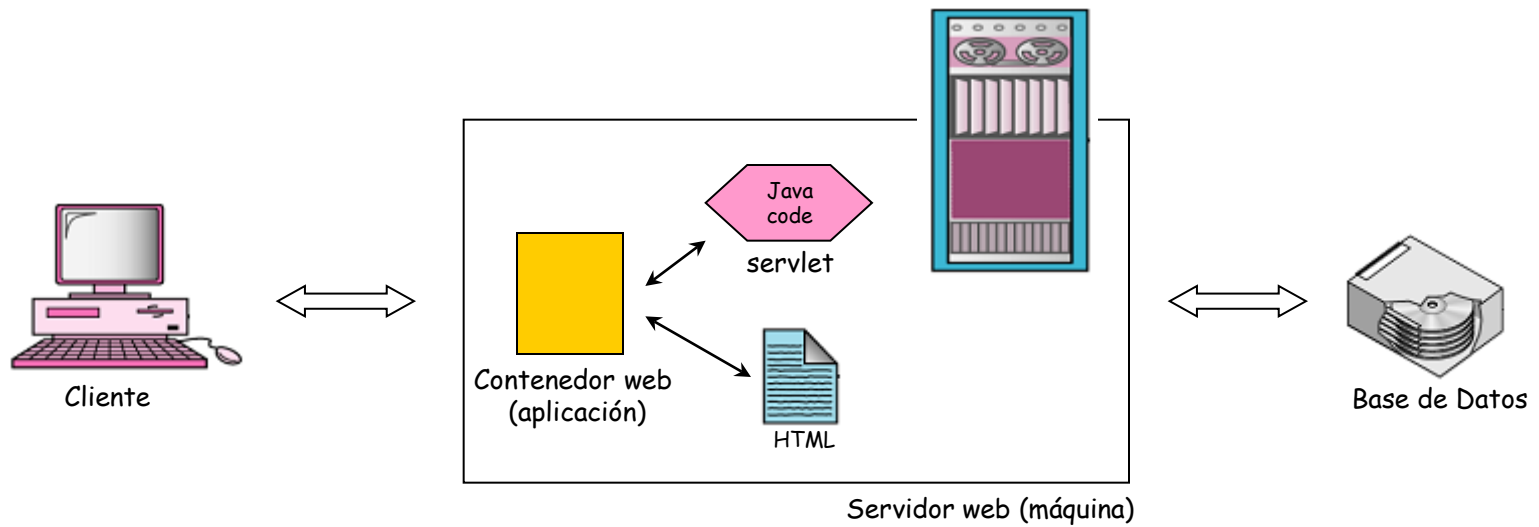


Desarrollo de Aplicaciones Web

JDBC

JDBC



JDBC

El acceso a las a bases de datos, mediante JDBC, tiene siete pasos básicos:

1. Cargar el driver del JDBC correspondiente.
2. Definir la conexión URL.
3. Establecer la conexión.
4. Crear objeto *statement*.
5. Ejecutar una petición.
6. Procesar los resultados.
7. Cerrar la conexión.

JDBC

1. Cargar el driver del JDBC correspondiente:

String controlador= “controlador del gestor de bases de datos”

Class.forName(controlador).newInstance();

Controladores:

- connect.microsoft.MicrosoftDriver
- oracle.jdbc.driver.OracleDriver
- como.sybase.jdbc.sybDriver
- com.mysql.jdbc.Driver

JDBC

2. Definir la conexión URL:

```
String URL_DB= "jdbc:mysql://host:port/nombreDB"
```

Ej.-

```
String URL_bd = "jdbc:mysql://127.0.0.1:3306/bDatos"
```

JDBC

3. Establecer la conexión:

String username= "username"

String password= "contraseña"

Connection connection= DriverManager.getConnection(URL_bd,
username, password)

JDBC

4. Crear objeto *statement*: permite enviar peticiones a la DB

```
Statement statement= connection.createStatement()
```

JDBC

5. Ejecutar una petición: mediante el método *executeQuery*

```
String query= "SELECT col_1, col_2, ... FROM Tabla_n"
```

```
ResultSet resultSet= statement.executeQuery(query)
```

Modificación de DB → utilizar el método *executeUpdate*, en lugar de *executeQuery* e incluir en la petición *INSERT*, *UPDATE* o *DELETE*.

JDBC

6. Procesar los resultados: se procesan fila a fila mediante el método *next()* de *ResultSet*.

Para cada fila, *ResultSet* proporciona métodos *getXXX*, que permiten extraer valores por columna y tipo (*getInt()*, *getString()*).

Ej.-

```
while(resultSet.next()) {  
    System.out.println(resultSet.getString(1) + " " +  
                        resultSet.getString(2) + " " +  
                        resultSet.getString(3) )  
}
```

JDBC

6. Procesar los resultados:

Otros métodos de `ResultSet` son:

- *findColumn()*.- devuelve el índice de una columna dada.
- *wasNull()*.- compara el valor de retorno con *null*.
- *getMetaData()*.- devuelve información acerca de los objetos *ResultSet* y *ResultSetMetaData*.

JDBC

6. Procesar los resultados:

Métodos de `ResultSetMetaData` son:

- *getColumnCount()*.- devuelve número de columnas.
- *columnName()*.- devuelve nombre de columna.
- *getColumnType()*.- devuelve tipo de dato en columna.
- *isReadOnly()*.- entrada solo lectura?
- *isSearchable()*.- uso en clausulas where.
- *isNullable()*.- valor nulo permitido?

JDBC

7. Cerrar la conexión :

```
Connection.close()
```