



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA  
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)  
DOCUMENTO DEL SISTEMA DE CALIDAD**

## Formato para prácticas de laboratorio

CARRERA	PLAN DE ESTUDIO	CLAVE ASIGNATURA	NOMBRE DE LA ASIGNATURA
IC	2003-1	5046	Bases de Datos

PRÁCTICA No.	LABORATORIO DE	Bases de Datos	DURACIÓN (HORA)
12	NOMBRE DE LA PRÁCTICA	Aplicaciones Web con SQL Server	2 horas

### 1. INTRODUCCIÓN

Java ha emergido como uno de los lenguajes mas populares gracias a su independencia de plataforma. Puede usar Java para crear applets que se ejecuten en páginas Webs, escribir JSPs(Java Server Pages) o desarrollar aplicaciones solas. Java siempre utiliza JDBC(Java Database Connectivity) para conectarse a una base de datos. Si escribe el código de manera adecuada y se adhiere al estandar SQL puede ser fácil pasarlo de una plataforma a otra.

Un driver JDBC es requerido para que los programas en Java o JSP puedan acceder a informacióón de las bases de datos.

Para descargar el driver vaya a la dirección:

[http://download.microsoft.com/download/d/2/e/d2e1ffb6-2cfa-4a62-a22d-a413cce93118/Download\\_SQL\\_JDBC\\_Driver.htm](http://download.microsoft.com/download/d/2/e/d2e1ffb6-2cfa-4a62-a22d-a413cce93118/Download_SQL_JDBC_Driver.htm)

### 2. OBJETIVO (COMPETENCIA)

Realizar una conexion al servidor de bases de datos SQL Server con Java o JSP

Formuló M.C. Monceni Anabel Perez	Revisó M.C. Gloria Etelbina Chavez Valenzuela	Aprobó	Autorizó M.C. Miguel Ángel Martínez Romero
Maestro	Coordinador de la Carrera	Gestión de la Calidad	Director de la Facultad



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA  
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)  
DOCUMENTO DEL SISTEMA DE CALIDAD**

## Formato para prácticas de laboratorio

### 3. FUNDAMENTO

El Driver JDBC no es parte de Java SDK y debe establecerse en el classpath e incluir el archivo sqljdbc.jar si quiere usarlo. De lo contrario su aplicación dirá "Class not found" exception. El sqljdbc.jar esta instalado en la siguiente ubicación:

*<installation directory>*\sqljdbc\_1.0\loc\sqljdbc.jar

El siguiente es un ejemplo de CLASSPATH usado en una aplicación Windows:

**CLASSPATH =.;C:\Program Files\Microsoft SQL Server 2005 JDBC  
Driver\sqljdbc\_1.0\enu\sqljdbc.jar**

Si nota que los espacios le entre los folders le causan problemas, mueva el archivo .jar a un directorio mas sencillo.

Para conectarse a una base de datos usando el Driver Manager class, primero debe registrar el driver de la siguiente manera:

**Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");**

Una vez que el driver haya sido cargado, puede establecer una conexión usando el URL:

**Connection con DriverManager.getConnection("jdbc:sqlserver://localhost;user=  
MyUserName;password=\*\*\*\*\*");**

En el ejemplo inferior definimos primero el paquete que contendrá la clase, después se importan dos clases estandar de Tomcat. Después se define la clase como pública con el nombre *ConnectionCreator*, luego definimos el único método de la clase *getSqlServerConnection* que recibirá como parámetro una lista de valores, para devolver luego el objeto *java.sql.connection*.

Los parámetros que recibirá la clase serán database(nombre de la base de datos), servername(nombre o ip del servidor de base de datos), port(puerto de conexión, por lo general en SQL Server, 1433), username y password.



## Formato para prácticas de laboratorio

Un ejemplo completo sería:

```
package notas;
```

```
import java.sql.DriverManager;
```

```
import java.sql.Connection;
```

```
public class ConnectionCreator {
```

```
    public static java.sql.Connection  getSqlServerConnection (String database,
    String servername, int port, String username, String password) {
```

```
    try {
```

```
        Class.forName("com.microsoft.jdbc.sqlserver.SQLServerDriver");
```

```
        String url = "jdbc:microsoft:sqlserver://" + servername + ":" + port +
```

```
        ";DatabaseName=" + database + ";user=" + username + ";password=" + password;
```

```
        Connection conn = DriverManager.getConnection(url);
```

```
        if (conn != null)
```

```
            System.out.println(" --> CONECTANDO AL SERVIDOR : "+servername);
```

```
        else
```

```
            System.out.println(" --> NO ES POSIBLE CONECTARSE AL SERVIDOR : "+servername);
```

```
        return conn;
```

```
    }
```

```
    catch (Exception e) {
```

```
        System.out.println( " ERROR = " +e);
```

```
        return null;
```

```
    }
```

// Para usarlo dentro de una página JSP, anote lo siguiente:



UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA  
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)  
DOCUMENTO DEL SISTEMA DE CALIDAD

## Formato para prácticas de laboratorio

```
<%@ page language="java" %>
<%@ page import = "notas.ConnectionCreator"%>
<%@ page import = "java.sql.Connection"%>

<%
Connection miConexion = ConnectionCreator.getSqlServerConnection("BD", "127.0.0.1",1433,"sa",
"");

if (!miConexion.isClosed())
out.print("FUNCIONA !");

/* CUERPO DE LA PAGINA */

miConexion.close(); /* no olvidarse de cerrar las conexiones. */

%>
```



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA  
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)  
DOCUMENTO DEL SISTEMA DE CALIDAD**

## **Formato para prácticas de laboratorio**

### **4. PROCEDIMIENTO (DESCRIPCIÓN)**

<b>A)</b>	<b>EQUIPO NECESARIO</b>	<b>MATERIAL DE APOYO</b>
-----------	-------------------------	--------------------------

Computadoras con Java y SQL Server Instalados

Driver de conexión jdbc SQL Server

#### **B) DESARROLLO DE LA PRÁCTICA**

Haga una conexión entre Java o JSP con SQL Server a la base de datos "Ventas" creada en la práctica pasada.

#### **C) CÁLCULOS Y REPORTE**

### **5. RESULTADOS Y CONCLUSIONES**

### **6. ANEXOS**

### **7. REFERENCIAS**