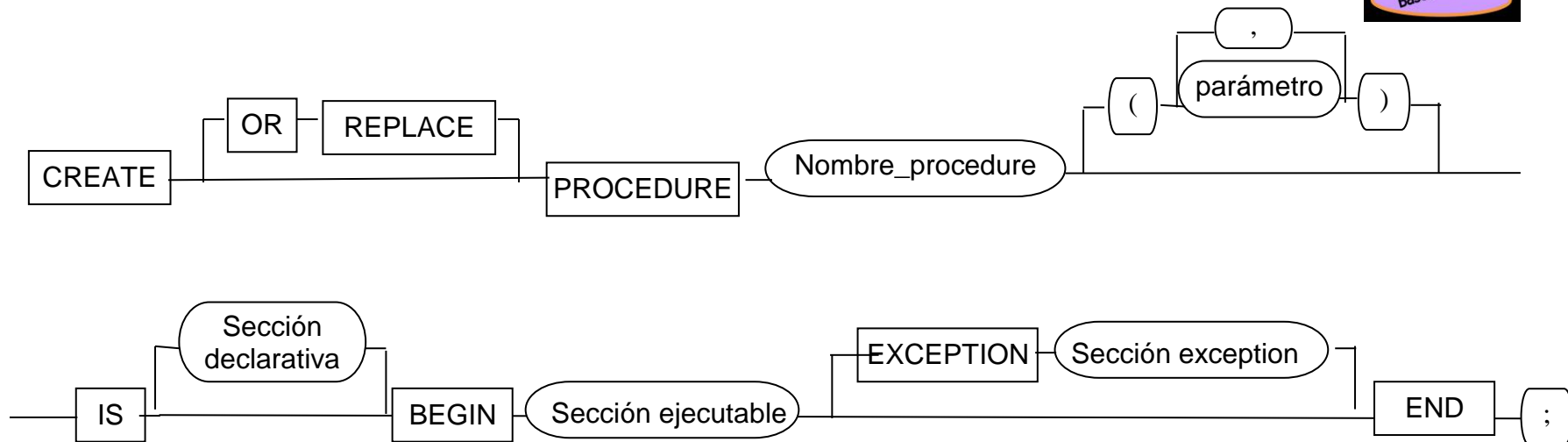


Sesión 6: CREATE PROCEDURE



Al definir los parámetros se debe poner: *nombre_parámetro tipo_de_parámetro tipo_de_datos(Sin longitud)*

Como tipo de parámetro puede ser:

- IN(por defecto) pudiendo pasarle valores en la llamada al procedimiento. Este valor no se puede modificar. Se le puede asignar un valor por defecto.

CREATE PROCEDURE EJEMPLO(entrada in Lumber default 10) IS ...

- OUT para devolver valores.
- IN OUT que permite pasarle valores en la llamada al procedimiento y luego devolver valores.

Para borrar un procedimiento **DROP PROCEDURE** nombre_procedure

EJEMPLOS

create or replace procedure verminsuperficie is

```
CURSOR c1 IS SELECT numero, superficie, supMin FROM habitacion , categoria WHERE
categoria=nombre;
```

BEGIN

```
FOR regc1 IN c1 LOOP
```

```
  IF regc1.superficie < regc1.supMin THEN
```

```
    escribir(' Habitación '|| regc1.numero || ' dimensión INCORRECTA ***');
```

```
    INSERT INTO MALCATEGORIA
```

```
      VALUES(regc1.numero, regc1.supMin-regc1.superficie);
```

```
  ELSE
```

```
    escribir(' Habitación '|| regc1.numero || ' dimensión correcta');
```

```
  END IF;
```

```
END LOOP;
```

END;

exec verminsuperficie;

```
create or replace procedure escribir(auxcad in varchar2) is
begin
dbms_output.enable;
dbms_output.put_line(auxcad);
end;
```

Se puede ejecutar directamente **exec** escribir('Hola')

O bien desde dentro de otro procedimiento

```
create or replace
procedure obtenerprecio_tipo(nomcateg categoria.nombre%type, posiblereserva date) is
auxsa pvptemporada.psa%type; auxad pvptemporada.pad%type;
auxmp pvptemporada.pmp%type; auxpc pvptemporada.ppc%type;

begin
select pSA, pAD, pMP, pPC into auxsa, auxad, auxmp, auxpc
from pvptemporada p , calendario ca
where fecha=posiblereserva and p.temporada =ca.temporada and p.categoria=nomcateg;
escribir('El precio para SOLO ALOJAMIENTO en esa fecha es '||auxsa);
escribir('El precio para ALOJAMIENTO Y DESAYUNO en esa fecha es '||auxad);
escribir('El precio para MEDIA PENSIÓN en esa fecha es '||auxmp);
escribir('El precio para PENSIÓN COMPLETA en esa fecha es '||auxpc);
end;
```

exec obtenerprecio_tipo('D', '16/08/2013');

```
create or replace  
procedure obtenerprecio(posiblereserva date) is  
cursor las_categorias is select nombre from categoria;  
begin  
FOR cat in las_categorias LOOP  
Escribir('** Precio para ' || cat.nombre || es ' **);  
obtenerprecio_tipo(cat.nombre, posiblereserva);  
END LOOP;  
end;
```

```
exec obtenerprecio('16/08/2013');
```

Al crear un procedure, una función o un trigger(en próximas sesiones) con SQL-Developer, os situáis sobre él y seleccionáis la opción COMPILE. **Si se os indica que se ha creado con errores de compilación debéis corregirlos.**

