

## Programación orientada a objetos con JAVA

### Acceso a base de datos utilizando JDBC

Para utilizar los siguientes ejemplos importar los siguientes paquetes de la API de Java.

```
import java.io.*;
import java.sql.*;
```

#### Iniciar una conexión ODBC

```
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver").newInstance();
    Connection connection =
        DriverManager.getConnection("jdbc:odbc:DB_ODBC");
} catch (Exception e) {
    e.printStackTrace();
}
```

#### Mostrar registros de una tabla

```
try {
    Statement statement = connection.createStatement();
    ResultSet resultSet =
        statement.executeQuery("select * from nombre_tabla");
    while (resultSet.next()) {
        System.out.println(resultSet.getString(1));
    }
} catch (Exception e) {
    e.printStackTrace();
}
```

#### o bien

```
try {
    Statement statement = connection.createStatement();
    ResultSet resultSet =
        statement.executeQuery("select * from nombre_tabla");
    while (resultSet.next()) {
        for (int i = 1; i <=
            resultSet.getMetaData().getColumnCount() ; i++) {
            System.out.println(resultSet.getMetaData().getColumnName(i)+
                                ":" + resultSet.getString(i));
        }
    }
} catch (Exception e) {
    e.printStackTrace();
}
```

#### Crear una tabla

```
try {
    Statement statement = connection.createStatement();
    statement.execute("create table nombre_table (cadena char(25),
        numero int)");
} catch (Exception e) {
    e.printStackTrace();
}
```

#### Insertar un registro

```
try {
    Statement statement = connection.createStatement();
    statement.execute("insert into nombre_tabla values ('juan',25)");
} catch (Exception e) {
    e.printStackTrace();
}
```

**o bien**

```
try {
    PreparedStatement preparedStatement =
        connection.prepareStatement("insert into nombre_tabla values
(?,?)");
    preparedStatement.setString(1,"juan");
    preparedStatement.setInt(2,25);
    preparedStatement.executeUpdate();
} catch (SQLException e) {
    e.printStackTrace();
}
```

**Ejecutar procedimientos almacenados**

```
try {
    CallableStatement callableStatement =
        connection.prepareCall("{call nombre_procedimiento}");
    ResultSet resultSet = callableStatement.executeQuery();
    while (resultSet.next()) {
        System.out.println(resultSet.getString(1));
    }
} catch (SQLException e) {
    e.printStackTrace();
}
```

**Manejar de transacciones**

```
try {
    connection.setAutoCommit(false);
    // agregar aquí sentencias sql
    connection.commit();
    connection.setAutoCommit(true);
} catch (SQLException e) {
    connection.rollback();
    e.printStackTrace();
}
```

**Obtener información de la conexión**

```
try {
    System.out.println(connection.getMetaData().getDatabaseProductName());
} catch (SQLException e) {
    e.printStackTrace();
}
```

**Obtener información de las tablas de la base de datos**

```
try {
    ResultSet resultSet =
        connection.getMetaData().getTables(null,null,null,null);
    while (resultSet.next()) {
        for (int i = 1; i <=
            resultSet.getMetaData().getColumnCount() ; i++) {
            System.out.println(
                resultSet.getMetaData().getColumnName(i)+ ":" +
                resultSet.getString(i));
        }
    }
} catch (SQLException e) {
    e.printStackTrace();
}
```