

## INTRODUCCIÓN A BASE DE DATOS

### Acceso a base de datos, creación, conexión

#### CAPACIDAD EN PROCESO:

- Crea una base de datos con una tabla
- Establece una conexión a la base de datos desde su aplicación

#### Acceso a Base de Datos

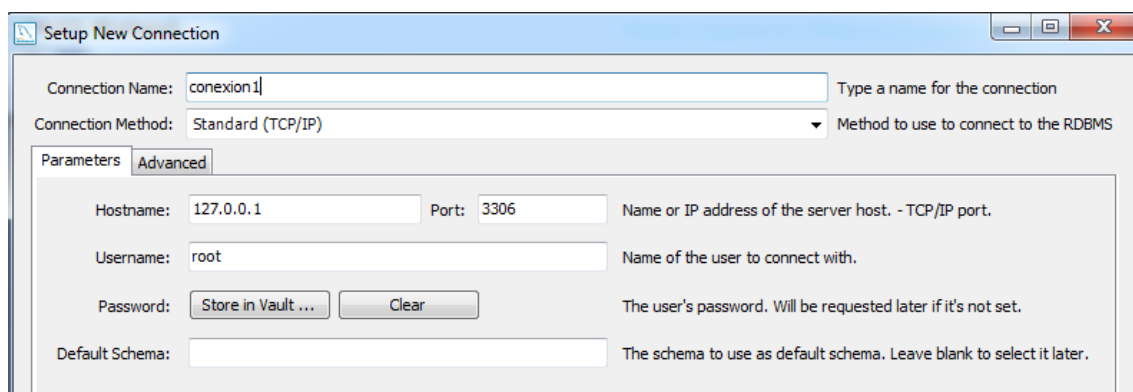
El acceso a una base de datos es siempre un punto infaltable a considerar por nuestras aplicaciones para conservar los datos en el disco. Para esto, en Java tenemos una API muy sencilla que es **Java Database Connectivity** conocido por la abreviación **JDBC**.

Cuando instalamos **JDK** (Java Development Kit) automáticamente se instala esta API (**JDBC**) y con ella en forma muy sencilla podemos hacer consultas, mantenimientos y todo tipo de transacciones a una base de datos.

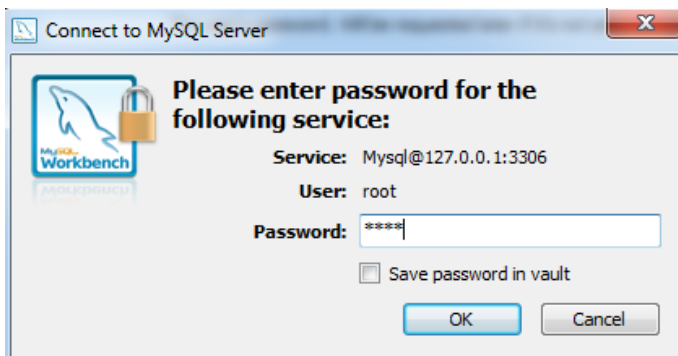
Para desarrollar nuestras primeras aplicaciones nosotros usaremos la forma de conexión a base de datos con el driver de protocolo de java puro para mysql que está disponible en internet para su descarga gratuita (mysql-connector-java-5.1.37) y la librería Mysql JDBC driver incorporado en el entorno de NetBeans.

#### Creación de una Base de Datos

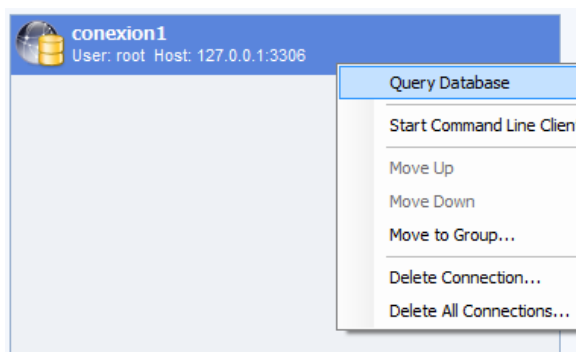
Vamos a crear nuestra base de datos en el gestor de base de datos MySQL WorkBench. Abrimos Workbeanch, elegimos **New Conexion** para llegar a la siguiente ventana:



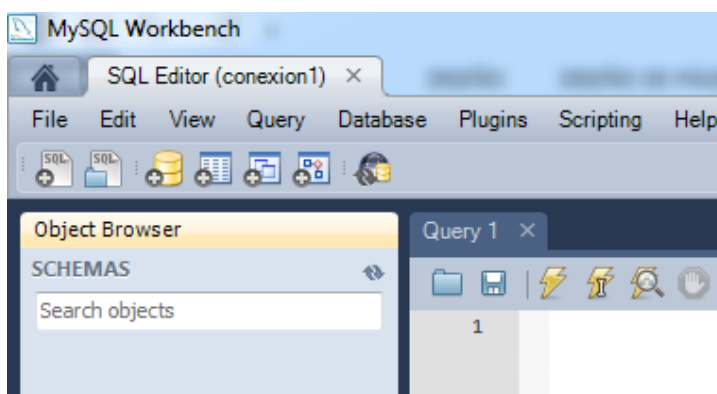
Escribimos el nombre de la nueva conexión (conexion1) y damos clic en el botón Test connection para llegar a la siguiente ventana:



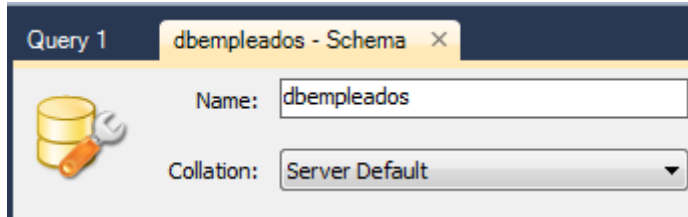
Damos el password (root) y presionamos el botón OK para llegar a la ventana de conexión con éxito y luego damos OK nuevamente para que quede establecida nuestra nueva conexión.



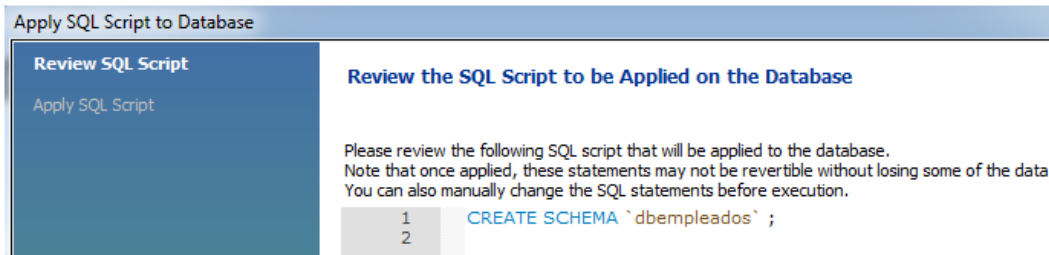
Damos clic derecho en la nueva conexión creada y elegimos Query Database para llegar a la siguiente ventana:



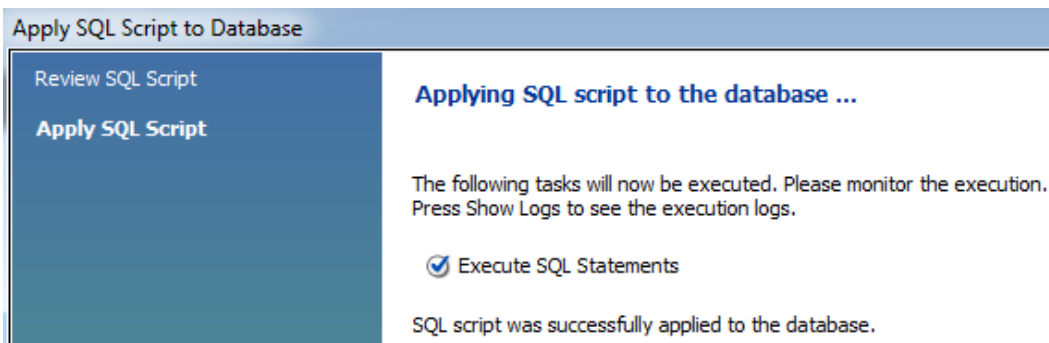
Damos clic en el ícono para crear un nuevo esquema



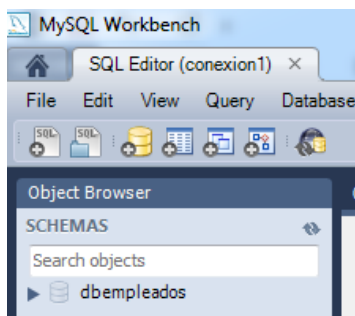
Escribimos el nombre del esquema (dbempleados) y damos clic en el botón Apply para llegar a la siguiente ventana:



Damos clic en el botón Apply nuevamente para llegar a la siguiente ventana:



Damos clic en el botón Finish para llegar a la siguiente ventana:



Donde comprobamos que se ha creado un nuevo esquema (dbempleados) donde daremos clic derecho y elegimos set as default schema.

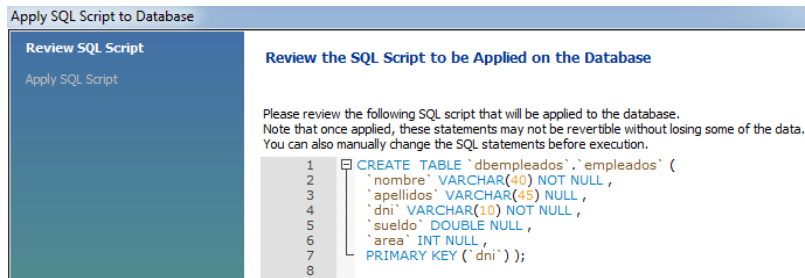
Luego damos clic en el ícono para crear una nueva tabla donde escribimos el nombre y llenamos los campos (column name) conforme a la siguiente ventana:

Query 1 dbempleados - Schema empleados - Table x

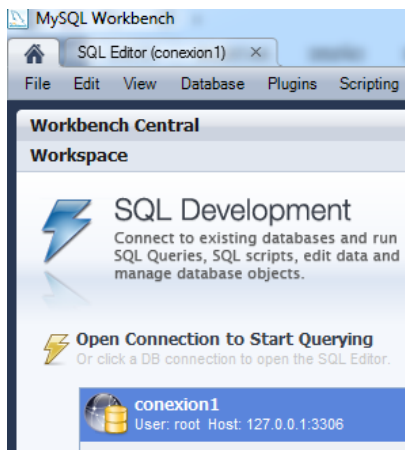
Table Name:  Schema: **dbempleados**

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
nombre	VARCHAR(40)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
apellidos	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
dni	VARCHAR(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
sueldo	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
area	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

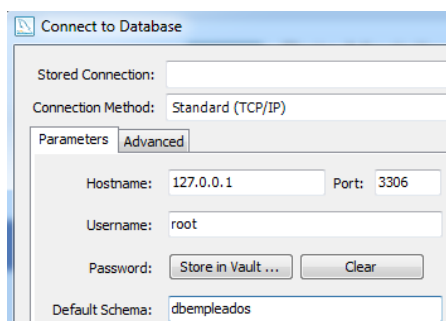
Damos clic en el botón Apply para llegar a la siguiente ventana:



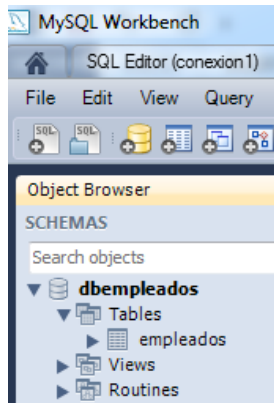
Damos clic en el botón Apply y luego en Finish. Vamos a la ventana de inicio de Workbench donde comprobamos que se ha creado una nueva conexión.



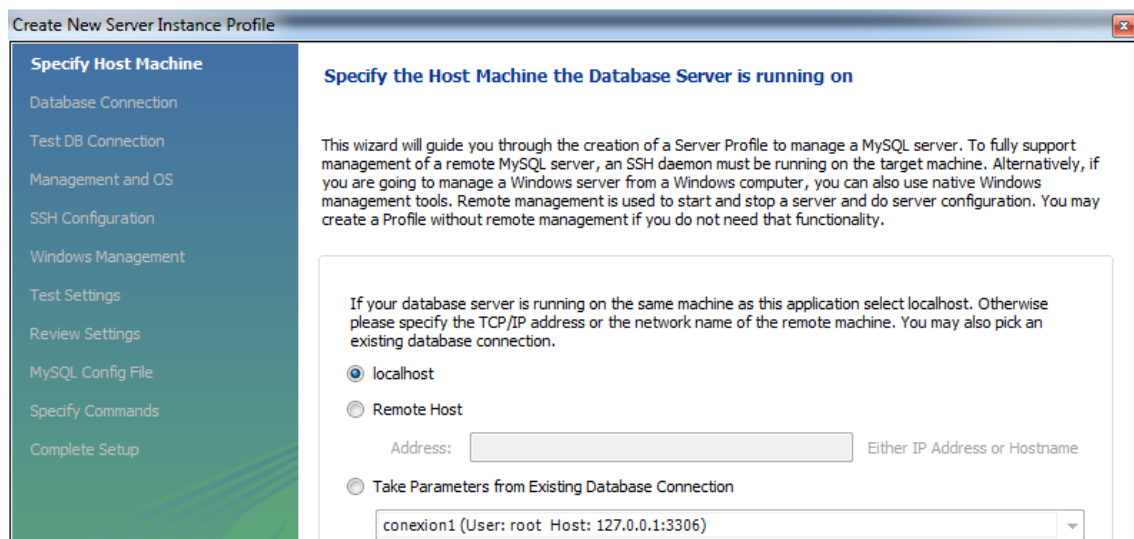
Damos clic en Open Connection to Start Querying para llegar a la siguiente ventana donde escribimos el nombre del esquema (dbempleados):



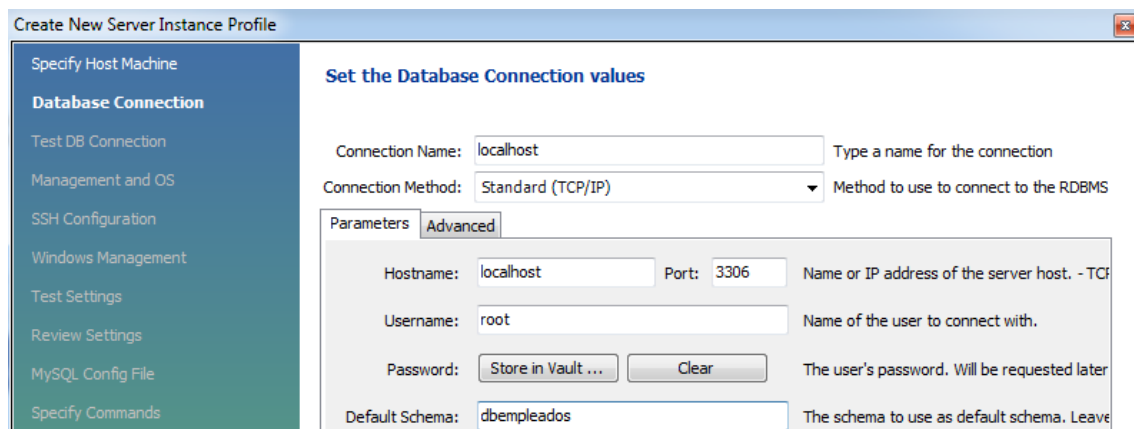
Damos clic en el botón OK, ingresamos el password para llegar a la siguiente ventana:



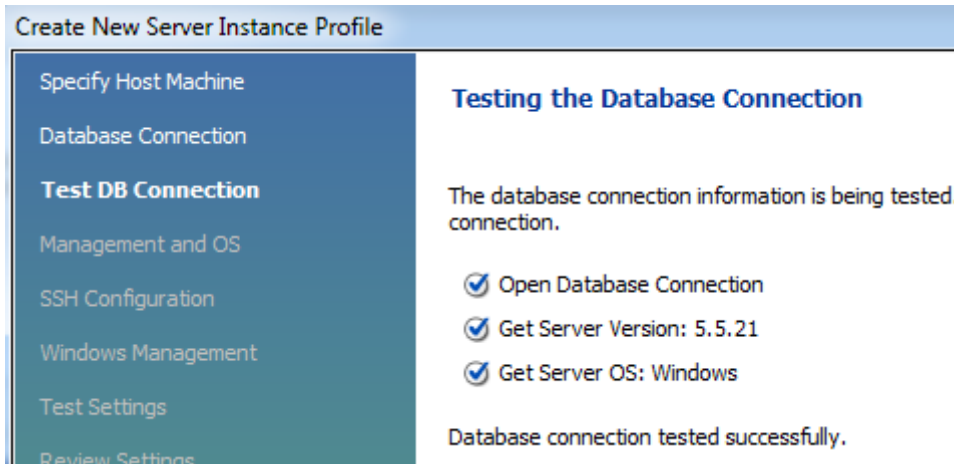
Regresamos al inicio de Workbench para crear una nueva instancia de servidor:



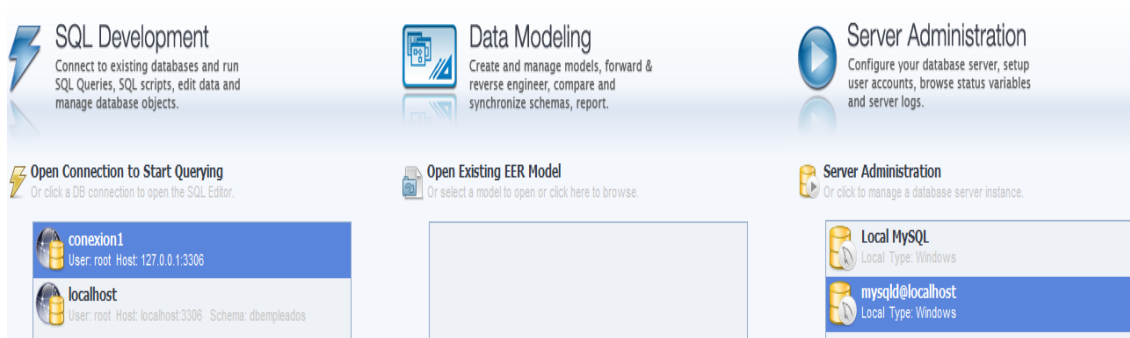
Damos clic en el botón Next y en la siguiente ventana escribimos el nombre del esquema “dbempleados”



Damos clic en el botón Next para llegar a la siguiente ventana:

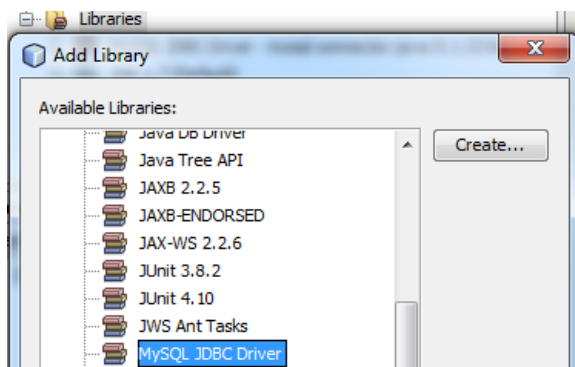


Damos clic en el botón Next hasta finalizar. Vamos al inicio del Workbench para verificar que se ha creado la conexión y la instancia del servidor:



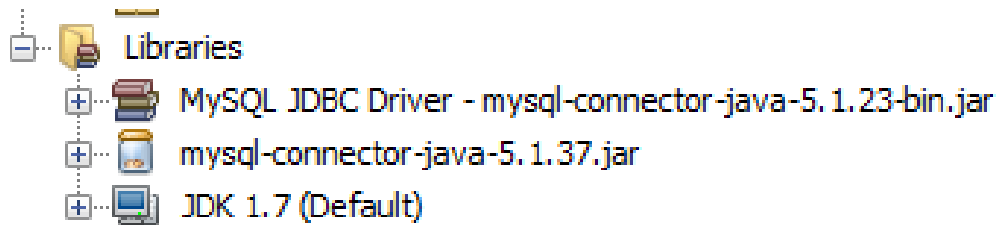
## Conexión a una Base de Datos

Entramos a NetBeans, creamos un proyecto para incorporar el driver descargado y la librería en la carpeta librerías de nuestra aplicación: Clic derecho en Libraries de nuestra aplicación y elegimos Library y seleccionamos MySQL JDBC Driver.

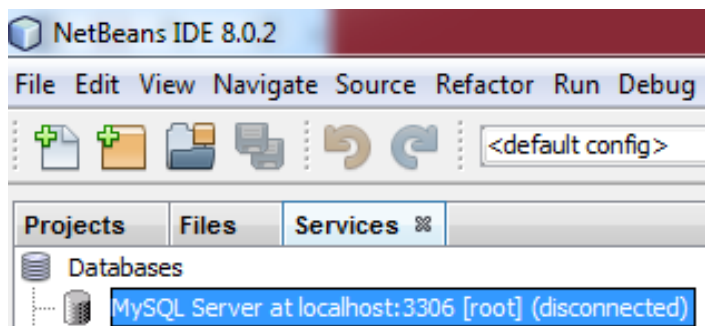


Luego de aceptar, verifique su incorporación en las librerías de nuestra aplicación:

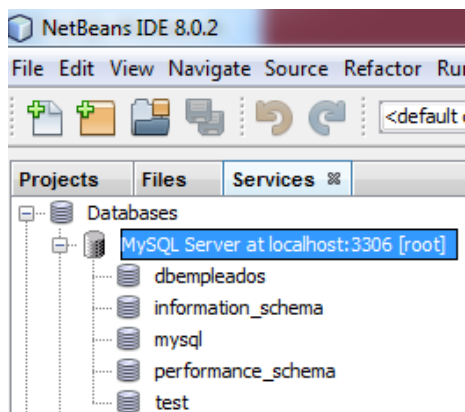
Ahora vamos a incorporar el driver conector que hemos descargado de internet. Nuevamente clic derecho en la carpeta Libraries de nuestra aplicación, elegimos add JAR/folder para que finalmente quede así la carpeta Libraries de nuestra aplicación:



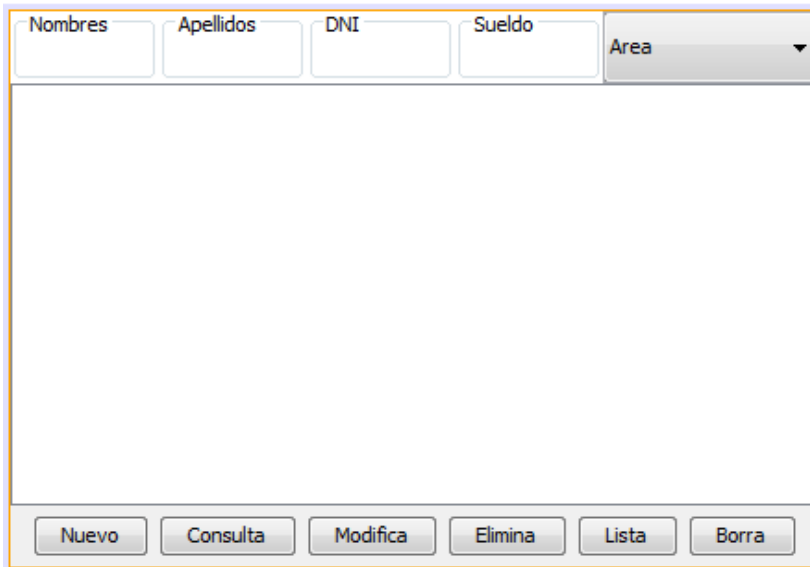
Ahora vamos a la ficha Services para conectar el servidor localhost que inicialmente está desconectado:



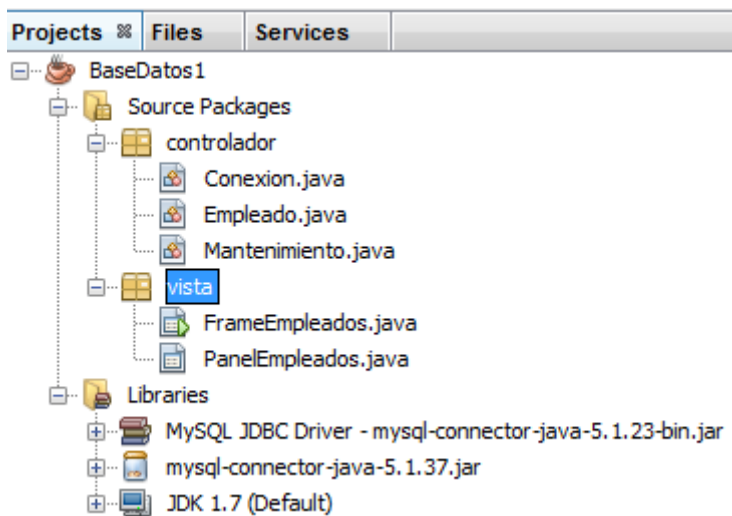
Para conectarlo, le damos clic derecho y elegimos Connect para que quede así: (Observe que nuestra base de datos creada en Workbench (dbempleados) ya es reconocida ):



Ahora, vamos a crear una aplicación con un formulario para hacer el mantenimiento de la tabla de empleados considerando el siguiente diseño:



En la ficha Projects de nuestra aplicación creamos un paquete de nombre **controlador** donde diseñamos las clases Empleado, Conexión y Mantenimiento. Luego crearemos el paquete **vista** donde diseñamos un PanelEmpleados para el formulario y un FrameEmpleados para ejecutar el panel, de la siguiente manera:



Las clases Empleado y Mantenimiento lo desarrollaremos en la siguiente sesión. Ahora sólo probaremos la conexión con el diseño de la siguiente clase Conexión:



```
package controlador;
import java.sql.*; // acceso a Connection, Statement y ResultSet
public class Conexion {
    // atributos de uso compartido
    protected static String servidor="jdbc:mysql://localhost/dbempleados";
    protected static String usuario="root";
    protected static String password="root";
    protected static String driver="com.mysql.jdbc.Driver";
    protected static Connection conexion;

    // constructor que realiza la conexión a la base de datos
    public Conexion(){
        try{
            Class.forName(driver);
            conexion=DriverManager.getConnection(servidor,usuario,password);
            System.out.println("conexion exitosa!");
        }catch(ClassNotFoundException | SQLException e){
            System.out.println("Conexion fallida: "+e.getMessage());
        }
    }

    // método que retorna el resultado de la conexión realizada
    public Connection getConexion(){
        return conexion;
    }
}
```

En el constructor del PanelEmpleados escribimos lo siguiente:

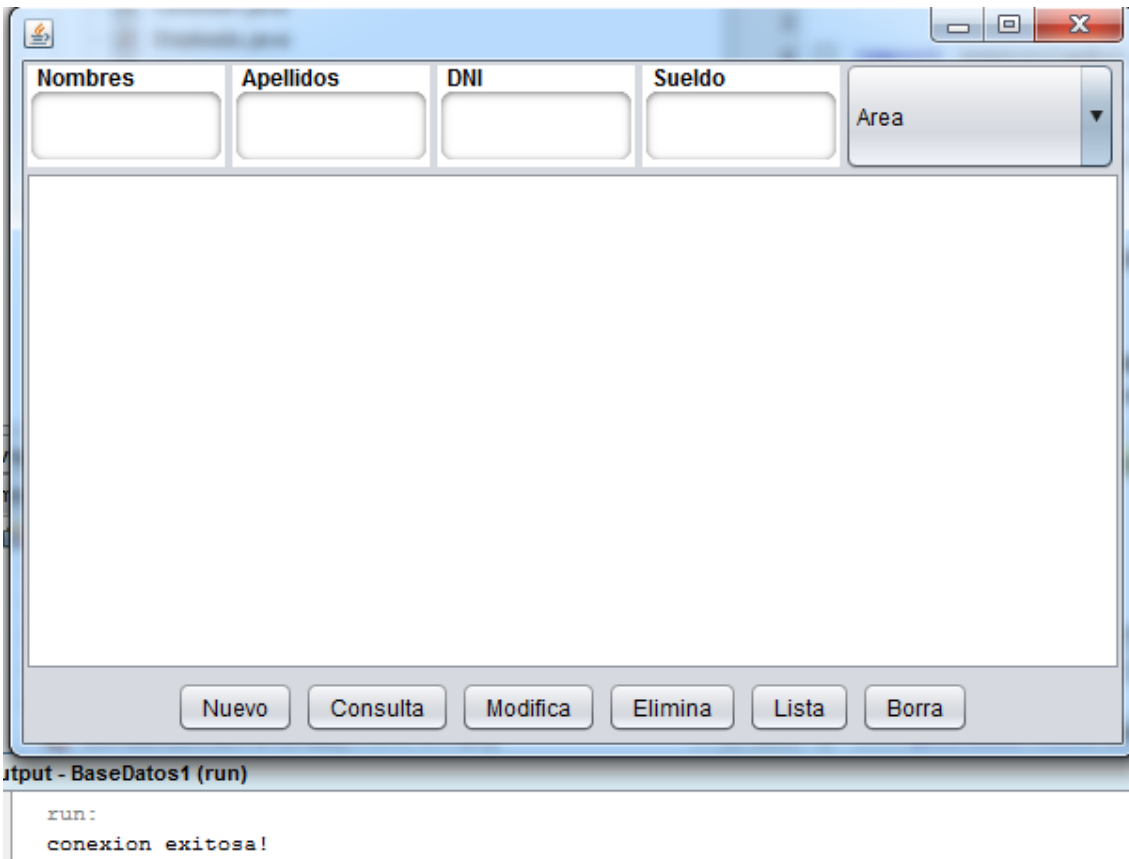
```
package vista;

import controlador.Conexion;

public class PanelEmpleados extends javax.swing.JPanel {

    public PanelEmpleados() {
        initComponents();
        // conecta a la base de datos
        Conexion con = new Conexion();
    }
}
```

Ejecutamos nuestra aplicación:



Output - BaseDatos1 (run)

```
run:
conexion exitosa!
```

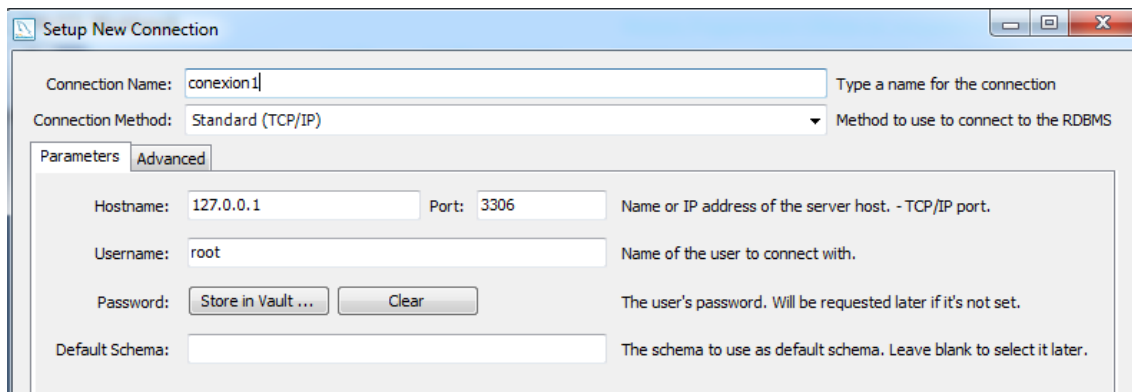
Observe el mensaje en la salida de consola: “conexión exitosa! “

## GUIA DE LABORATORIO 13

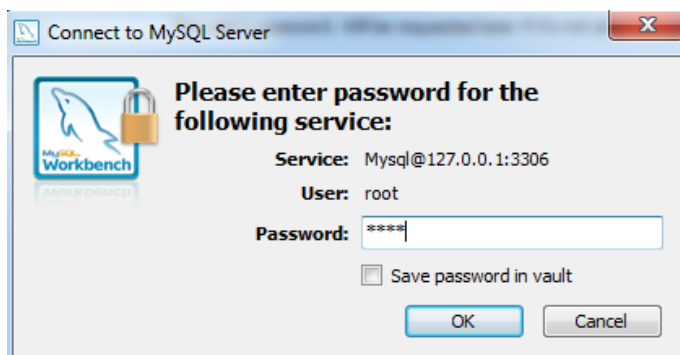
### Introducción a bases de datos

#### Creación de una Base de Datos

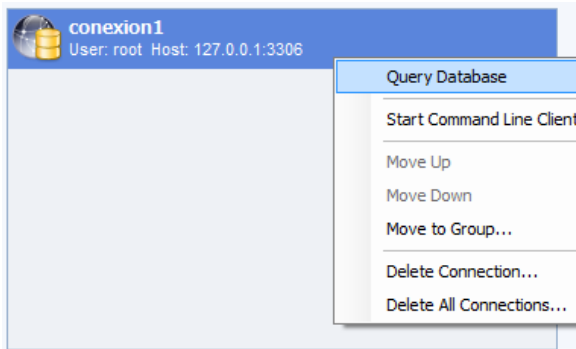
Vamos a crear nuestra base de datos en el gestor de base de datos MySQL WorkBench. Abrimos Workbeanch, elegimos **New Conexion** para llegar a la siguiente ventana:



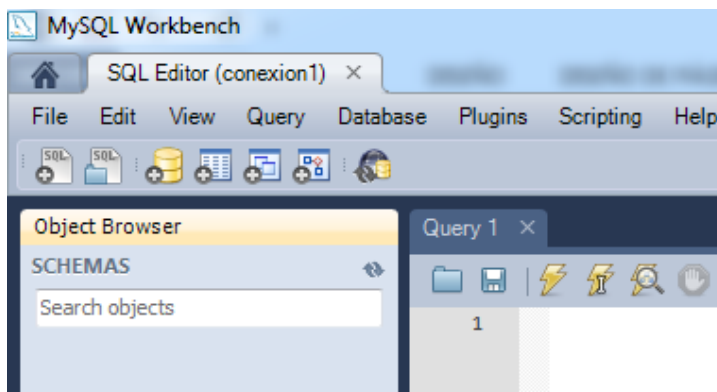
Escribimos el nombre de la nueva conexión (conexion1) y damos clic en el botón Test connection para llegar a la siguiente ventana:



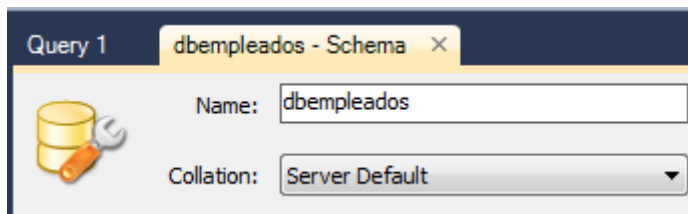
Damos el password (root) y presionamos el botón OK para llegar a la ventana de conexión con éxito y luego damos OK nuevamente para que quede establecida nuestra nueva conexión.



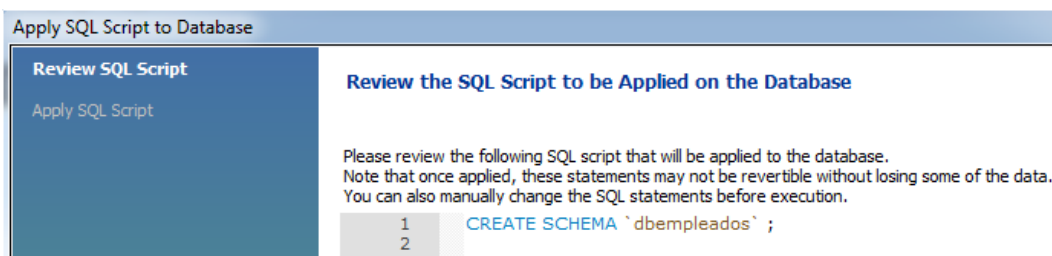
Damos clic derecho en la nueva conexión creada y elegimos Query Database para llegar a la siguiente ventana:



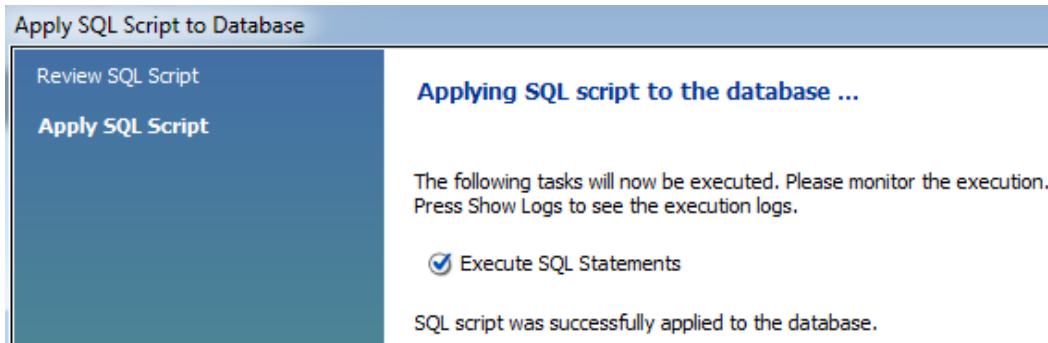
Damos clic en el ícono para crear un nuevo esquema



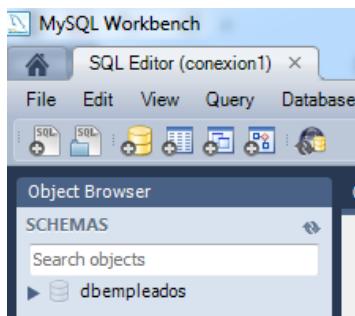
Escribimos el nombre del esquema (dbempleados) y damos clic en el botón Apply para llegar a la siguiente ventana:



Damos clic en el botón Apply nuevamente para llegar a la siguiente ventana:

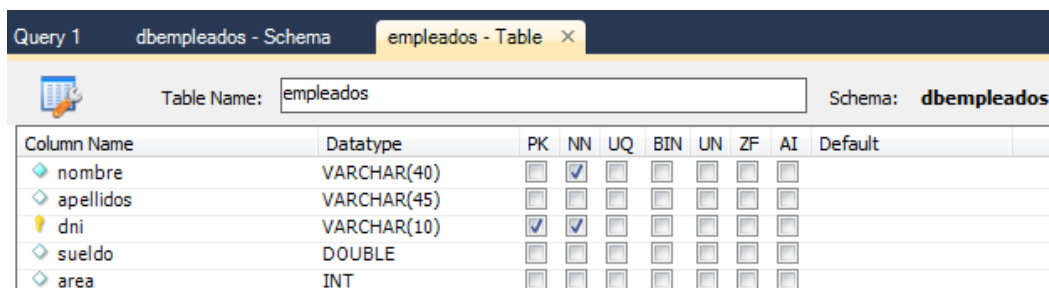


Damos clic en el botón Finish para llegar a la siguiente ventana:

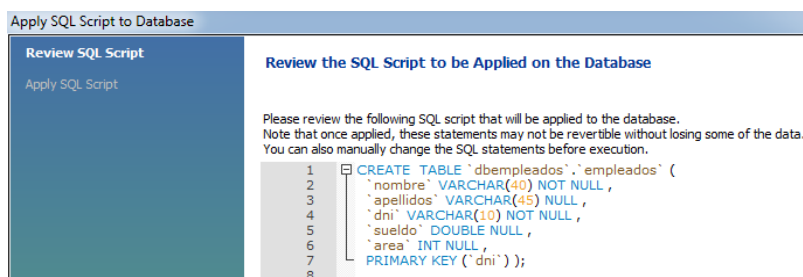


Donde comprobamos que se ha creado un nuevo esquema (dbempleados) donde daremos clic derecho y elegimos set as default schema.

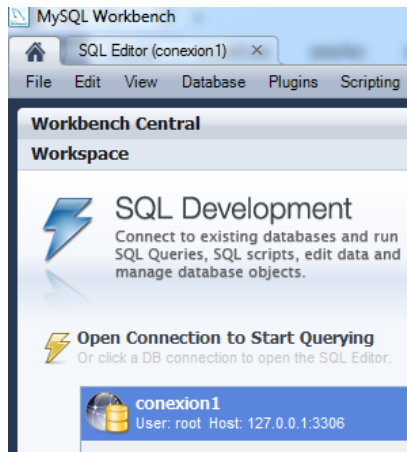
Luego damos clic en el ícono para crear una nueva tabla donde escribimos el nombre y llenamos los campos (column name) conforme a la siguiente ventana:



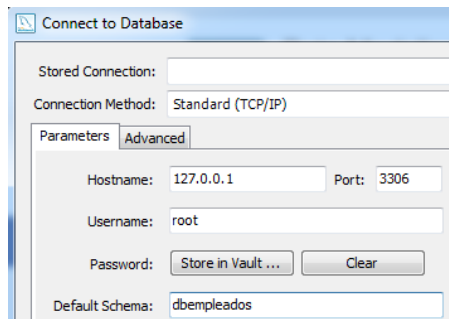
Damos clic en el botón Apply para llegar a la siguiente ventana:



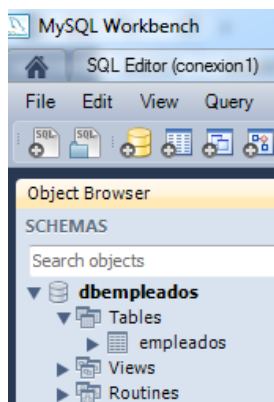
Damos clic en el botón Apply y luego en Finish. Vamos a la ventana de inicio de Workbench donde comprobamos que se ha creado una nueva conexión.



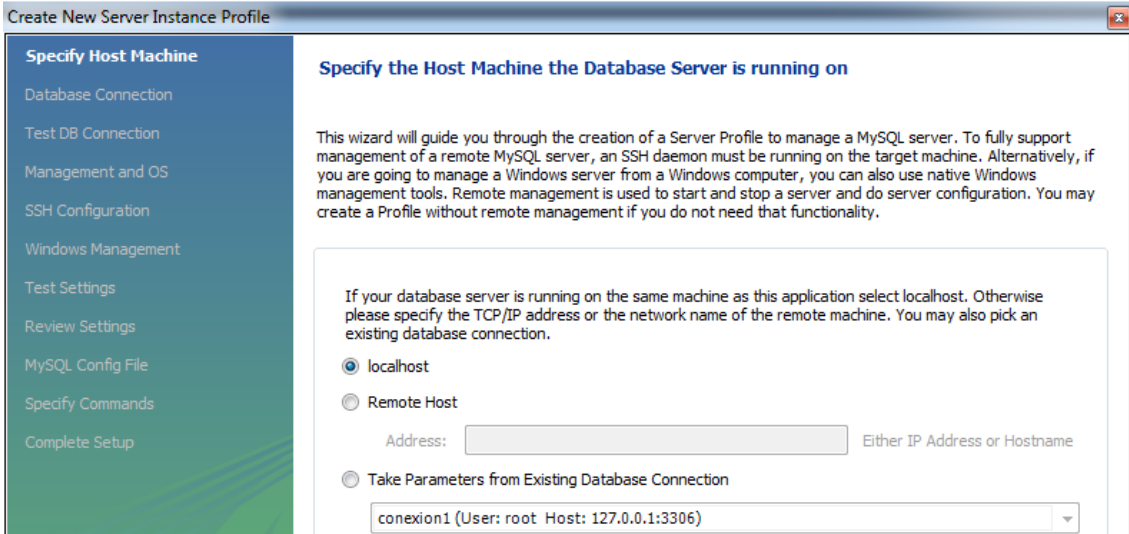
Damos clic en Open Connection to Start Querying para llegar a la siguiente ventana donde escribimos el nombre del esquema (dbempleados):



Damos clic en el botón OK, ingresamos el password para llegar a la siguiente ventana:



Regresamos al inicio de Workbench para crear una nueva instancia de servidor:



**Create New Server Instance Profile**

**Specify Host Machine**

Database Connection  
Test DB Connection  
Management and OS  
SSH Configuration  
Windows Management  
Test Settings  
Review Settings  
MySQL Config File  
Specify Commands  
Complete Setup

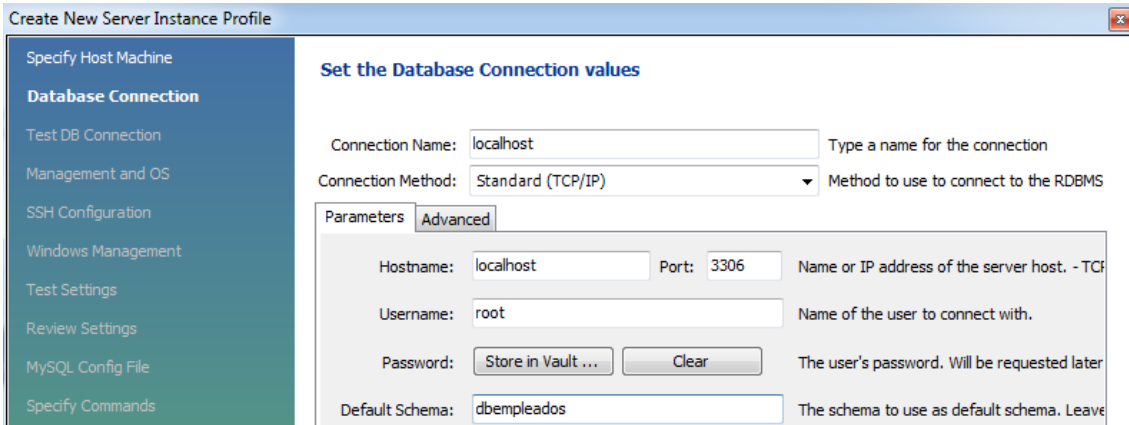
**Specify the Host Machine the Database Server is running on**

This wizard will guide you through the creation of a Server Profile to manage a MySQL server. To fully support management of a remote MySQL server, an SSH daemon must be running on the target machine. Alternatively, if you are going to manage a Windows server from a Windows computer, you can also use native Windows management tools. Remote management is used to start and stop a server and do server configuration. You may create a Profile without remote management if you do not need that functionality.

If your database server is running on the same machine as this application select localhost. Otherwise please specify the TCP/IP address or the network name of the remote machine. You may also pick an existing database connection.

☒ localhost  
☐ Remote Host  
 Address:  Either IP Address or Hostname  
☐ Take Parameters from Existing Database Connection  
 conexion1 (User: root Host: 127.0.0.1:3306)

Damos clic en el botón Next y en la siguiente ventana escribimos el nombre del esquema “dbempleados”



**Create New Server Instance Profile**

Specify Host Machine  
**Database Connection**  
Test DB Connection  
Management and OS  
SSH Configuration  
Windows Management  
Test Settings  
Review Settings  
MySQL Config File  
Specify Commands

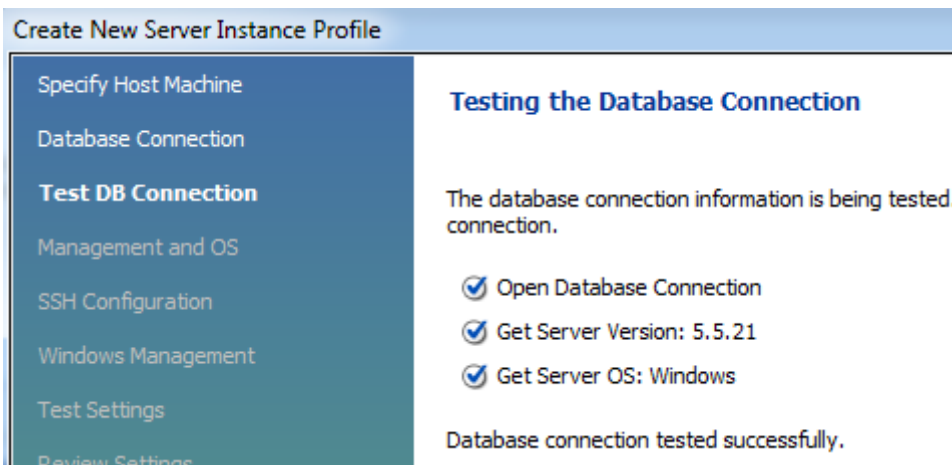
**Set the Database Connection values**

Connection Name:  Type a name for the connection  
 Connection Method:  Method to use to connect to the RDBMS

Parameters **Advanced**

Hostname:  Port:  Name or IP address of the server host. - TCP  
 Username:  Name of the user to connect with.  
 Password:   The user's password. Will be requested later  
 Default Schema:  The schema to use as default schema. Leave

Damos clic en el botón Next para llegar a la siguiente ventana:



**Create New Server Instance Profile**

Specify Host Machine  
Database Connection  
**Test DB Connection**  
Management and OS  
SSH Configuration  
Windows Management  
Test Settings  
Review Settings

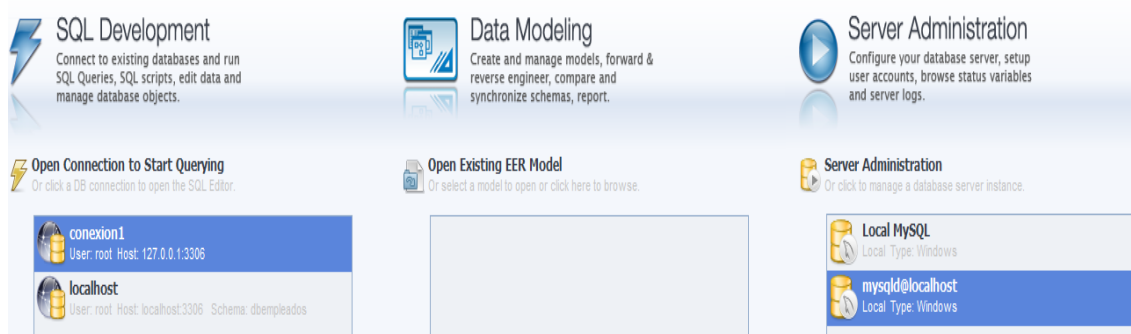
**Testing the Database Connection**

The database connection information is being tested. connection.

☒ Open Database Connection  
☒ Get Server Version: 5.5.21  
☒ Get Server OS: Windows

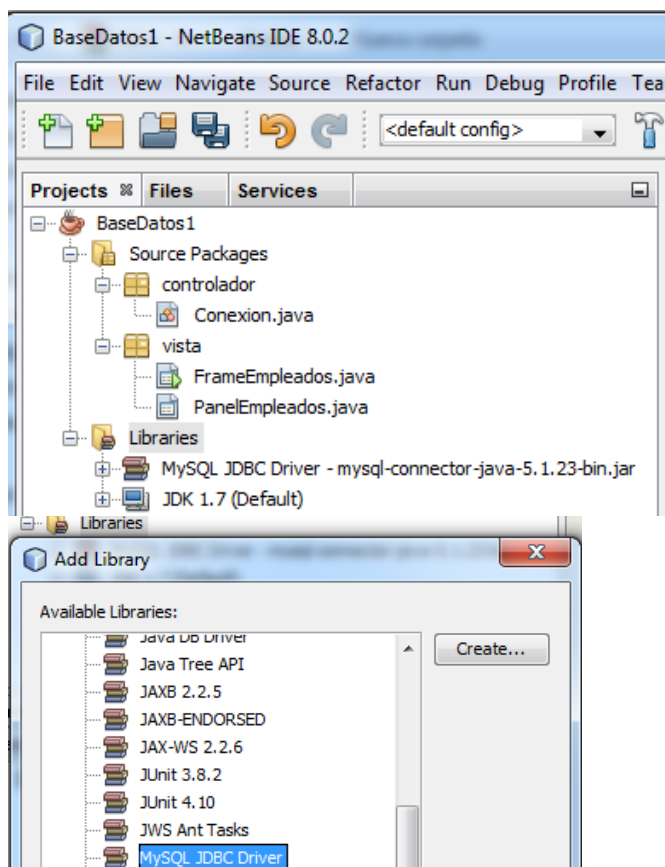
Database connection tested successfully.

Damos clic en el botón **Next** hasta finalizar. Vamos al inicio del Workbench para verificar que se ha creado la conexión y la instancia del servidor:



## Conexión a una Base de Datos

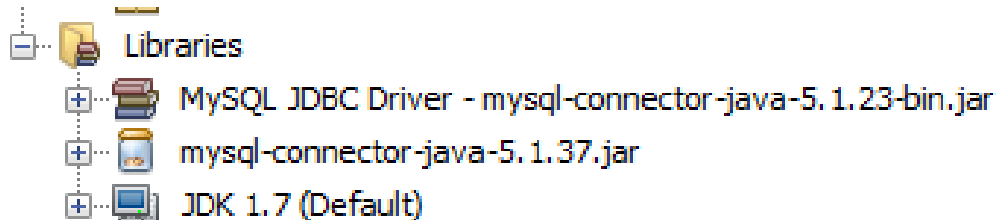
Entramos a NetBeans, creamos un proyecto para incorporar el driver descargado y la librería en la carpeta librerías de nuestra aplicación: Clic derecho en Libraries de nuestra aplicación y elegimos Library y seleccionamos MySQL JDBC Driver.



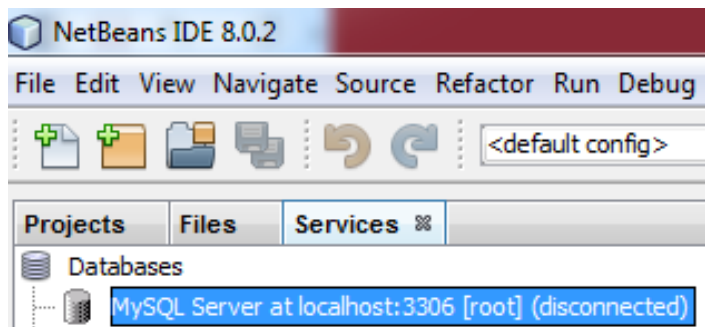


Luego de aceptar, verifique su incorporación en las librerías de nuestra aplicación:

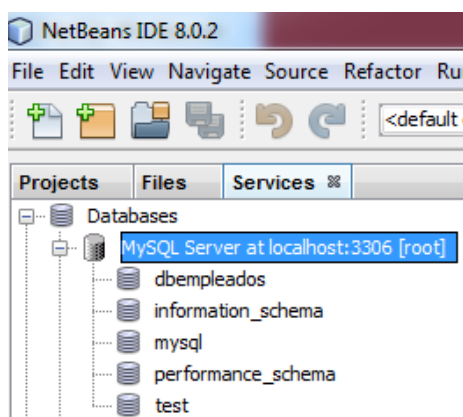
Ahora vamos a incorporar el driver conector que hemos descargado de internet. Nuevamente clic derecho en la carpeta Libraries de nuestra aplicación, elegimos add JAR/folder para que finalmente quede así la carpeta Libraries de nuestra aplicación:



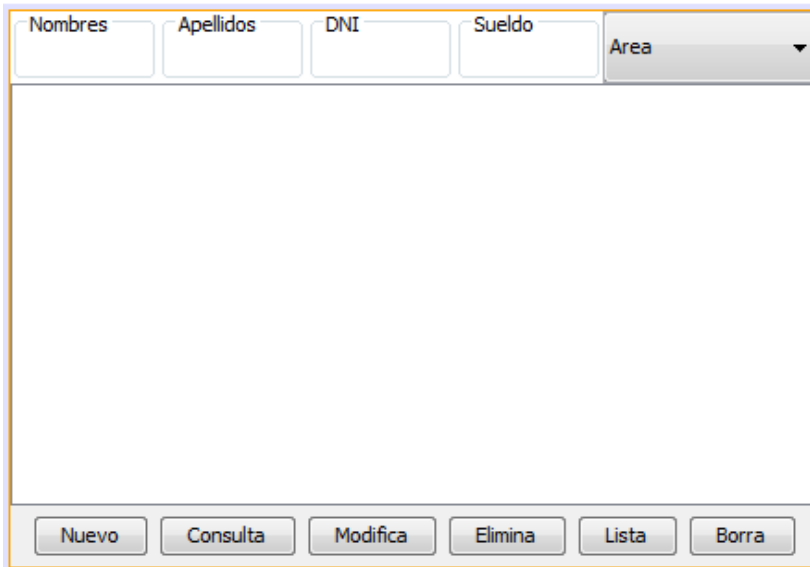
Ahora vamos a la librería Services para conectar el servidor localhost que inicialmente está desconectado:



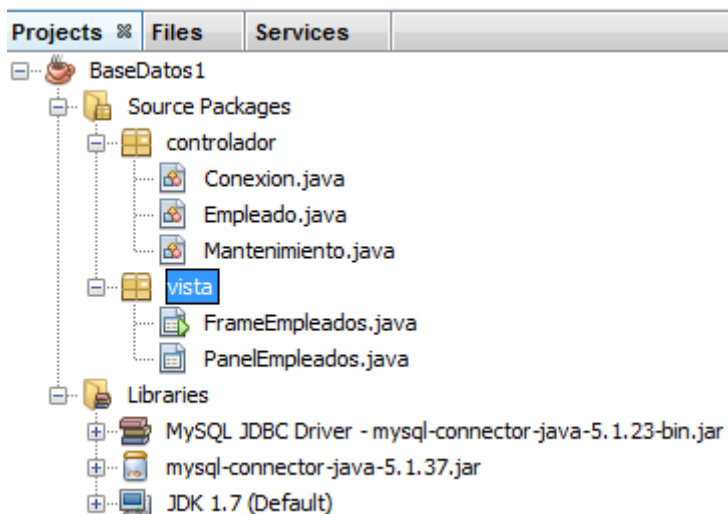
Para conectarlo, le damos clic derecho y elegimos **Connect** para que quede así: (Observe que nuestra base de datos creada en Workbench (dbempleados) ya es reconocida):



Ahora, vamos a crear una aplicación con un formulario para hacer el mantenimiento de la tabla de empleados considerando el siguiente diseño:



En la ficha Projects de nuestra aplicación creamos un paquete de nombre **controlador** donde diseñamos las clases Empleado, Conexión y Mantenimiento. Luego crearemos el paquete **vista** donde diseñamos un PanelEmpleados para el formulario y un FrameEmpleados para ejecutar el panel, de la siguiente manera:



Las clases **Empleado** y **Mantenimiento** lo desarrollaremos en la siguiente sesión. Ahora sólo probaremos la conexión con el diseño de la siguiente clase Conexión:

```
package controlador;
import java.sql.*; // acceso a Connection, Statement y ResultSet
public class Conexion {
    // atributos de uso compartido
    protected static String servidor="jdbc:mysql://localhost/dbempleados";
    protected static String usuario="root";
    protected static String password="root";
    protected static String driver="com.mysql.jdbc.Driver";
    protected static Connection conexion;

    // constructor que realiza la conexión a la base de datos
    public Conexion(){
        try{
            Class.forName(driver);
            conexion=DriverManager.getConnection(servidor,usuario,password);
            System.out.println("conexion exitosa!");
        }catch(ClassNotFoundException | SQLException e){
            System.out.println("Conexion fallida: "+e.getMessage());
        }
    }

    // método que retorna el resultado de la conexión realizada
    public Connection getConexion(){
        return conexion;
    }
}
```

En el constructor del PanelEmpleados escribimos lo siguiente:

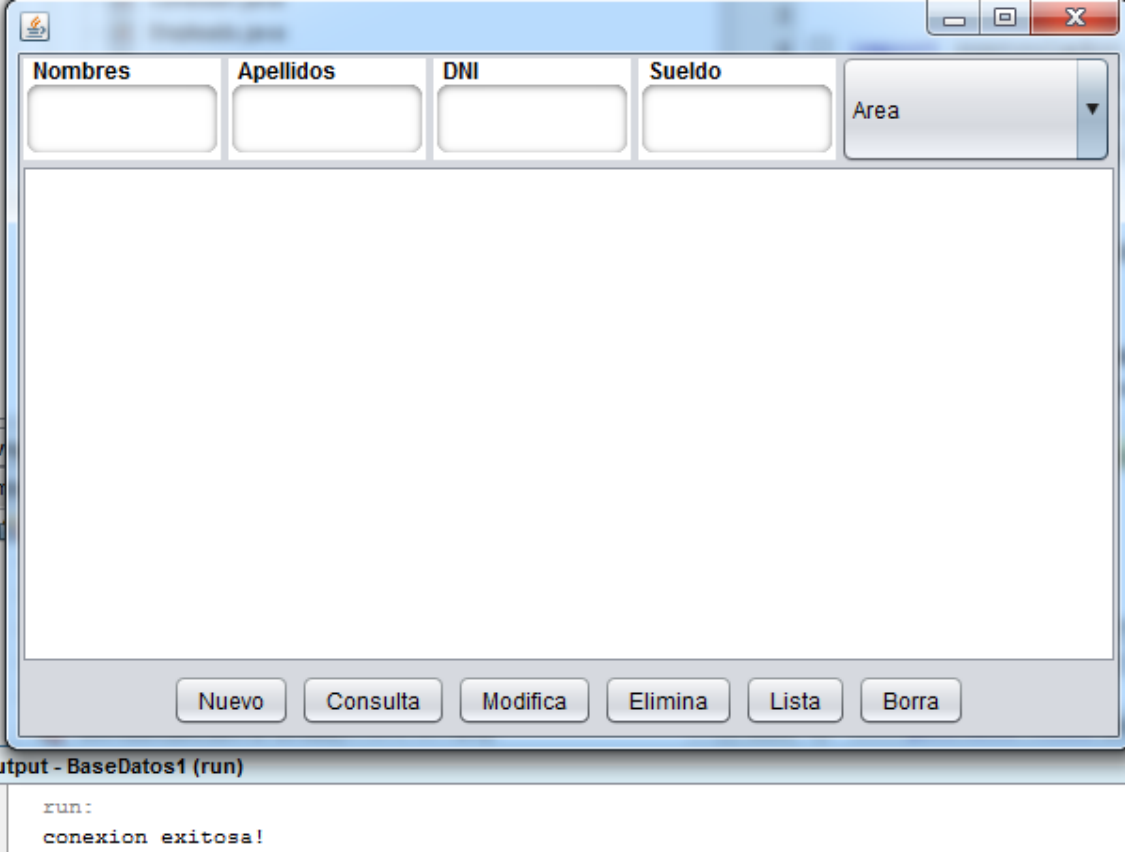
```
package vista;

import controlador.Conexion;

public class PanelEmpleados extends javax.swing.JPanel {

    public PanelEmpleados() {
        initComponents();
        // conecta a la base de datos
        Conexion con = new Conexion();
    }
}
```

Ejecutamos nuestra aplicación:



Output - BaseDatos1 (run)

```
run:  
conexion exitosa!
```

Observe el mensaje en la salida de consola: “conexión exitosa!”