

# PLSQL

Álvaro González Sotillo

30 de marzo de 2018

## Índice

1. Introducción	1
2. Bloques anónimos	2
3. Variables	2
4. Control de flujo	3
5. Sentencias SQL	4
6. Referencias	5

## 1. Introducción

### 1.1. Palabras reservadas

- Vista V\$RESERVED\_WORDS
- Definen estructuras de programa
- No pueden ser usados como identificadores

### 1.2. Identificadores

- Nombres definidos por el programador
  - No puede ser una palabra reservada
  - Constante, variable, excepción, paquete, función, procedimiento, tabla, cursor...
  - Hasta 30 caracteres
  - Comienza por una letra.
  - Puede contener \$, #, pero no puede contener operadores + % = / \*

---

## 2. Bloques anónimos

```
select * from pepe where nombre='a';
SET SERVEROUTPUT ON;

begin
    dbms_output.put_line('Hola');
END;
/
```

Listing 1: Bloque anónimo

## 3. Variables

- Valores referenciados por un identificador
- Deben declararse al principio de los bloques

```
SET SERVEROUTPUT ON;

DECLARE
    msg varchar(255);
BEGIN
    msg := 'Hola';
    dbms_output.put_line(msg);
END;
/
```

### 3.1. Tipos de variable

- Se pueden utilizar todos los tipos SQL
  - char, varchar
  - number, integer, float
  - date, timestamp
  - blob, clob
- Tipos propios de PLSQL
  - bool
  - pls\_integer

### 3.2. Tipos referidos

- %type : Tipo de un campo de una tabla
- %rowtype : Tipo compuesto, referido a una fila de una tabla

---

```

create table cliente( id integer, nombre varchar(255) );

DECLARE
    filacliente cliente%rowtype;
BEGIN
    filacliente.id := 1;
    filacliente.nombre := 'María';
    insert into cliente values filacliente;
END;
/

```

## 4. Control de flujo

### 4.1. Condicional

```

DECLARE
    numero integer := 1;
BEGIN
    if( numero < 0 ) then
        dbms_output.put_line( "Menor que cero");
    elsif( numero > 0 ) then
        dbms_output.put_line( "Mayor que cero");
    else
        dbms_output.put_line( "Igual que cero");
    end if;
END;
/

```

### 4.2. Condicional múltiple (I)

```

case
    when vsalario<0 then
        dbms_output.put_line('Incorrecto');
    when vsalario=0 THEN
        dbms_output.put_line('Gratis!');
    when vsalario<10000 then
        dbms_output.put_line('Salado!');
    when vsalario<90000 then
        dbms_output.put_line('Mas o menos');
    else
        dbms_output.put_line('Correcto');
end case;

```

### 4.3. Condicional múltiple (II)

```

case v_job_grade
    when 1 THEN
        dbms_output.put_line('Jefe!');
    when 2 then
        dbms_output.put_line('Jefecito');
    when 3 then
        dbms_output.put_line('Empleado regular');
    ELSE
        dbms_output.put_line('CEO');
end case;

```

---

## 4.4. Bucle loop

```
LOOP
  -- Instrucciones
  IF (expresion) THEN
    -- Instrucciones
  EXIT;
END IF;
END LOOP;
```

## 4.5. Bucle while

```
WHILE (expresion) LOOP
  -- Instrucciones
END LOOP;
```

## 4.6. Bucle for

```
DECLARE
  c PLS_INTEGER DEFAULT 0;
BEGIN
  FOR c IN REVERSE 1..10 LOOP
    dbms_output.put_line ('Contador = '||c);
  END LOOP;
END;
```

# 5. Sentencias SQL

## 5.1. Variables en select

```
create table empleados( empno number(20), salario number(8,2), nombre varchar(255));
DECLARE
  vsalario NUMBER;
BEGIN
  SELECT salario INTO vsalario FROM empleados WHERE empno=7369;
  dbms_output.put_line('El empleado numero 7369 tiene un vsalario de '||vsalario||' ');
end;
/
```

## 5.2. Variables en insert, update, delete

- Se utilizan como un valor inmediato

```
declare
  vempno number;
begin
  vempno := 100;
  insert into empleados(empno, salario, nombre)
    values( vempno, 1000, 'Juan');
  update empleados
    set salario = salario + 100
    where empno = vempno;
  delete from empleados where empno = vempno;
end;
/
```

---

### 5.3. Recorrer consultas

```
DECLARE
  c empleados %ROWTYPE;
  salariototal number;
  numeroempleados number;
  mediasalario number;
begin
  numeroempleados := 0;
  for c in (select * from empleados) loop
    dbms_output.put_line(c.nombre);
    numeroempleados := numeroempleados + 1;
    salariototal := salariototal + c.salario;
  end loop;
  mediasalario := salariototal / numeroempleados;
end;
/
```

## 6. Referencias

- Formatos:
  - [Transparencias](#)
  - [PDF](#)
- Creado con:
  - [Emacs](#)
  - [org-reveal](#)
  - [Latex](#)