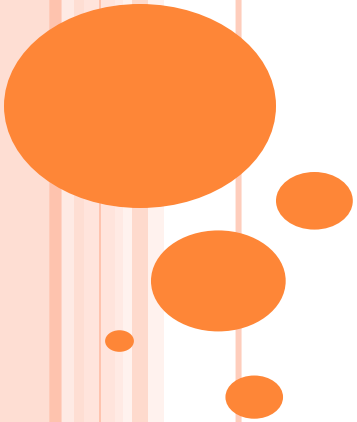
Acceso a bases de datos con JDBC



Tipos de operaciones

- **A la hora de acceder a una base de datos desde una aplicación Java mediante JDBC, se pueden realizar dos tipos de operaciones:**
  - **Operaciones de Acción: Inserción, actualización y eliminación de registros**
  - **Operaciones de selección: Recuperación de registros de una base de datos**

Antonio Martín Sierra

Operaciones de Acción

Pasos para operar contra una BD

- **El proceso para realizar operaciones de acción contra una BD es:**

1. **Carga del driver**
2. **Establecimiento de la conexión con la BD**
3. **Ejecución de la consulta SQL de acción**
4. **Cierre de la conexión**

Antonio Martín Sierra

Carga del driver

- **El driver es una librería .jar que se incluye dentro del classpath de la aplicación.**
- **Deberá ser cargado en memoria mediante:**

```
Class.forName("com.mysql.jdbc.Driver");
```

- **Desde JDBC 4 no es necesario realizar esta operación**

Antonio Martín Sierra

Establecimiento de la conexión

- **La conexión con la base de datos se establece a través del método getConnection() de DriverManager, que devuelve un objeto Connection:**

```
Connection con=DriverManager.getConnection(String cadena, String user, String pwd);
```

```
Connection con=DriverManager.getConnection(String cadena, Properties prop);
```

- **La cadena de conexión tiene el siguiente formato:**

```
jdbc:<subprotocolo>:subname
```

- **Donde *subprotocolo* es el tipo de base de datos y *subname* depende de la base de datos. Ejemplos:**

```
jdbc:mysql://localhost:3306/mydata
jdbc:oracle:thin:@localhost:1521/servicedata
jdbc:db2://localhost:50000/datasets
```

Antonio Martín Sierra

Ejecución de consulta SQL

- **Para ejecutar una consulta SQL se utilizan los objetos Statement o PreparedStatement:**

Statement

```
String sql="insert into tabla(col1,col2) values(40,'wwww')";
Statement st=con.createStatement();
st.execute(sql);
```

PreparedStatement

```
String sql="insert into tabla(col1,col2) values(?,?)";
PreparedStatement st=con.prepareStatement(sql);
st.setInt(1,40);
st.setString(2,"wwww");
st.execute();
```

Posición de los parámetros, el primero es el 1

- **Si se desea obtener el número de registros afectados puede utilizarse el método executeUpdate():**

```
String sql="Delete from tabla where campo=?";
PreparedStatement st=con.prepareStatement(sql);
st.setString(1,"myvalue");
int results=st.executeUpdate();
```

Cierre de la conexión

- **Las conexiones deben cerrarse cuando no van a ser utilizadas:**
  - **Utilizando el método *close()* de Connection:**

```
try{
    Connection con=….
    …
}
finally{
   con.close();
}
```

  - **Mediante un try con recursos:**

```
try(Connection con=…){

    …
} //se cierra automáticamente al abandonar el try
```

Ángel Martín Sierra

Operaciones de selección

Pasos para operar contra una BD

- **El proceso es el mismo que en el caso anterior, aunque se incluye un nuevo paso:**

1. **Carga del driver**
2. **Establecimiento de la conexión con la BD**
3. **Ejecución de la consulta SQL**
4. ***Manipulación de resultados***
5. **Cierre de la conexión**

Antonio Martín Sierra

Ejecución de consulta SQL

- **En el caso de una consulta de selección, se debe obtener el objeto ResultSet para acceder a los registros:**

```
String sql="select * from data";
Statement st=con.createStatement();
st.execute(sql);
ResultSet rs=st.getResultSet();
```

```
String sql="Select * from data";
Statement st=con.createStatement();
ResultSet rs=st.executeQuery(sql);
```

- **Al igual que en las consultas de acción, puede utilizarse PreparedStatement en lugar de Statement**

Antonio Martínez Sierra

Manipulación de resultados

- **Para acceder a los registros empleamos los siguientes métodos de ResultSet:**

  - **boolean next(). Se desplaza al siguiente registro, si no hay ninguno devolverá false:**

```
//recorre todas las filas
while(rs.next()){
   …
}
```

  - **xxx getXxx(int col). Métodos para obtener el valor de la columna indicada. La posición de la primera columna es la 1. xxx es el nombre del tipo Java (getInt, getString,…)**

  - **xxxgetXxx(String col). Igual que el anterior, utilizando el nombre de la columna.**

Antonio Martín Sierra