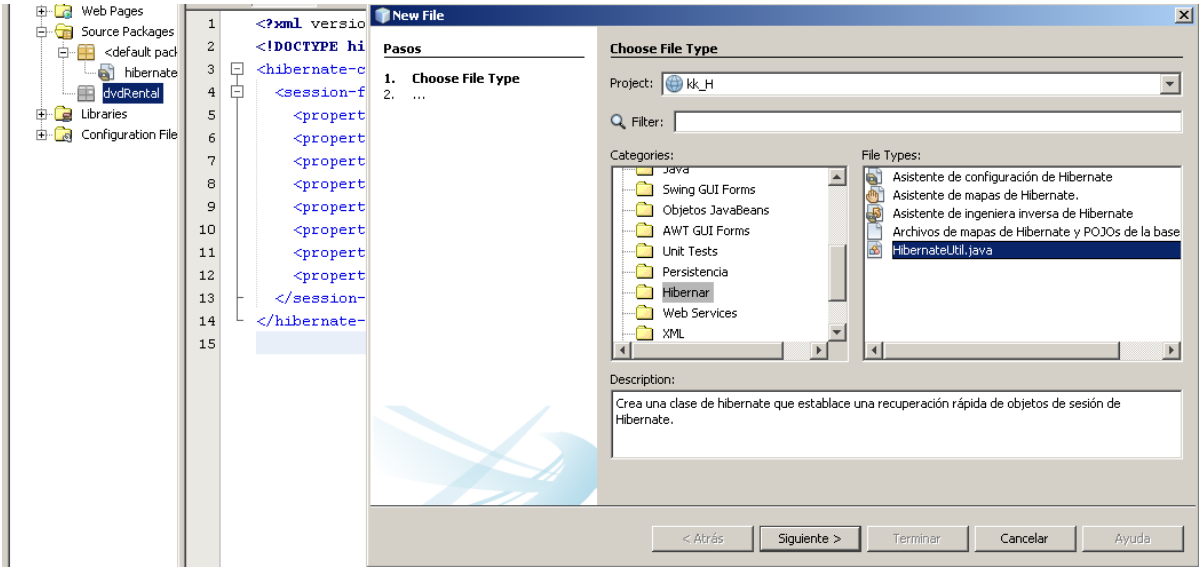
Si hace clic en la pestaña XML en el editor, puede ver el archivo en la vista XML.

```
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
    <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
    <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/sakila?zeroDateTimeBehavior=convertToNull</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password">trebujena</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.current_session_context_class">thread</property>
    <property name="hibernate.query.factory_class">org.hibernate.hql.internal.classic.ClassicQueryTranslatorFactory</property>
  </session-factory>
</hibernate-configuration>
```

## Creación del archivo HibernateUtil.java

Para usar Hibernate, necesita crear una clase auxiliar que maneje el inicio y que acceda a SessionFactory de Hibernate para obtener un objeto Session.

Utilizaremos el asistente para crear la clase auxiliar HibernateUtil.java.

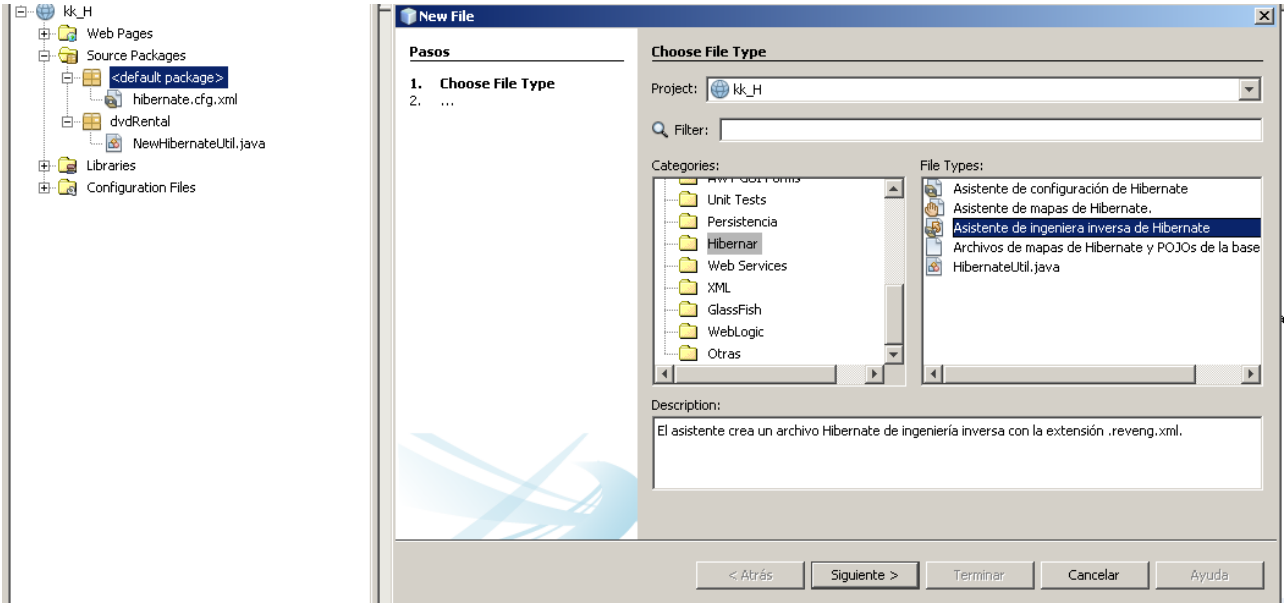


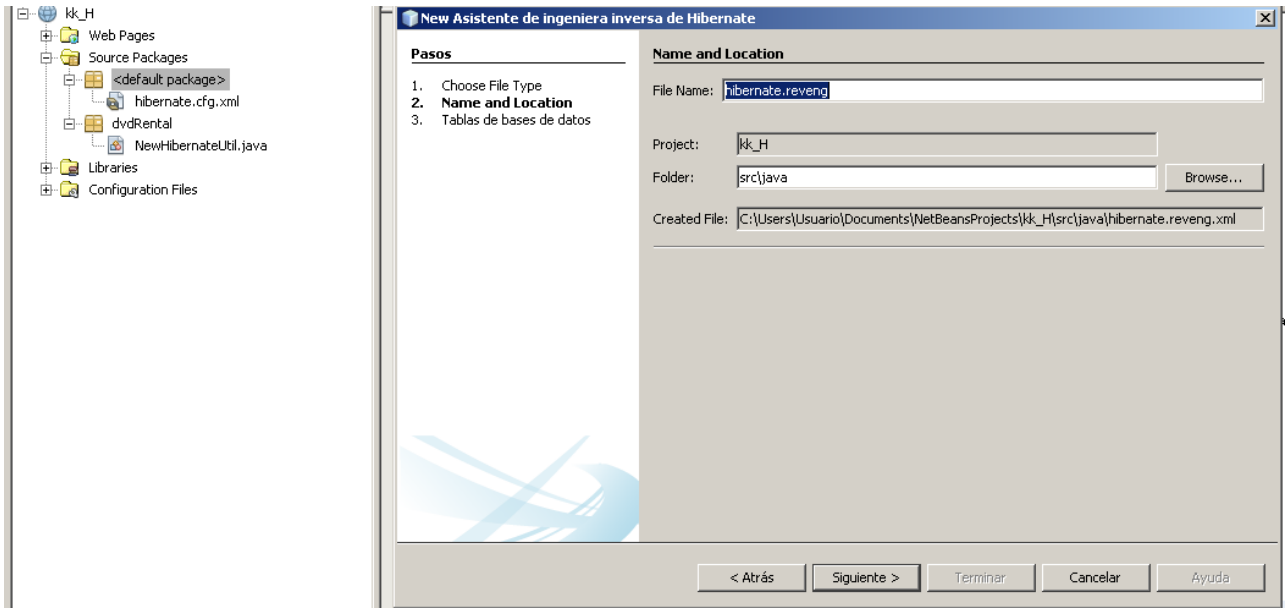
```java
package dvdRental;
import org.hibernate.cfg.AnnotationConfiguration;
import org.hibernate.SessionFactory;

public class NewHibernateUtil {

    private static final SessionFactory sessionFactory;

    static {
        try {
            // Create the SessionFactory from standard (hibernate.cfg.xml)
            // config file.
            sessionFactory = new AnnotationConfiguration().configure().buildSessionFactory();
        } catch (Throwable ex) {
            // Log the exception.
            System.err.println("Initial SessionFactory creation failed." + ex);
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }
}
```

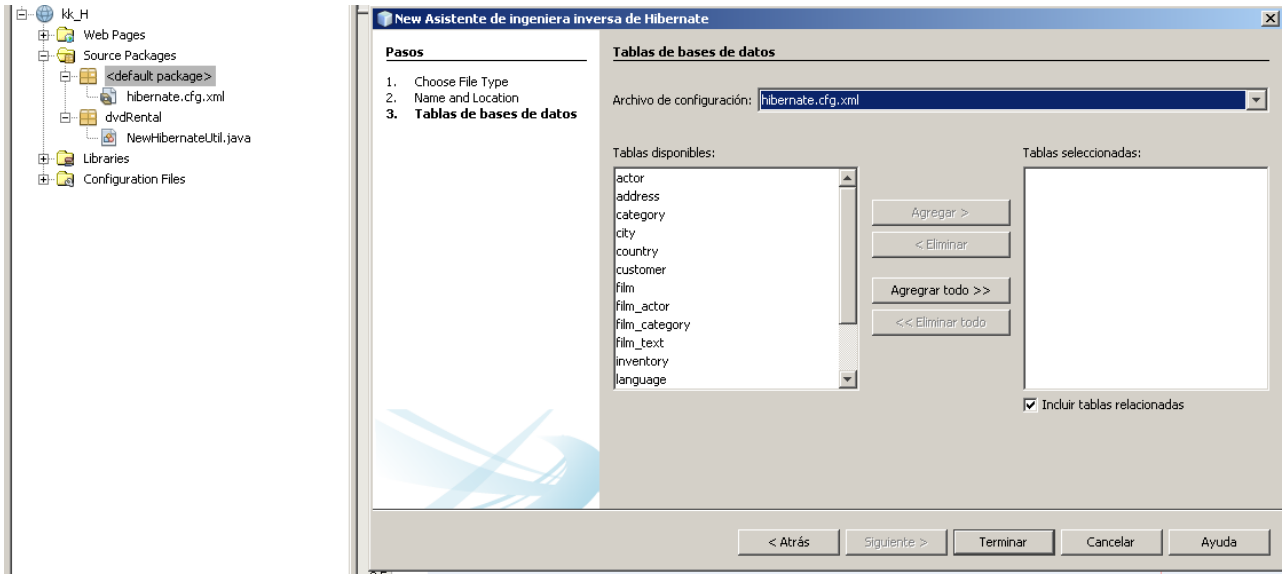
## Generación de archivos de mapeo de Hibernate y las clases Java

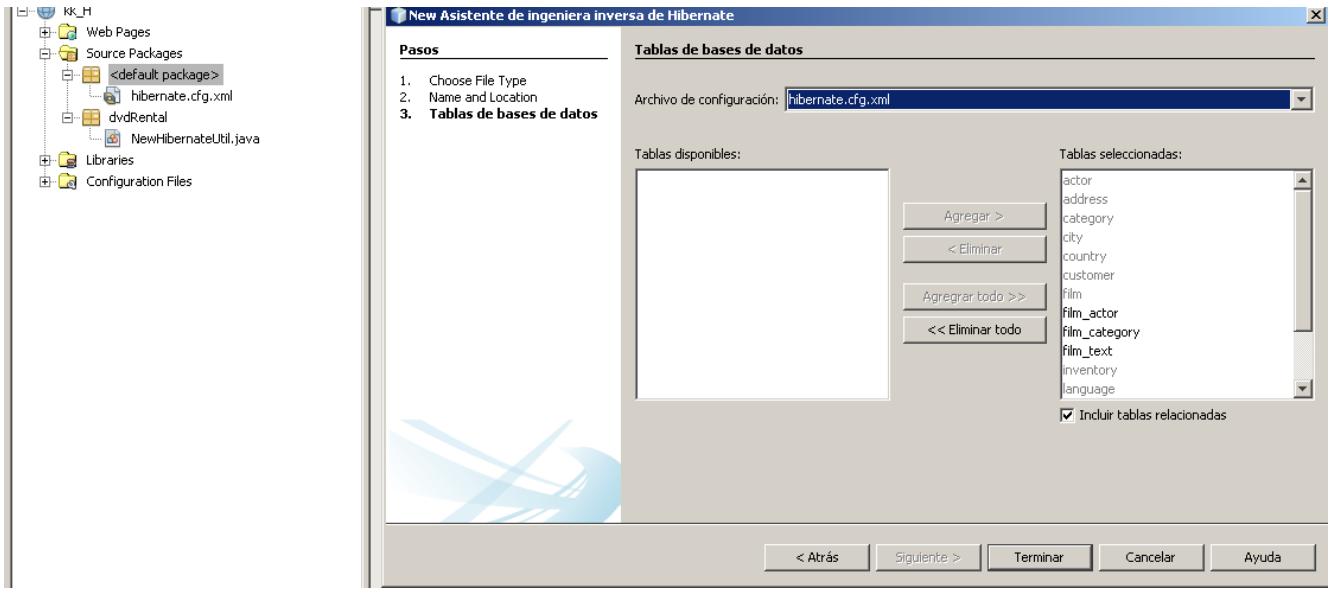
El archivo de ingeniería inversa permite tener un mayor control sobre la estrategia de mapeo de la base de datos.

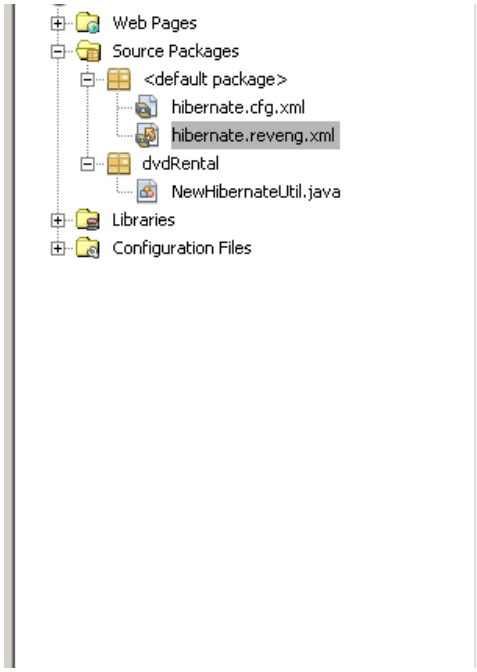
El asistente de ingeniería inversa de Hibernate crea un archivo de ingeniería inversa con una configuración que puede editar en el editor XML.




kk_H
Web Pages
Source Packages
<default package>
hibernate.cfg.xml
dvdRental
NewHibernateUtil.java
Libraries
Configuration Files
New Asistente de ingeniera inversa de Hibernate
Pasos
1. Choose File Type
2. Name and Location
3. Tablas de bases de datos
Name and Location
File Name: hibernate.reveng
Project: kk_H
Folder: src\java
Browse...
Created File: C:\Users\Usuario\Documents\NetBeansProjects\kk_H\src\java\hibernate.reveng.xml
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda


kk_H
Web Pages
Source Packages
<default package>
hibernate.cfg.xml
dvdRental
NewHibernateUtil.java
Libraries
Configuration Files
New Asistente de ingeniera inversa de Hibernate
Pasos
1. Choose File Type
2. Name and Location
3. Tablas de bases de datos
Tablas de bases de datos
Archivo de configuración: hibernate.cfg.xml
Tablas disponibles:
actor
address
category
city
country
customer
film
film_actor
film_category
film_text
inventory
language
Agregar >
< Eliminar
Agregrar todo >>
<< Eliminar todo
Tablas seleccionadas:
Incluir tablas relacionadas
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda


kk_H
Web Pages
Source Packages
<default package>
hibernate.cfg.xml
dvdRental
NewHibernateUtil.java
Libraries
Configuration Files
New Asistente de ingeniera inversa de Hibernate
Pasos
1. Choose File Type
2. Name and Location
3. Tablas de bases de datos
Tablas de bases de datos
Archivo de configuración: hibernate.cfg.xml
Tablas disponibles:
Agregar >
< Eliminar
Agregrar todo >>
<< Eliminar todo
Tablas seleccionadas:
actor
address
category
city
country
customer
film
film_actor
film_category
film_text
inventory
language
Incluir tablas relacionadas
< Atrás
Siguiente >
Terminar
Cancelar
Ayuda



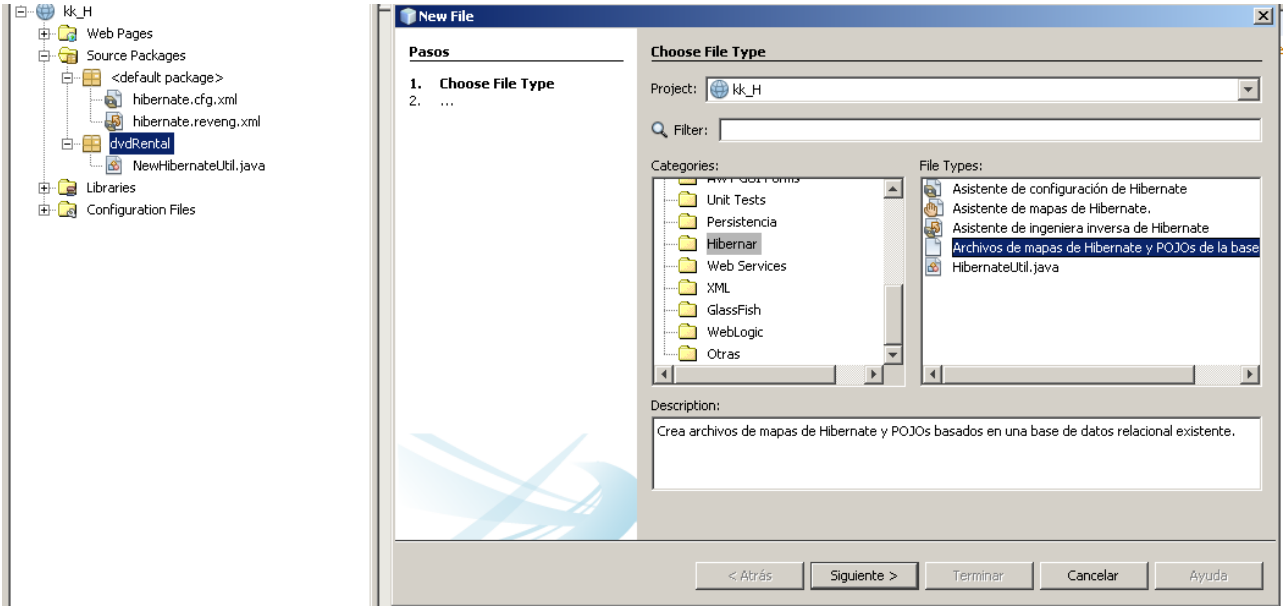

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-reverse-engineering PUBLIC "-//Hibe
<hibernate-reverse-engineering>
  <schema-selection match-catalog="sakila"/>
  <table-filter match-name="store"/>
  <table-filter match-name="film_text"/>
  <table-filter match-name="film_category"/>
  <table-filter match-name="language"/>
  <table-filter match-name="country"/>
  <table-filter match-name="staff"/>
  <table-filter match-name="customer"/>
  <table-filter match-name="rental"/>
  <table-filter match-name="city"/>
  <table-filter match-name="actor"/>
  <table-filter match-name="payment"/>
  <table-filter match-name="film"/>
  <table-filter match-name="film_actor"/>
  <table-filter match-name="inventory"/>
  <table-filter match-name="category"/>
  <table-filter match-name="address"/>
</hibernate-reverse-engineering>
```

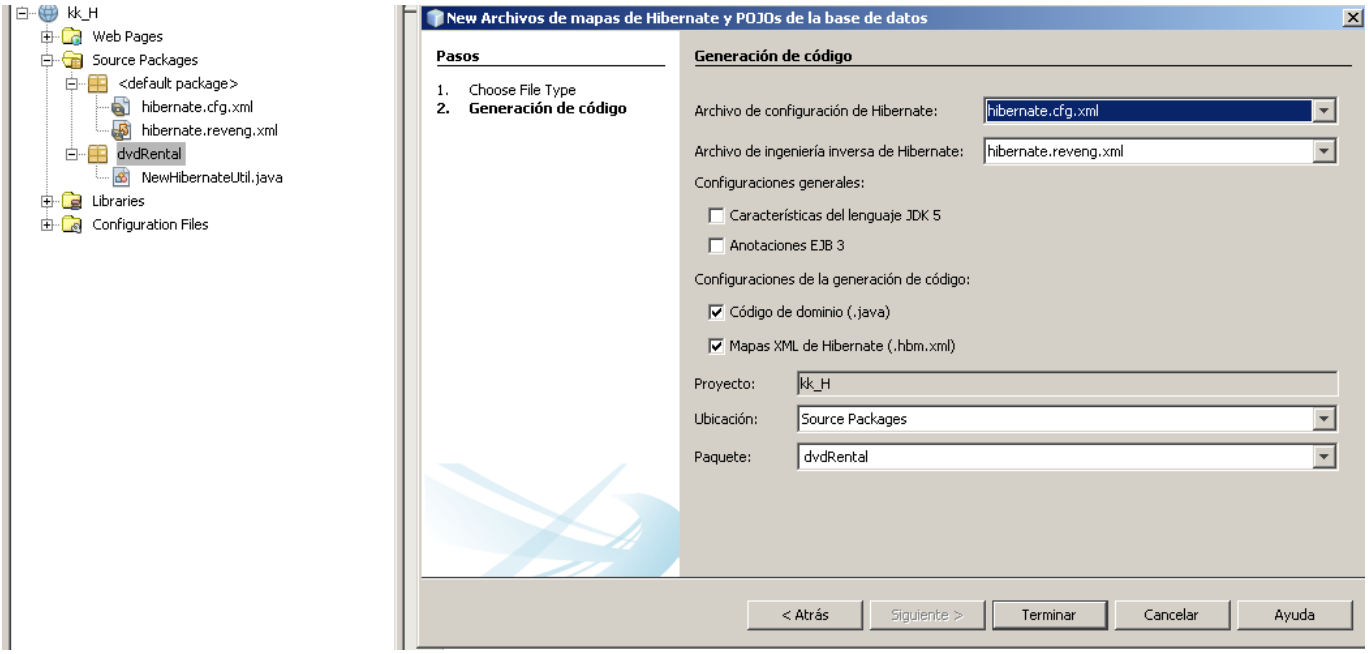
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-reverse-engineering PUBLIC "-//Hibernate/Hibe
<hibernate-reverse-engineering>
  <schema-selection match-catalog="sakila"/>
  <table-filter match-name="language"/>
  <table-filter match-name="rental"/>
  <table-filter match-name="inventory"/>
  <table-filter match-name="customer"/>
  <table-filter match-name="film_category"/>
  <table-filter match-name="actor"/>
  <table-filter match-name="film"/>
  <table-filter match-name="payment"/>
  <table-filter match-name="city"/>
  <table-filter match-name="film_actor"/>
  <table-filter match-name="film_text"/>
  <table-filter match-name="address"/>
  <table-filter match-name="store"/>
  <table-filter match-name="staff"/>
  <table-filter match-name="country"/>
  <table-filter match-name="category"/>
</hibernate-reverse-engineering>
```

## La creación de los archivos de mapeo Hibernate y POJOs

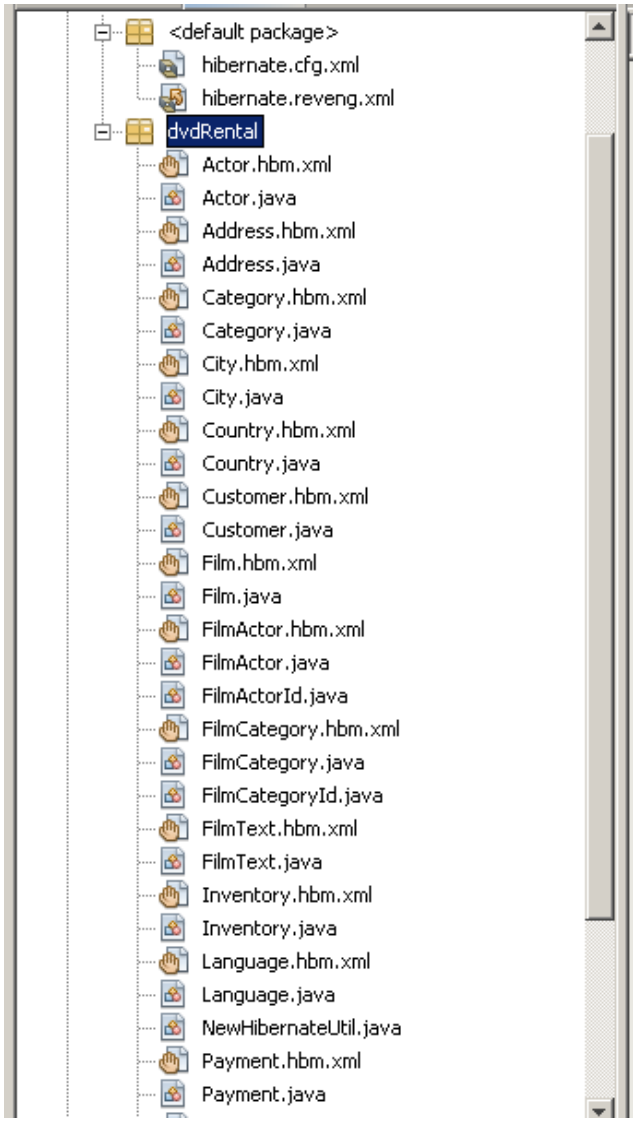
El asistente puede generar un POJO y un archivo de mapeo correspondiente para cada tabla. Los archivos de asignación son archivos XML que contienen datos sobre cómo se asignan las columnas en las tablas a los campos en los POJO.

Debe usar los archivos hibernate.reveng.xml e hibernate.cfg.xml para usar el asistente.




Al expandir el paquete dvdrental podemos  ver los archivos generados por el asistente.




El IDE también agrega entradas de mapeo a hibernate.cfg.xml .

```
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
    <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
    <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/sakila?zeroDateTimeBehavior=convertToNull<
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password">trebujena</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.current_session_context_class">thread</property>
    <property name="hibernate.query.factory_class">org.hibernate.hql.internal.classic.ClassicQueryTranslatorFactory<
    <mapping resource="dvdRental/Actor.hbm.xml"/>
    <mapping resource="dvdRental/FilmText.hbm.xml"/>
    <mapping resource="dvdRental/Store.hbm.xml"/>
    <mapping resource="dvdRental/Payment.hbm.xml"/>
    <mapping resource="dvdRental/City.hbm.xml"/>
    <mapping resource="dvdRental/Country.hbm.xml"/>
    <mapping resource="dvdRental/Address.hbm.xml"/>
    <mapping resource="dvdRental/Language.hbm.xml"/>
    <mapping resource="dvdRental/Customer.hbm.xml"/>
    <mapping resource="dvdRental/Staff.hbm.xml"/>
    <mapping resource="dvdRental/Category.hbm.xml"/>
    <mapping resource="dvdRental/FilmCategory.hbm.xml"/>
    <mapping resource="dvdRental/Film.hbm.xml"/>
    <mapping resource="dvdRental/FilmActor.hbm.xml"/>
    <mapping resource="dvdRental/Inventory.hbm.xml"/>
    <mapping resource="dvdRental/Rental.hbm.xml"/>
  </session-factory>
</hibernate-configuration>
```

# Creación de la clase FilmHelper.java

Ahora creará una clase auxiliar en el paquete dvdrental que se usará para realizar consultas Hibernate en la base de datos. Utilizará el editor Hibernate Query Language (HQL) para construir y probar las consultas para recuperar datos. Después de probar las consultas, creará métodos en la clase auxiliar que construirá y ejecutará las consultas.

```
package dvdRental;

import org.hibernate.Session;
import org.hibernate.SessionFactory;

public class FilmHelper {
    Session session=null;  // creamos la session
    public FilmHelper() {

        // EL objeto SessionFactory se encarga de decir al sistema, donde se encuentran todos los ficheros de mapeo de Hibernate
        SessionFactory sessionFactory = HibernateUtil.getSessionFactory();

        // optenemos la session
        this.session = sessionFactory.openSession();

        //Para trabajar con una base de datos usamos las transacciones.
        org.hibernate.Transaction tx = session.beginTransaction();

    }
}
```

Las interfaces principales de programación que forman parte de Hibernate son los siguientes:

- **org.hibernate.SessionFactory:** se utiliza básicamente para obtener una instancia de Session, y puede ser visto como una analogía con el mecanismo de agrupación de conexiones. Se trata de hilos de seguridad, como todos los hilos de aplicación puede utilizar un SessionFactory único (siempre que Hibernate utilice una sola base de datos). Esta interfaz se configura mediante el archivo de configuración, que determina la asignación de archivos al ser cargado.
- **org.hibernate.Session:** proporciona un único hilo que determina la conversación entre la aplicación y la base de datos. Esto es análogo a una conexión específica. Una instancia de Session es "poco pesada" y su creación y destrucción es muy "barata". Esto es importante, ya que nuestra aplicación necesitará crear y destruir sesiones todo el tiempo, quizá en cada petición. Puede ser útil pensar en una sesión como en una caché o colección de objetos cargados (a o desde una base de datos) relacionados con una única unidad de trabajo.
- **org.hibernate.Transaction:** proporciona un único objeto hilo que se extiende a través de la solicitud y determina una unidad atómica de trabajo. Básicamente, los resúmenes de JDBC.
- **org.hibernate.Query** se utiliza para realizar una consulta, ya sea en HQL o en el dialecto SQL de la base de datos subyacente. Una instancia Query es ligera, y es importante señalar que no se puede utilizar fuera de la sesión a través de la cual se creó.