

CURSO DE JAVA CON JDBC

EJERCICIO

INTRODUCCIÓN A JDBC



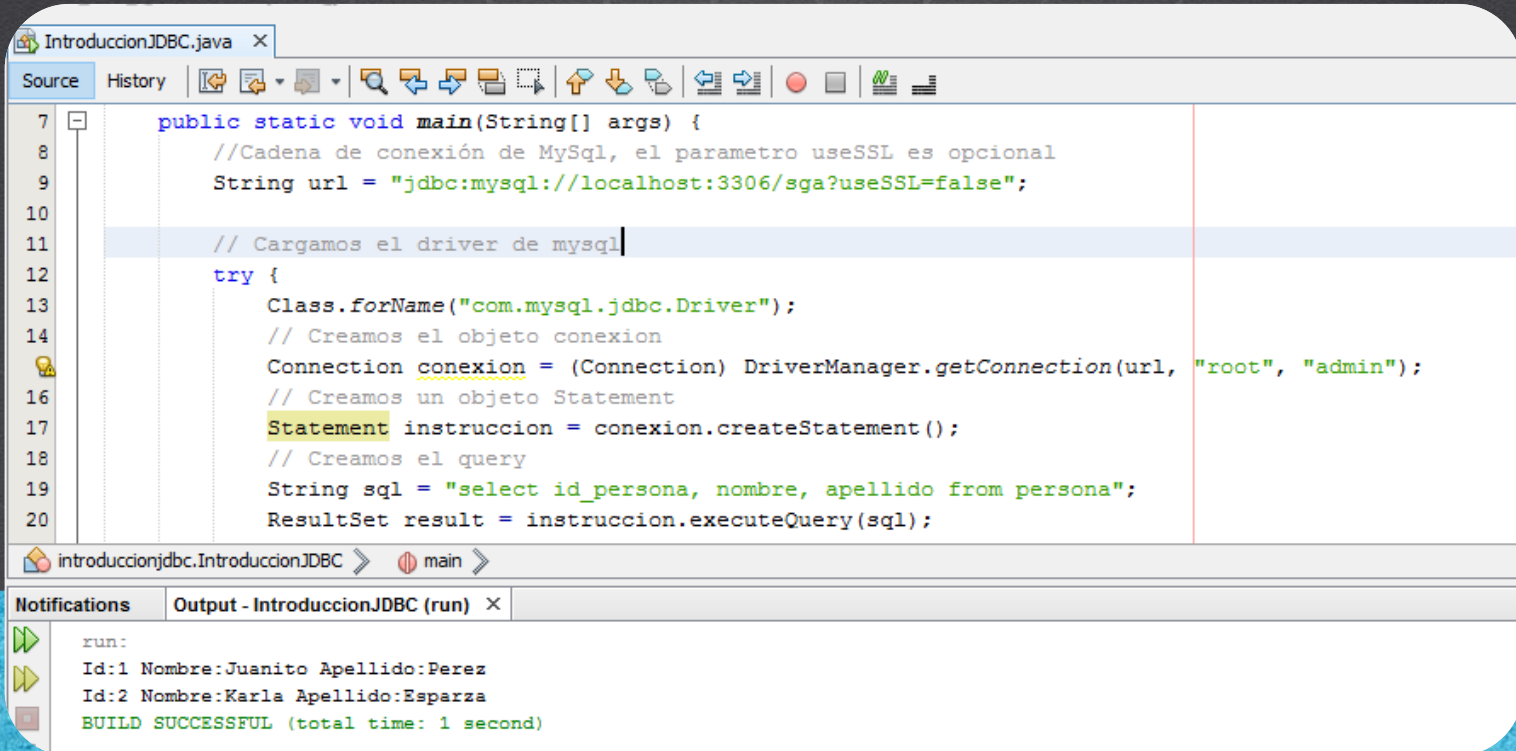
Experiencia y Conocimiento para tu vida

CURSO DE JAVA CON JDBC

www.globalmentoring.com.mx

OBJETIVO DEL EJERCICIO

Crear el ejercicio de introducción a JDBC. Al finalizar deberemos observar lo siguiente:



The screenshot displays an IDE window titled 'IntroduccionJDBC.java'. The code defines a `main` method that establishes a JDBC connection to a MySQL database. It includes comments in Spanish explaining each step: connecting to the database, loading the driver, creating a connection, creating a statement, and executing a query. The query selects data from a table named 'persona'. The IDE's interface includes a toolbar with various editing and running tools. Below the code editor, the 'Output - IntroduccionJDBC (run)' window shows the execution results, indicating a successful build and providing the output of the query.

```
7 public static void main(String[] args) {
8     //Cadena de conexión de MySql, el parametro useSSL es opcional
9     String url = "jdbc:mysql://localhost:3306/sga?useSSL=false";
10
11     // Cargamos el driver de mysql
12     try {
13         Class.forName("com.mysql.jdbc.Driver");
14         // Creamos el objeto conexion
15         Connection conexion = (Connection) DriverManager.getConnection(url, "root", "admin");
16         // Creamos un objeto Statement
17         Statement instruccion = conexion.createStatement();
18         // Creamos el query
19         String sql = "select id_persona, nombre, apellido from persona";
20         ResultSet result = instruccion.executeQuery(sql);
```

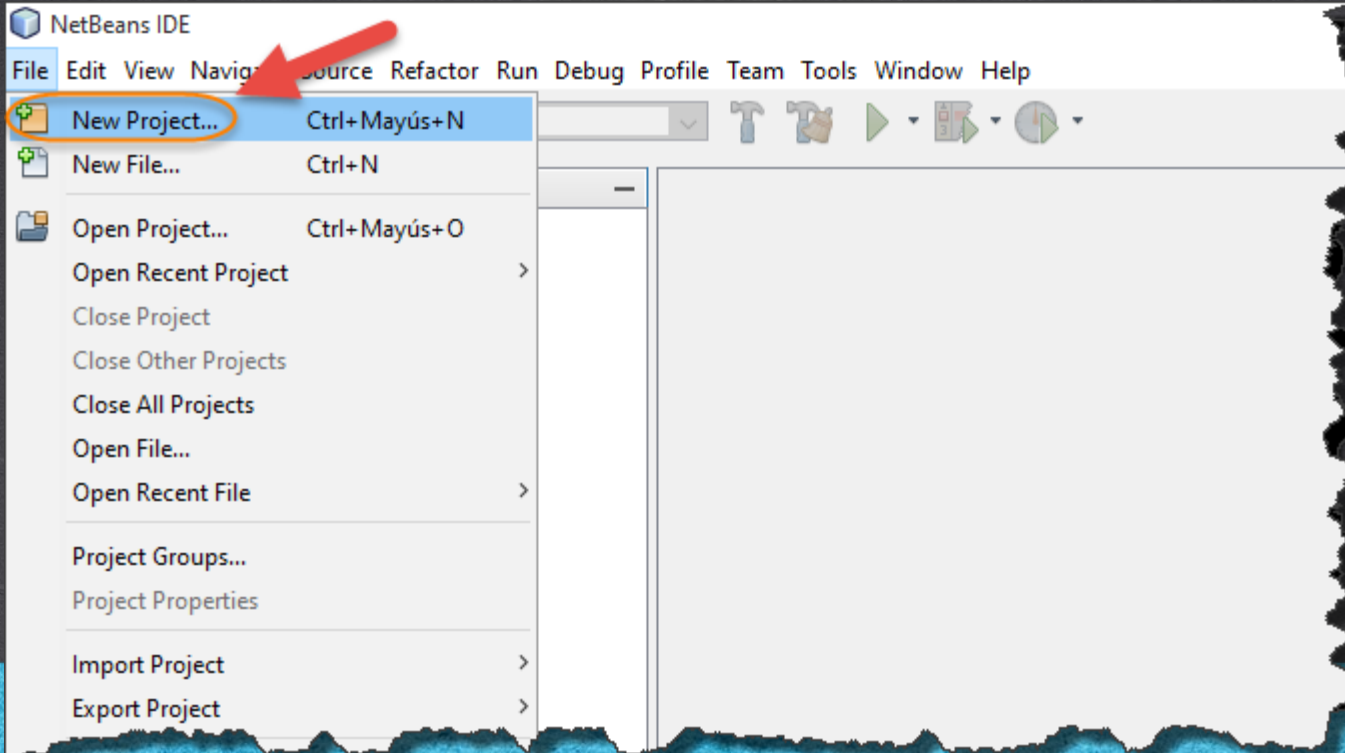
introduccionjdbc.IntroduccionJDBC > main >

Notifications Output - IntroduccionJDBC (run) X

run:
Id:1 Nombre:Juanito Apellido:Perez
Id:2 Nombre:Karla Apellido:Esparza
BUILD SUCCESSFUL (total time: 1 second)

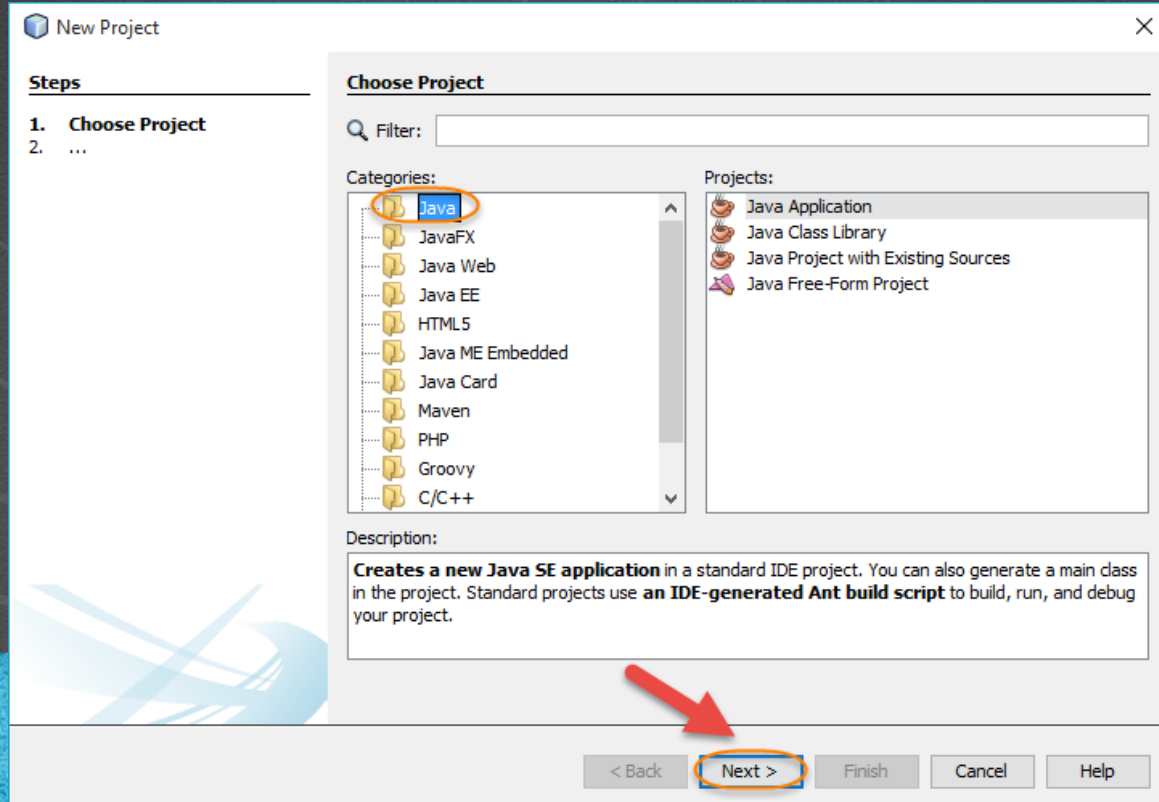
PASO 1. CREACIÓN DEL PROYECTO

Vamos a crear el proyecto:



PASO 1. CREACIÓN DEL PROYECTO (CONT)

Vamos a crear el proyecto:



PASO 1. CREACIÓN DEL PROYECTO (CONT)

Vamos a crear el proyecto:

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

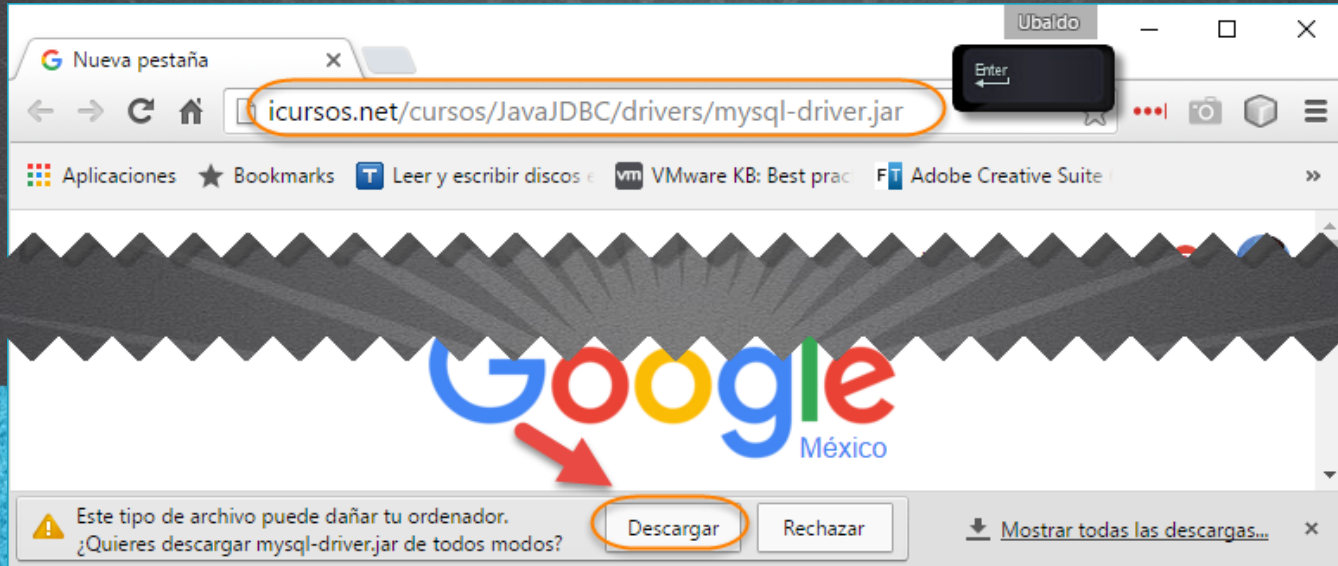
☒ Create Main Class

PASO 2. DESCARGAR EL DRIVER DE MYSQL

Descargar el driver de MySql del link:

<http://icursos.net/cursos/JavaJDBC/drivers/mysql-driver.jar>

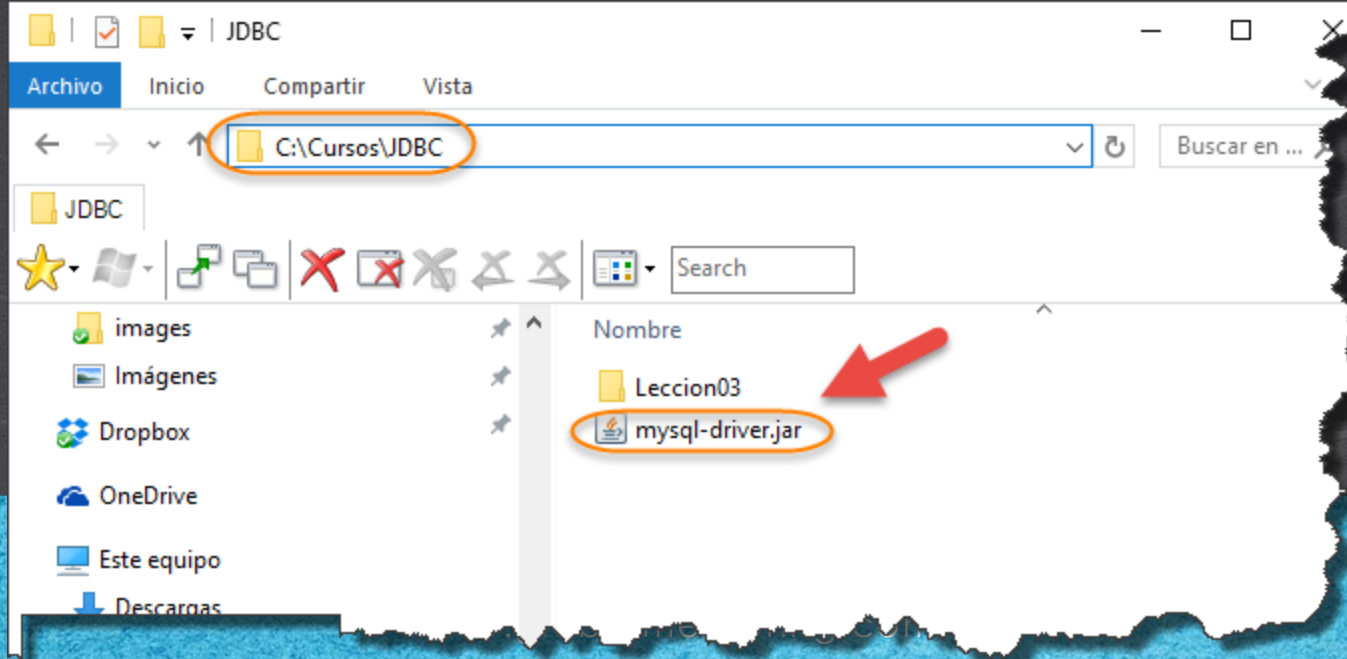
Aceptar la descarga en caso de que pregunte:



PASO 3. GUARDAR EL DRIVER DE MYSQL

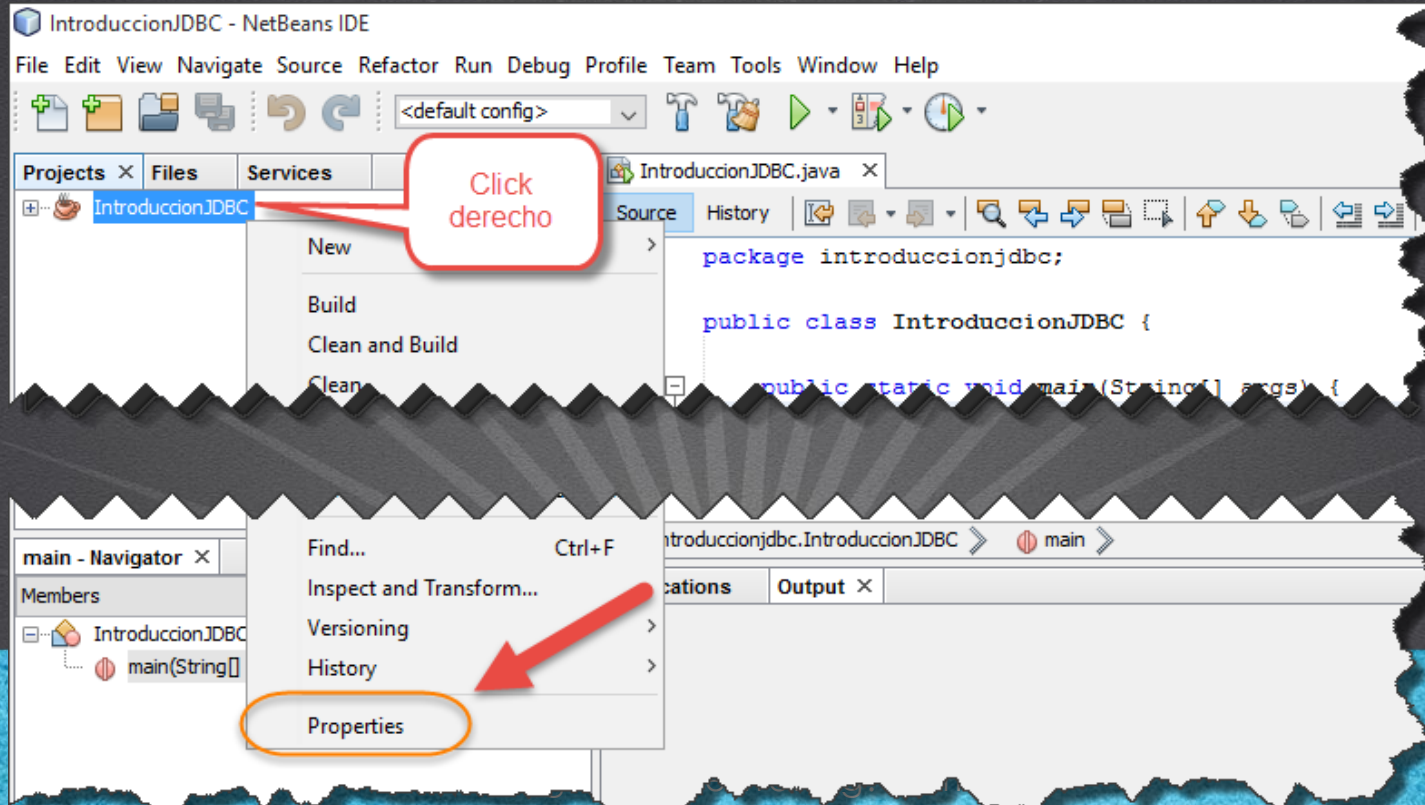
Guardamos el driver de mysql en alguna carpeta, por ejemplo:

C:\Cursos\JDBC



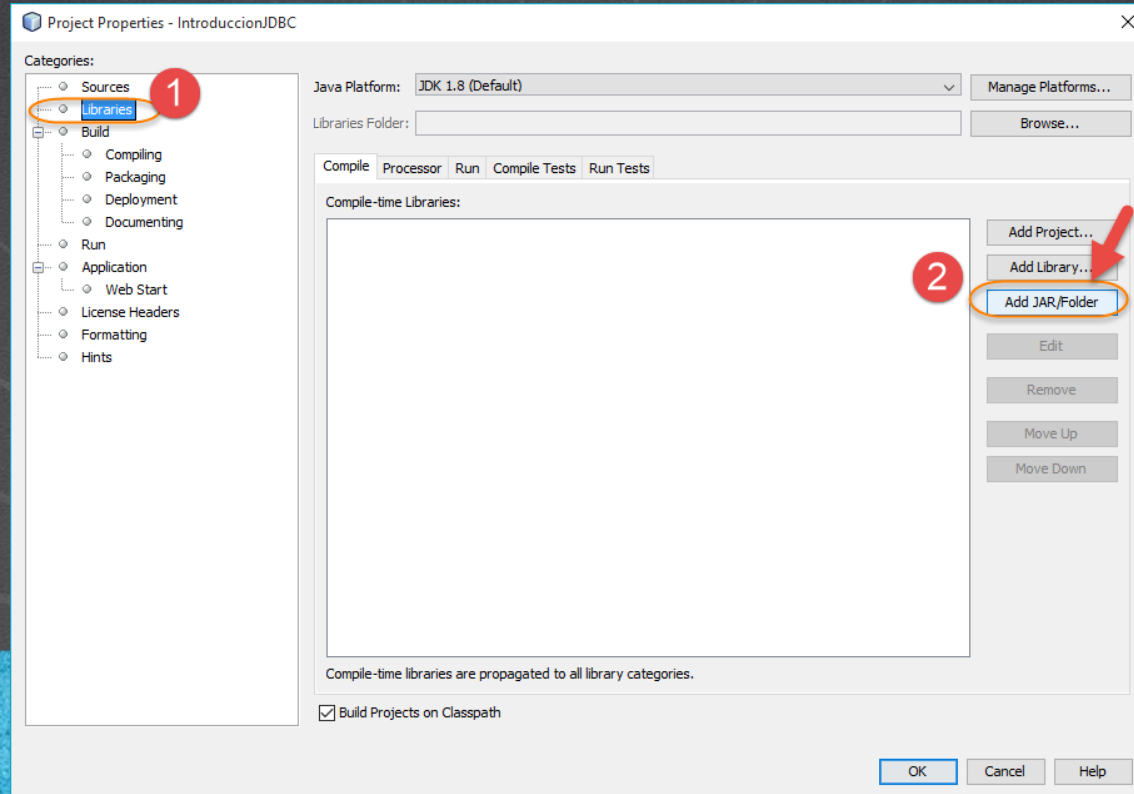
PASO 4. AGREGAR EL DRIVER AL CLASSPATH

Agregamos el driver al classpath de la aplicación como sigue:



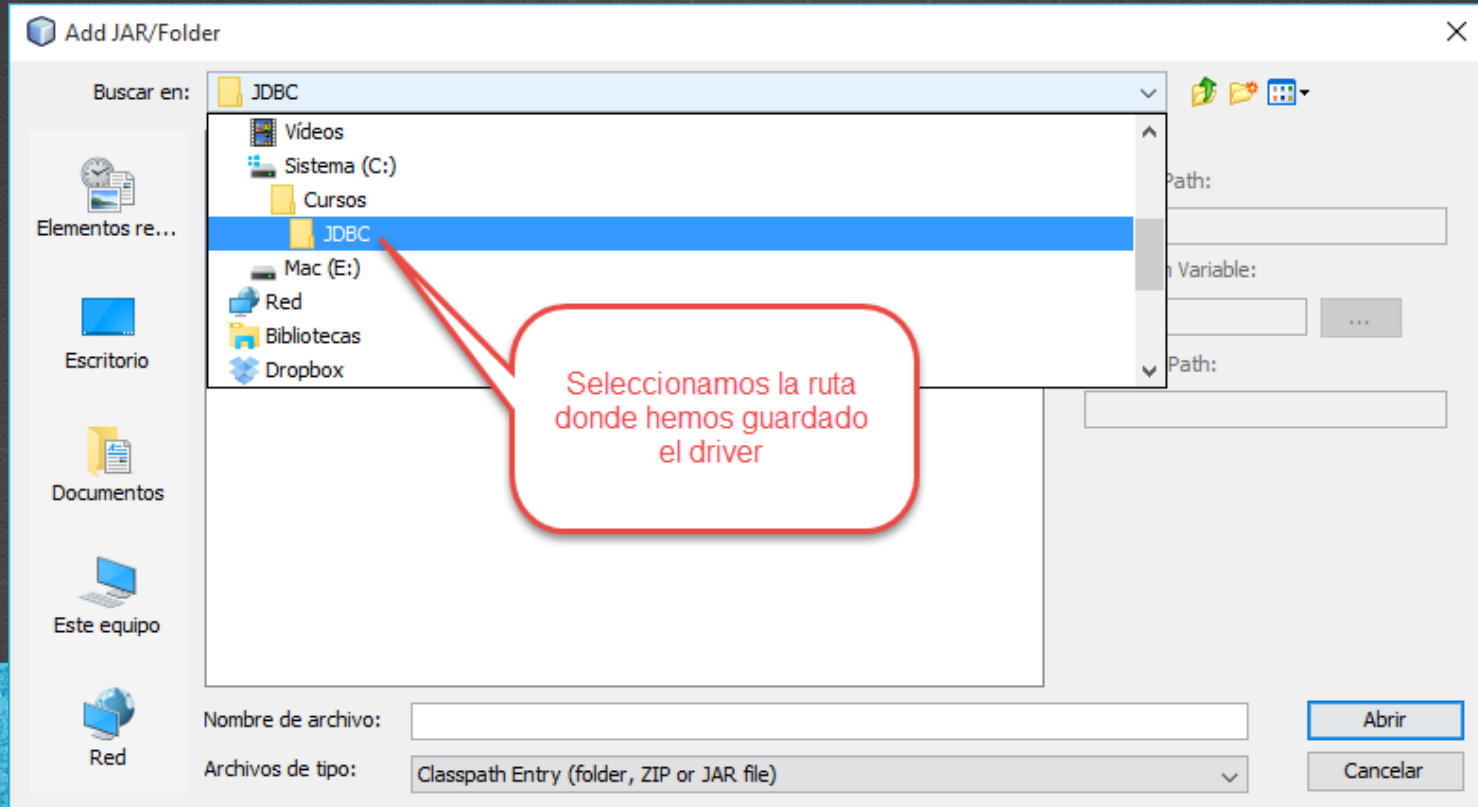
PASO 4. AGREGAR EL DRIVER AL CLASSPATH (CONT)

Agregamos el driver al classpath de la aplicación como sigue:



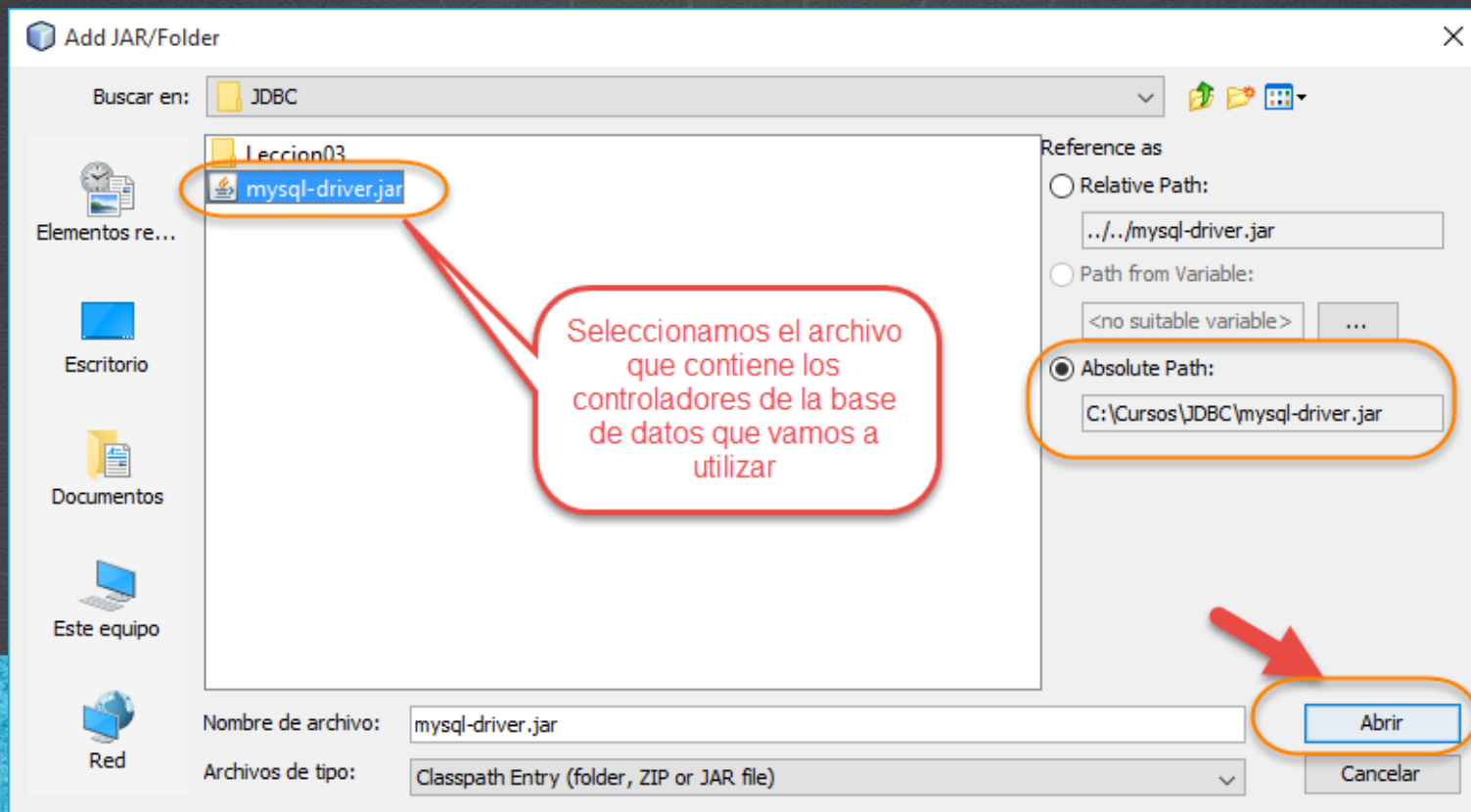
PASO 4. AGREGAR EL DRIVER AL CLASSPATH (CONT)

Agregamos el driver al classpath de la aplicación como sigue:



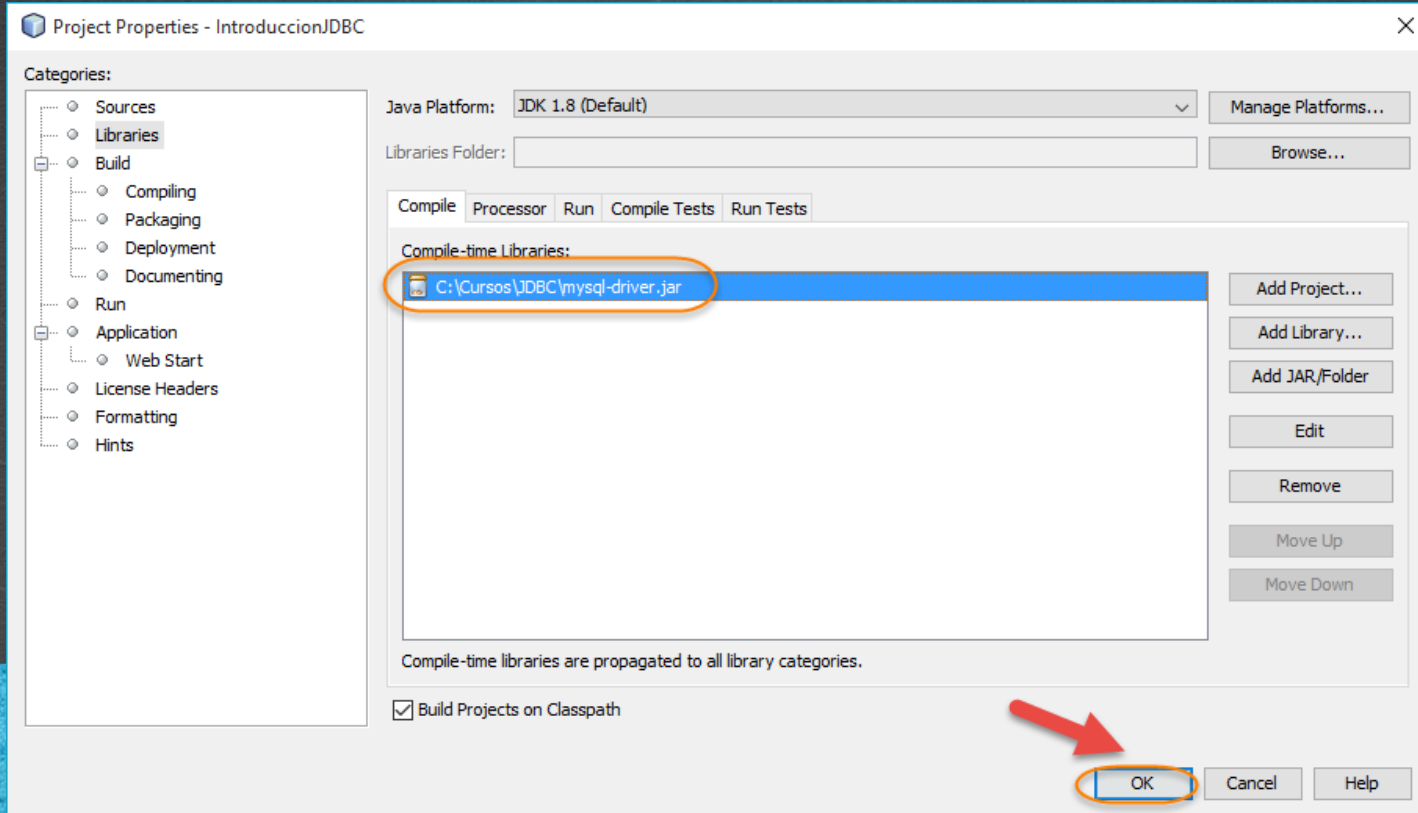
PASO 4. AGREGAR EL DRIVER AL CLASSPATH (CONT)

Agregamos el driver al classpath de la aplicación como sigue:



PASO 4. AGREGAR EL DRIVER AL CLASSPATH (CONT)

Agregamos el driver al classpath de la aplicación como sigue:



PASO 5. MODIFICAMOS EL CÓDIGO

Archivo IntroduccionJDBC.java:

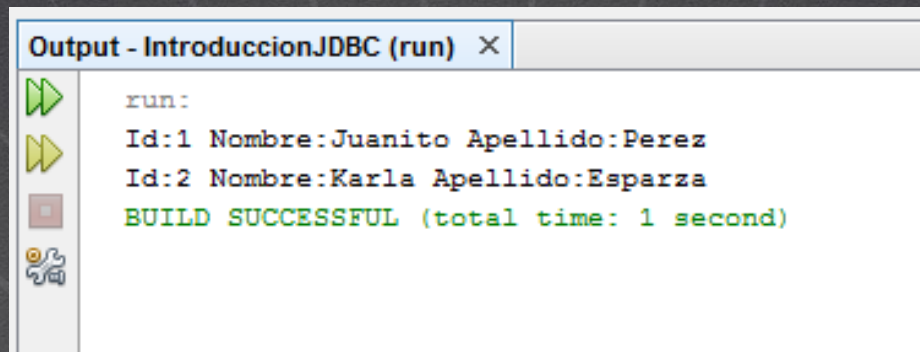
```
package introduccionjdbc;
import java.sql.*;

public class IntroduccionJDBC {

    public static void main(String[] args) {
        //Cadena de conexión de MySql, el parametro useSSL es opcional
        String url = "jdbc:mysql://localhost:3306/sga?useSSL=false";
        // Cargamos el driver de mysql
        try {
            Class.forName("com.mysql.jdbc.Driver");
            // Creamos el objeto conexion
            Connection conexion = (Connection) DriverManager.getConnection(url, "root", "admin");
            // Creamos un objeto Statement
            Statement instruccion = conexion.createStatement();
            // Creamos el query
            String sql = "select id_persona, nombre, apellido from persona";
            ResultSet result = instruccion.executeQuery(sql);
            while (result.next()) {
                System.out.print("Id:" + result.getInt(1));
                System.out.print(" Nombre:" + result.getString(2));
                System.out.println(" Apellido:" + result.getString(3));
            }
            // Cerrar cada uno de los objetos utilizados
            result.close();
            instruccion.close();
            conexion.close();
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
        }
    }
}
```

PASO 6. EJECUTAMOS EL PROYECTO

El resultado es como sigue:

A screenshot of an IDE's output window titled "Output - IntroduccionJDBC (run)". The window contains a list of icons on the left (run, debug, test, and a gear icon) and a text area on the right. The text area displays the following output: "run:", "Id:1 Nombre:Juanito Apellido:Perez", "Id:2 Nombre:Karla Apellido:Esparza", and "BUILD SUCCESSFUL (total time: 1 second)".

```
run:
Id:1 Nombre:Juanito Apellido:Perez
Id:2 Nombre:Karla Apellido:Esparza
BUILD SUCCESSFUL (total time: 1 second)
```


CONCLUSIÓN DEL EJERCICIO

- Con este ejercicio hemos hecho nuestra primera conexión a JDBC, y consultado información de la base de datos que creamos en la lección anterior.
- No es necesario que todo el código quede claro en este momento, conforme vayamos avanzando iremos trabajando y profundizando en cada uno de los elementos de este ejemplo.



CURSO ONLINE

JAVA CON JDBC

Por: Ing. Ubaldo Acosta



Experiencia y Conocimiento para tu vida

CURSO DE JAVA CON JDBC

www.globalmentoring.com.mx