

¿Qué es JDBC?

Java Database Connectivity o JDBC es una API que permite la ejecución de operaciones y consultas sobre bases de datos desde Java.

Concepto



Independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede

Es una colección de interfaces Java y métodos para la gestión y conexión hacia cada motor específico de base de datos

Cada proveedor de base de datos debe implementar su propio driver o adaptador

A partir de allí puede realizar con cualquier tipo operaciones con la base de datos: consulta, creación, modificación y borrado



Java App

API JDBC

Driver
MySQL

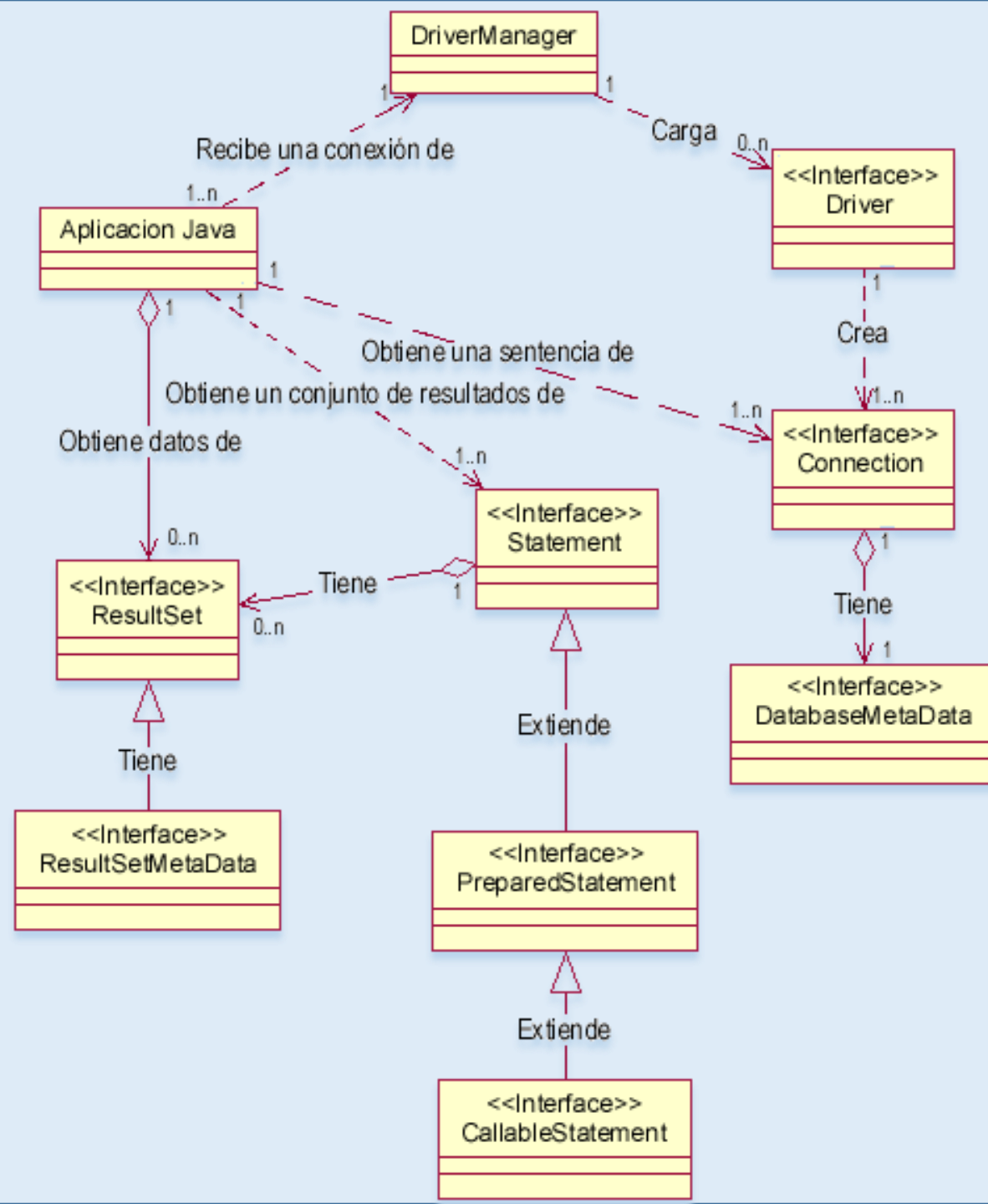
Driver
Postgresql

Driver
Oracle

Driver
SqlServer



Clase/Interface	Descripción
Driver	Permite conectarse a una base de datos: cada gestor de base de datos requiere un driver distinto
DriverManager	Permite gestionar todos los drivers instalados en el sistema
Connection	Representa una conexión con una base de datos. Una aplicación puede tener más de una conexión a más de una base de datos
DatabaseMetadata	Proporciona información acerca de una Base de Datos, como las tablas que contiene, etc.
Statement	Permite ejecutar sentencias SQL sin parámetros
PreparedStatement	Permite ejecutar sentencias SQL con parámetros de entrada
CallableStatement	Permite ejecutar sentencias SQL con parámetros de entrada y salida, típicamente procedimientos almacenados
ResultSet	Contiene las filas o registros obtenidos al ejecutar una sentencia SELECT
ResultSetMetadata	Permite obtener información sobre un ResultSet, como el número de columnas, sus nombres, etc.



Pasos para procesar sentencias JDBC

1. Descargar el Driver y agregarlo al classpath
2. Registrar el Driver JDBC
3. Establecer una conexión a la base de datos
4. Crear un objeto sentencia (Statement)
5. Ejecutar la sentencia SQL (Query)
6. Obtener y procesar el objeto ResultSet
7. Cerrar los recursos como Statement, ResultSet y Connection



Crear sentencias JDBC

```
Statement stmt = conn.createStatement();
```

```
ResultSet resultado = stmt.executeQuery("select * from productos");
```

Obtener datos de la consulta JDBC

```
Statement stmt = conn.createStatement();
ResultSet resultado = stmt.executeQuery("select * from productos");
while( resultado.next() ){
    System.out.print(resultado.getInt("id"));
    System.out.print(" | ");
    System.out.print(resultado.getString("nombre"));
    System.out.print(" | ");
    System.out.print(resultado.getDouble("precio"));
    System.out.print(" | ");
    System.out.println(resultado.getDate("fecha"));
}
resultado.close();
stmt.close();
conn.close();
```


Obtener datos usando índices

```
Statement stmt = conn.createStatement();
ResultSet resultado = stmt.executeQuery("select * from productos");
while( resultado.next() ){
    System.out.print(resultado.getInt(1));
    System.out.print(" | ");
    System.out.print(resultado.getString(2));
    System.out.print(" | ");
    System.out.print(resultado.getDouble(3));
    System.out.print(" | ");
    System.out.println(resultado.getDate(4));
}
resultado.close();
stmt.close();
conn.close();
```

Realizar operaciones DML JDBC

```
Statement stmt = conn.createStatement();  
  
int resultado = stmt.executeUpdate("INSERT INTO productos (nombre, precio, fecha) VALUES ('Bicicleta', 50, NOW())");  
  
int resultado = stmt.executeUpdate("UPDATE productos SET precio=59, nombre='Mountain Bike' WHERE id=7");  
  
int resultado = stmt.executeUpdate("DELETE FROM productos WHERE id=5");
```

Sentencias Preparadas

```
PreparedStatement stmt = conn.prepareStatement("SELECT * FROM productos WHERE id = ?");  
stmt.setInt(1, 4);
```

```
ResultSet resultado = stmt.executeQuery();
```

Obtener datos de la consulta Preparada

```
PreparedStatement stmt = conn.prepareStatement("select * from productos WHERE id=?");
stmt.setInt(1, 4);

ResultSet resultado = stmt.executeQuery();
if( resultado.next() ){
    System.out.print(resultado.getInt("id"));
    System.out.print(" | ");
    System.out.print(resultado.getString("nombre"));
    System.out.print(" | ");
    System.out.print(resultado.getDouble("precio"));
    System.out.print(" | ");
    System.out.println(resultado.getDate("fecha"));
}
resultado.close();
stmt.close();
conn.close();
```

Realizar operaciones DML con PreparedStatement

```
PreparedStatement insertStmt = conn.prepareStatement("INSERT INTO productos (nombre, precio, fecha) VALUES (?, ?, ?)");  
insertStmt.setString(1, "Bicicleta");  
insertStmt.setDouble(2, 59.99);  
insertStmt.setDate(3, new Date(System.currentTimeMillis()));  
  
insertStmt.executeUpdate();
```

Realizar operaciones DML con PreparedStatement

```
PreparedStatement updateStmt = conn.prepareStatement("UPDATE productos SET precio=?, nombre=? WHERE id=?");  
updateStmt.setDouble(1, 99.99);  
updateStmt.setString(2, "Mountain Bike");  
updateStmt.setInt(3, 7);  
  
updateStmt.executeUpdate();
```

Realizar operaciones DML con PreparedStatement

```
PreparedStatement deleteStmt = conn.prepareStatement("DELETE FROM productos WHERE id=?");  
deleteStmt.setInt(1, 5);  
  
deleteStmt.executeUpdate();
```