

Instituto Politecnico Naional Escuela Superior de Cómputo

Sistemas Distribuidos

Tarea Statements

Juan Carlos Jirón Juárez

Introducción

Un objeto Statement se usa para enviar sentencias SQL a la base de datos. Actualmente hay tres tipos de objetos Statement, todos los cuales actúan como contenedores para la ejecución de sentencias en una conexión dada: **Statement**, **PreparedStatement** que hereda de Statement y **CallableStatement** que hereda de PreparedStatement. Estas están especializadas para enviar tipos particulares de sentencias SQL.

- Un objeto Statement se usa para ejecutar una sentencia SQL simple sin parámetros.
- Un objeto PreparedStatement se usa para ejecutar sentencias SQL precompiladas con o sin parámetros IN.
- Un objeto CallableStatement se usa para ejecutar un procedimiento de base de datos almacenado.

La interfase Statement suministra métodos básicos para ejecutar sentencias y devolver resultados. La interfase PreparedStatement añade métodos para trabajar con los parámetros IN; y la interfase CallableStatement añade métodos para trabajar con parameters OUT.

Creación de objetos Statement

Una vez establecida la conexión con una base de datos particular, esta conexión puede usarse para enviar sentencias SQL. Un objeto Statement se crea mediante el método de Connection createStatement, como podemos ver en el siguiente fragmento de código.

```
Connection con = DriverManager.getConnection(url, "sunny", "");  
Statement stmt = con.createStatement();
```

La sentencia SQL que será enviada a la base de datos es alimentada como un argumento a uno de los métodos de ejecución del objeto Statement. Por ejemplo:

```
ResultSet rs = stmt.executeQuery("SELECT a, b, c FROM Table2");
```

Ejecución de sentencias usando objetos Statement.

La interfase Statement nos suministra tres métodos diferentes para ejecutar sentencias SQL, **executeQuery**, **executeUpdate** y **execute**. El método a usar esta determinado por el producto de la sentencia SQL

- El método executeQuery esta diseñado para sentencias que producen como resultado un único ResultSet tal como las sentencias SELECT.
- El método executeUpdate se usa para ejecutar sentencias INSERT, UPDATE ó DELETE así como sentencias SQL DDL (Data Definition Language) como CREATE TABLE o DROP TABLE. El efecto de una sentencia INSERT, UPDATE o DELETE es una modificación de una o más columnas en cero o más filas de una tabla. El valor devuelto de executeUpdate es un entero que indica el número de filas que han sido afectadas (referido como update count). Para sentencias tales como CREATE TABLE o DROP TABLE, que no operan sobre filas, le valor devuelto por executeUpdate es siempre cero.
- El método execute se usa para ejecutar sentencias que devuelven más de un ResultSet, más que un update count o una combinación de ambos.

Todos los métodos que ejecutan sentencias cierran los objetos Resultset abiertos como resultado de las llamadas a Statement. Esto quiere decir que es necesario completar el proceso con el actual objeto Resultset antes de reejecutar una sentencia Statement.

Debe notarse que la interfase PreparedStatement, que hereda los métodos de la interfase Statement, tiene sus propias versiones de los métodos executeQuery, executeUpdate y execute. Los objetos Statement en si mismos no contienen una sentencia SQL, por tanto debe suministrarse como un argumento a los métodos Statement.execute. Los objetos PreparedStatement no suministran una sentencia SQL como argumento a estos métodos puesto que ya tienen la sentencia precompilada. Los objetos CallableStatement heredan las formas de estos métodos de PreparedStatement. Usar un parametro de query con las versiones de los métodos de PreparedStatement o CallableStatement produciría una SQLException,.