

EJECUCIÓN DE PROCEDIMIENTOS Y FUNCIONES.

Javier García-Retamero Redondo

CONCEPTO

- Un procedimiento o una función es un conjunto de sentencias SQL que se puede llamar por su nombre para llevar a cabo alguna tarea.
- Normalmente reciben parámetros para ser más versátiles.
- Las funciones siempre devuelven un dato.
- Los procedimientos normalmente sólo reciben parámetros de entrada pero también pueden tener parámetros de salida.

CONSIDERACIONES

- **En la BBD, el usuario debe tener permisos para ejecutar procedimientos y funciones:**

Desde el usuario System ejecutar: GRANT EXECUTE PROCEDURE TO usuario;

- **Para obtener los valores devueltos por una función o procedimiento:**

getXXX(número_parámetro) como hacíamos con los ResultSet

```
String salida1 = llamada.getString(1);  
String salida2 = llamada.getString(3);
```

EJECUCIÓN DE PROCEDIMIENTOS

Realización de llamadas



EJECUCIÓN DE PROCEDIMIENTOS

- Declaración de llamadas a un procedimiento o función:

Declaración	Para llamar a
{call nombre_procedimiento}	Procedimiento almacenado sin parámetros
{call nombre_procedimiento(?,?,...)}	Procedimiento almacenado que recibe parámetros o que devuelve valores a través de los mismos.

EJECUCIÓN DE PROCEDIMIENTOS

- **Preparamos el string para la llamada:**

```
String sql = "{call p_nombre_depart (?, ?)}";
```

- **Creamos un objeto llamando al método `prepareCall`:**

```
CallableStatement llamada=conexion.prepareCall(sql);
```

- **Le damos valor al parámetro de entrada:**

```
int dep = 1;  
llamada.setInt(1, dep);
```

- **Registramos los parámetros de salida (OUT) del procedimiento:**

```
llamada.registerOutParameter(2, Types.VARCHAR);
```

```
CREATE OR REPLACE PROCEDURE p_nombre_depart(  
    ndepart IN NUMBER,  
    nombre_depart OUT VARCHAR2)  
AS  
BEGIN  
    SELECT dnombre INTO nombre_depart  
    FROM departamentos  
    WHERE dept_no = ndepart;  
END;
```

BBDD

EJECUCIÓN DE PROCEDIMIENTOS

- **Realizamos la llamada al procedimiento:**
`llamada.executeUpdate();`
- **Recogemos el valor devuelto a través de parámetros:**
`String salida_return = llamada.getString(2);`

EJECUCIÓN DE FUNCIONES

Realización de llamadas



EJECUCIÓN DE FUNCIONES

- Declaración de llamadas a un procedimiento o función:

Declaración	Para llamar a
{?=call nombre_procedimiento}	Función almacenada que devuelve un valor y no recibe parámetros
{?=call nombre_procedimiento(?,?,...)}	Función almacenada que devuelve un valor y recibe varios parámetros

EJECUCIÓN DE FUNCIONES

- **Preparamos el string para la llamada:**

```
String sql = "{?=call f_nombre_depart (1)}";
```

- **Creamos un objeto llamando al método prepareCall:**

```
CallableStatement llamada=conexion.prepareCall(sql);
```

- **Registramos los parámetros de salida (OUT) de la función:**

```
llamada.registerOutParameter(1, Types.VARCHAR);
```

- **Le damos valor al parámetro de entrada:**

```
int dep = 1;  
llamada.setInt(2, dep);
```

```
CREATE OR REPLACE FUNCTION f_nombre_depart(  
    ndepart NUMBER) RETURN VARCHAR2  
IS  
    nombre_depart departamentos.dnombre%TYPE;  
BEGIN  
    SELECT dnombre INTO nombre_depart FROM  
    departamentos WHERE dept_no = ndepart;  
    RETURN (nombre_depart);  
END;
```

BBDD

EJECUCIÓN DE FUNCIONES

- **Realizamos la llamada a la función:**
`llamada.executeUpdate();`
- **Recogemos el valor devuelto a través de parámetros:**
`String salida_return = llamada.getString(1);`