



Procedimientos Almacenados

Valeria Beratto U.



Procedimientos Almacenados

- Un procedimiento es un **conjunto de sentencias SQL y de control de flujo**.
- Un procedimiento almacenado es una colección de instrucciones **guardadas** en la Base de Datos.
- La ventaja de un procedimiento almacenado es que al ser ejecutado, éste se ejecuta directamente en el motor de bases de datos, el cual usualmente corre en un servidor separado.



Beneficios

- **Simplifican la ejecución** de tareas repetitivas
- **Corren más rápido** que las mismas instrucciones ejecutadas en forma interactiva.
- **Reducen** el tráfico a través de la red
- **Pueden capturar** errores antes que ellos puedan entrar a la base de datos.
- Establece **consistencia** porque ejecuta las tareas de la misma forma.
- Permite el **desarrollo modular** de aplicaciones
- Ayuda a proveer **seguridad**.
- Puede **forzar reglas y defaults** complejos de los negocios

Sintaxis de Creación

```
CREATE [OR REPLACE]
PROCEDURE <procedure_name>
[(<param1> [IN|OUT|IN OUT]
<type>, <param2> [IN|OUT|IN OUT]
<type>, ...)]

IS
-- Declaración de variables
locales

BEGIN
-- Sentencias

[EXCEPTION]
-- Sentencias control de
excepción

END;
```

- La sintaxis es muy parecida a la de un bloque, salvo porque se reemplaza la sección **DECLARE** por la secuencia **PROCEDURE ... IS** en la especificación del procedimiento.
- Debemos especificar el tipo de datos de cada parámetro. **Al especificar el tipo de dato del parámetro NO debemos especificar la longitud del tipo.**
- Los parámetros pueden ser de entrada (**IN**), de salida (**OUT**) o de entrada salida (**IN OUT**). El valor por defecto es **IN**, y se toma ese valor en caso de que no lo especifiquemos.

Sintaxis Ejecución y Borrar SP

- `Exec <nombre_sp>`
- `Execute <nombre_sp>`
- En un bloque

```
BEGIN

    <nombre_sp>( parámetros);

END;
```
- Para borrar

```
DROP <nombre_sp>
```

Ejemplo

Esquema: Cuenta
(co_cuenta, saldo,
fx_actualización)

```
CREATE OR REPLACE  
PROCEDURE Actualiza_Saldo(cuenta NUMBER,  
giro NUMBER)  
IS  
-- Declaración de variables locales  
BEGIN  
    UPDATE SALDOS_CUENTAS  
    SET SALDO = SALDO - giro,  
    FX_ACTUALIZACION = SYSDATE  
    WHERE CO_CUENTA = cuenta;  
  
END;
```

Ejercicios

Empleado (cod_emp, nombre, salario, ano_ing, sexo, cod_dep)

Departamento (cod_depto, nombre_depto).

Cree un procedimiento que permita actualizar el salario de los empleados que ganan menos de salario mínimo.- El valor porcentual de modificación será de 15% y valor del salario mínimo se debe ingresar como parámetro.

```
CREATE OR REPLACE
PROCEDURE Actualiza_Salario(sal_min number)
IS

BEGIN
    UPDATE empleado
    SET SALARIO = (SALARIO*1.15)
    WHERE SALARIO <= sal_min;

END;

EXECUTE ACTUALIZA_SALARIO(300);
```

Ejercicios

Cree un procedimiento almacenado que imprima por pantalla los años trabajados en la empresa de un empleado. Deberá recibir por parámetro el código de empleado.

```
CREATE OR REPLACE PROCEDURE SP_CANT_ANO ( C_EMP INTEGER)
IS
ANO INTEGER;
CANT_AN INTEGER;
BEGIN
    SELECT ANO_ING INTO ANO FROM EMPLEADO WHERE
    COD_EMP=C_EMP;
    CANT_AN := EXTRACT(YEAR FROM sysdate) - ANO;
    DBMS_OUTPUT.PUT_LINE('LA CANTIDAD DE AÑOS ES ' ||
    CANT_AN );
END;
```


Ejercicios

Cree un procedimiento almacenado que retorne los años trabajados en la empresa de un empleado. Deberá recibir por parámetro el código de empleado.

```
CREATE OR REPLACE PROCEDURE SP_CANT_ANO ( C_EMP INTEGER,  
CANT_AN OUT INTEGER ) IS  
ANO INTEGER;  
BEGIN  
    SELECT ANO_ING INTO ANO FROM EMPLEADO WHERE  
    COD_EMP=C_EMP;  
    CANT_AN := EXTRACT(YEAR FROM sysdate) - ANO;  
END;
```

```
DECLARE  
CANTIDAD INTEGER;  
BEGIN  
    SP_CANT_ANO(1, cantidad);  
    DBMS_OUTPUT.PUT_LINE('LA CANTIDAD DE AÑOS ES ' ||  
    CANTIDAD);  
END;
```

Ejercicios

- Cree un procedimiento almacenado que permita ingresar un registro en la tabla departamento, debe recibir por parámetro código y nombre.

```
CREATE OR REPLACE PROCEDURE SP_INGRESO_DEPTO (COD
DEPARTAMENTO.COD_DEPTO%TYPE, NOMBRE
DEPARTAMENTO.NOMBRE_DEPTO%TYPE ) IS

BEGIN
    INSERT INTO DEPARTAMENTO VALUES (COD,NOMBRE);
END;
```

```
EXECUTE SP_INGRESO_DEPTO (41, 'DDD');
```

```
BEGIN
    SP_INGRESO_DEPTO (41, 'CIENCIAS');
END;
```

Ejercicios para Resolver

- Cree un procedimiento almacenado que permita generar una estadísticas de los empleados, considerando la cantidad de mujeres y hombres por cada departamento
- Cree un procedimiento almacenado que imprima por pantalla la cantidad en dinero que debe desembolsar cada departamento y la empresa por el concepto de salarios.
- Cree un procedimiento almacenado que permita calcular el total a pagar a las AFP por conceptos de imposiciones, considerando que es el 10% de los salarios.