JDBC Connection

5.0 Connection

A Connection object represents a connection with a database. A connection session includes the SQL statements that are executed and the results that are returned over that connection. A single application can have one or more connections with a single database, or it can have connections with many different databases.

A user can get information about a Connection object's database by invoking the Connection.getMetaData method. This method returns a DatabaseMetaData object that contains information about the database's tables, the SQL grammar it supports, its stored procedures, the capabilities of this connection, and so on.

5.1 Establishing Connection

The traditional way to establish a connection with a database is to call the method DriverManager.getConnection, which takes a string containing a URLwhich specifies the Host Computer Name, Port used and Oracle SID like Jdbc:oracle:thin:@HostName:Port:OracleSID

The code to connect to the database using the Thin client,

```
String url = "jdbc:oracle:thin:@teorabe:2481:OraBe";
String user = "svil_team13";
String password = "s456789!";
Connection con = DriverManager.getConnection( url, user, password );
```

The code to connect to the database using the OCI client connection specifies only the Oracle SID

```
String url = "jdbc:oracle:oci8:@OraBe";
String user = "svil_team13";
String password = "s456789!";
Connection con = DriverManager.getConnection( url, user, password );
```

5.2 Construction of JDBC URL

The three parts of a JDBC URL are broken down as follows:

jdbc - the protocol. The protocol in a JDBC URL is always jdbc.

subprotocol - the name of the driver or the name of a database connectivity mechanism,which may be supported by one or more drivers. For instance Oracle:thin:@ or oracle:oci8@ are the oracle's subprotocol is related on the driver which is used to communicate on the server.

subname - a way to identify the data source. The subname can vary, depending on the subprotocol, and it can have any internal syntax the driver writer chooses, including a subsubname. The point of a subname is to give enough information to locate the data source. A data source on a remote server requires more information, however. If the data source is to be accessed over the Internet, for example, the network address should be included in the JDBC URL as part of the subname and should adhere to the following standard URL naming convention,//hostname:port/subsubname

teorabe:2481:OraBe is the subname to connect to the oracle instance OraBe in our Test environment

teorabe - the host name or the network machine name, or its ip address, to locate the server where oracle is running.

2481 - The port number where Oracle is listening for connection

OraBe - System ID of the Oracle server database instance. SID is a required value.

5.3 Points to be considered.

If the Connection object is a pooled connection, an application should include a finally block to assure that the connection is closed even if an exception is thrown. That way a valid connection will always be put back into the pool of available connections

If a Connection object is part of a distributed transaction, an application should not call the methods Connection.commit or Connection.rollback, nor should it turn on the connection's auto-commit mode. These would interfere with the transaction manager's handling of the distributed transaction.

5.4 Reference

http://download.oracle.com/javase/tutorial/jdbc/basics/connecting.html