

Lenguaje de programación I



Caso

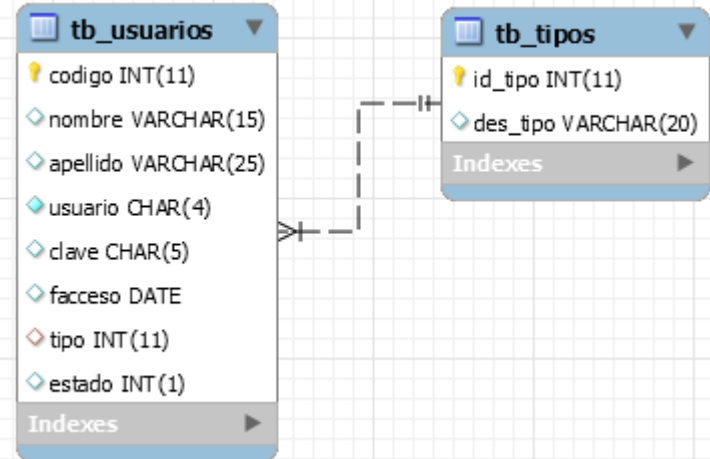
- ✓ El administrador de la empresa CiberFarma, necesita realizar una consulta de los usuarios según el tipo seleccionado
- ✓ El combo se obtiene con los datos de la tabla tipos
- ✓ La consulta debe mostrar el código del cliente, nombre, apellido y descripción del tipo.

Listado de Usuarios

Tipo: Seleccione...

- Seleccione...
- Administrador
- Cliente
- Cliente VIP

Reporte



¿Cuántos listados necesito realizar?

Contenido

- Uso de Bases de datos
- MySQL
- Ejercicio: Procedimientos almacenados
- Aplicaciones

Logros de la Unidad

- Crear aplicaciones de manejo de bases de datos



¿Qué son Procedimientos almacenados?

- ✓ Son rutinas que **encapsulan** un **conjunto de comandos SQL** almacenándolos en el **servidor**.
- ✓ Esto permite que una vez creado, los clientes no necesitan relanzar los comandos individuales, en su lugar usaremos el procedimiento.
- ✓ Utilidad:
 - Al realizar procesos con cálculos complejos o datos obtenidos de varias tablas
 - Cuando múltiples aplicaciones se escriben en distintos lenguajes o plataformas, pero necesitan realizar la misma operación.
 - Cuando la seguridad es muy importante. Debido a que las aplicaciones no obtendrían ningún acceso directo a las tablas de la base de datos, sólo pueden ejecutar algunos procedimientos almacenados.



Sintaxis en MySQL

- ✓ Para la creación del procedimiento almacenado en MySQL, tomaremos en cuenta lo siguiente:

DELIMITER \$\$

Parámetro

```
CREATE PROCEDURE USP_CONSULTAXTIPO(xtipo int)
```

```
BEGIN
```

```
SELECT codigo, nombre, apellido, des_tipo  
from tb_usuarios u inner join tb_tipos t  
on u.tipo = t.id_tipo  
where tipo = xtipo;
```

Cuerpo del
procedimiento

```
END $$
```

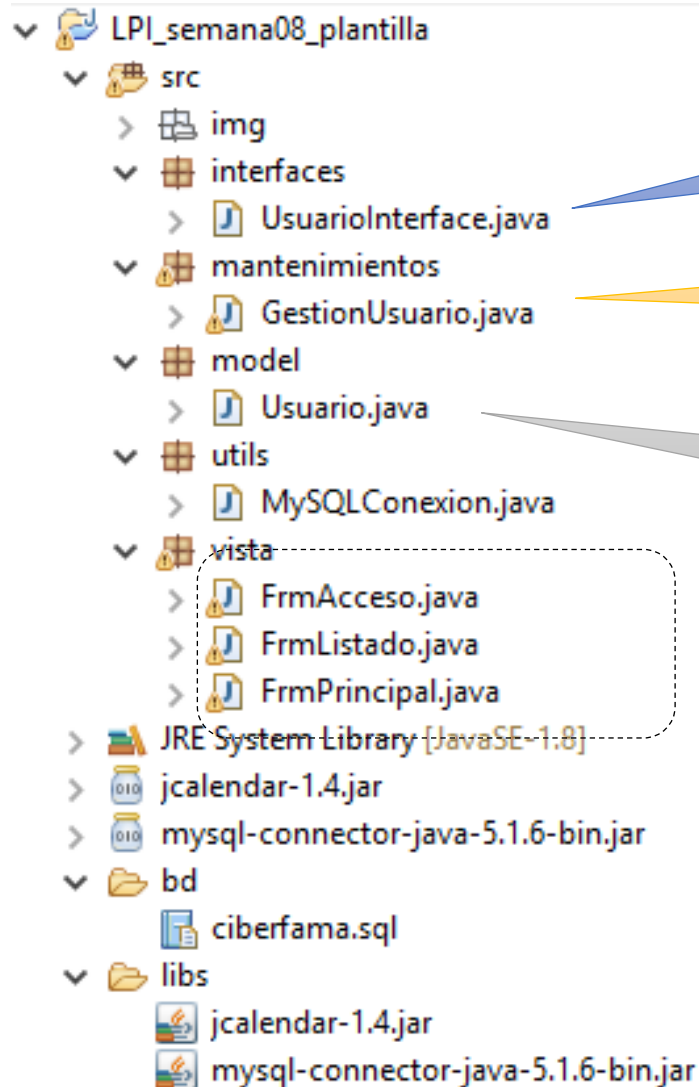
DELIMITER ;

```
CALL USP_CONSULTAXTIPO(2);
```

Ejecuta el procedimiento
pasando un parámetro

	codigo	nombre	apellido	des_tipo
▶	1	Tito	Siber	Cliente
	2	Zoila	Baca	Cliente

Estructura del Proyecto con BD



1

Listado de métodos para el mantenimiento

3

Implementación de los métodos de mantenimiento

2

Entidades utilizadas para transferir los datos desde la base de datos hacia la capa lógica y viceversa

✓ Interface?

```
public interface ConsultaInterface {  
    public List<Consulta1> usuarioxtipo(int tipo);  
}
```

✓ Clase Bean?

```
public class Consulta1 {  
    private int codigo;  
    private String nombre, apellido, destipo;
```

✓ Clase Gestión?

```
@Override  
public List<Consulta1> usuarioxtipo(int tipo) {  
    List<Consulta1> lista = new ArrayList<Consulta1>();  
    ResultSet rs = null;  
    Connection con = null;  
    PreparedStatement pst = null;  
    try {  
        con = MySQLConexion.getConnection();  
        String sql = "{call usp_consultaxtipo(?)}";  
        pst = con.prepareStatement(sql);  
        pst.setInt(1, tipo);
```

Solución Caso



- ✓ El administrador de la empresa CiberFarma, necesita realizar una consulta de los usuarios según el tipo seleccionado
- ✓ El combo se obtiene con los datos de la tabla tipos
- ✓ La consulta debe mostrar el código del cliente, nombre, apellido y descripción del tipo.

Listado de Usuarios

Tipo: Seleccione...

Administrador
Cliente
Cliente VIP

Reporte

tb_usuarios	
codigo	INT(11)
nombre	VARCHAR(15)
apellido	VARCHAR(25)
usuario	CHAR(4)
clave	CHAR(5)
facceso	DATE
tipo	INT(11)
estado	INT(1)
Indexes	

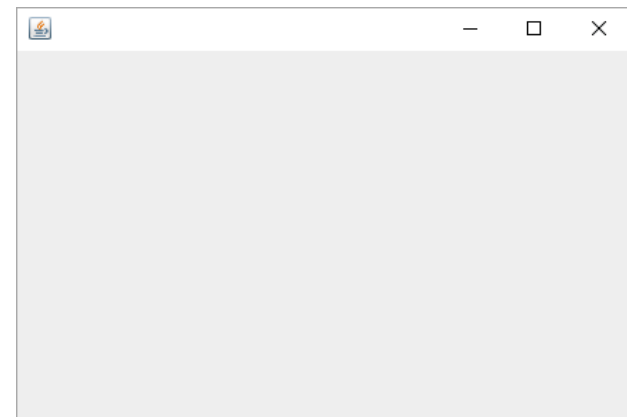
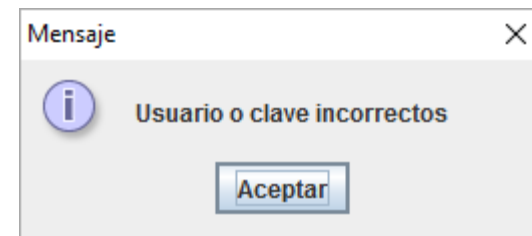
tb_tipos	
id_tipo	INT(11)
des_tipo	VARCHAR(20)
Indexes	



Caso 2



- ✓ Validar el acceso de los usuarios al sistema, según el nombre de usuario y su clave de acceso, si existe, se devuelven los datos del Usuario y se carga el formulario principal, sino se muestra un mensaje de error.

A screenshot of a login window titled "CIBERFARMA - Acceso al Sistema". On the left is a cartoon character of a male scientist in a white lab coat holding a green flask. To the right of the character are two input fields: "Usuario:" and "Clave:". Below these fields are two buttons: "Ingresar" and "Salir". The window has standard Windows-style controls (minimize, maximize, close) in the top right corner.

Sintaxis en MySQL

- ✓ El procedimiento **realiza una consulta** a la tabla `tb_usuarios`, según el usuario y clave pasados como parámetro.

-- sentencia

DELIMITER \$\$

CREATE PROCEDURE usp_validaUsuario(usr **char**(4), pas **char**(5))

BEGIN

select * **from** tb_usuarios

where usuario = usr **and** clave = pas;

END \$\$

DELIMITER ;

-- llamado

CALL usp_validaUsuario ('U001', '10001');

✓ Interface Usuario?

```
public Usuario validaAcceso(String usuario, String clave);
```

✓ Clase Gestión?

```
public Usuario validaAcceso(String usuario, String clave) {
```

```
    Usuario u = null;
```

```
    ResultSet rs = null; // tipo de resultado
```

```
    Connection con = null;
```

```
    PreparedStatement pst = null;
```

```
    try {
```

```
        con = MySQLConexion.getConnection();
```

```
        String sql = "{call usp_validaAcceso (?,?)}"; // sentencia sql
```

```
        pst = con.prepareStatement(sql);
```

```
        // parámetros según la sentencia
```

```
        pst.setString(1, usuario);
```

```
        pst.setString(2, clave);
```

```
        rs = pst.executeQuery(); // tipo de ejecu
```

```
        // Acciones adicionales en caso de consultas
```

```
        while (rs.next()) {
```

```
            u = new Usuario();
```

```
            u.setCodigo(rs.getInt(1));
```

```
            u.setNombre(rs.getString(2));
```

```
88 • CALL usp_validaAcceso ('U001','10001');
```

result Grid | Filter Rows: | Export: | Wrap Cell Content:

codigo	nombre	apellido	usuario	clave	facceso	tipo	estado
1	Tito	Siber	U001	10001	2017-05-27	2	1

Botón



```
void validaAcceso() {  
    // variables  
    String usuario, clave;  
  
    // entradas  
    usuario = leerUsuario();  
    clave = leerClave();  
  
    // procesos  
    GestionUsuario gu = new GestionUsuario();  
    Usuario u = gu.validaAcceso(usuario, clave);  
  
    // salidas  
    if (u == null) {  
        aviso("Usuario o clave incorrectos");  
    } else {  
        aviso("Bienvenido " + u.getNombre() + " " + u.getApellido());  
        FrmPrincipal p = new FrmPrincipal();  
        p.setVisible(true);  
        p.setExtendedState(MAXIMIZED_BOTH);  
        dispose();  
    }  
}
```

Referencias

- ✓ <http://programandoointentandolo.com/2013/11/como-ejecutar-un-procedimiento-almacenado-desde-java-con-jdbc.html>
- ✓ <http://pabletoreto.blogspot.pe/2013/03/procedimientos-almacenados-en-mysql-java.html>

GRACIAS



SEDE MIRAFLORES

Calle Díez Canseco Cdra 2 / Pasaje Tello
Miraflores – Lima
Teléfono: 633-5555

SEDE INDEPENDENCIA

Av. Carlos Izaguirre 233
Independencia – Lima
Teléfono: 633-5555

SEDE BREÑA

Av. Brasil 714 – 792
(CC La Rambla – Piso 3)
Breña – Lima
Teléfono: 633-5555

SEDE TRUJILLO

Calle Borgoño 361
Trujillo
Teléfono: (044) 60-2000

SEDE SAN JUAN DE LURIGANCHO

Av. Próceres de la Independencia 3023-3043
San Juan de Lurigancho – Lima
Teléfono: 633-5555

SEDE SAN MIGUEL

Av. Federico Gallese 847
San Miguel – Lima
Teléfono: 632-4900

SEDE BELLAVISTA

Av. Mariscal Oscar R. Benvides 3866 – 4070
(CC Mall Aventura Plaza)
Bellavista – Callao
Teléfono: 633-5555

SEDE AREQUIPA

Av. Porongoche 500
(CC Mall Aventura Plaza)
Paucarpata - Arequipa
Teléfono: (054) 60-3535