

Explicación Cursores

Algunos ejemplos que vemos en clase:

```
drop table prueba;
create table prueba (
    DNI number(8),
    Nombre Varchar2(20));
insert into prueba values (12345671,'Pepi');
insert into prueba values (12345672,'Pep');
insert into prueba values (12345673,'Pe');
insert into prueba values (12345674,'P');
insert into prueba values (12345675,'juan');
insert into prueba values (12345676,'joan');
insert into prueba values (12345677,'jaime');
--select * from prueba;

set serveroutput on;
declare
varDNI prueba.DNI%type;
varNombre prueba.Nombre%type;
cursor cur1 is
    select DNI,Nombre from prueba; --Declaracion del Cursor
begin
    open cur1;                                --Apertura del Cursor
    fetch cur1 into varDNI,varNombre;
    while cur1%FOUND loop
        dbms_output.put_line(varDNI||' '||varNombre);
        fetch cur1 into varDNI,varNombre; --Recoger
    end loop;
    close cur1;                               --Cerrar Cursor
end;
/
--Cursos con bucle While
declare
varreg prueba%rowtype;
cursor cur1 is
    select * from prueba;                  --Declaracion del Cursor
begin
    open cur1;                                --Apertura del Cursor
    fetch cur1 into varreg;
    while cur1%FOUND loop
        dbms_output.put_line(varreg.DNI||' '||varreg.Nombre);
        fetch cur1 into varreg;                --Recoger
    end loop;
    close cur1;                               --Cerrar Cursor
end;
/
```

```
--Cursos con bucle loop
declare
varreg prueba%rowtype;
cursor cur1 is
    select * from prueba;          --Declaracion del Cursor
begin
    open cur1;                  --Apertura del Cursor
    loop
        fetch cur1 into varreg;   --Recoger
        exit when cur1%notfound;
        dbms_output.put_line(varreg.DNI||' '||varreg.Nombre);
    end loop;                   --Cerrar Cursor
end;
/

```

```
--Cursos con bucle For
declare
cursor cur1 is
    select * from prueba;          --Declaracion del Cursor
begin
    for varreg in cur1 loop
        dbms_output.put_line(varreg.DNI||' '||varreg.Nombre);
    end loop;
end;
/

```

-- Cursos con variables de acoplamiento

```
CREATE OR REPLACE PROCEDURE ver_emple_en_dept (dep number) AS
    v_dept NUMBER(2);
    CURSOR c1 IS SELECT last_name FROM employees WHERE department_id = v_dept;
    v_apellido VARCHAR2(10);
BEGIN
    v_dept := dep;
    OPEN c1;
    FETCH c1 INTO v_apellido;
    WHILE c1%FOUND LOOP
        DBMS_OUTPUT.PUT_LINE(v_apellido);
        FETCH c1 INTO v_apellido;
    END LOOP;
    CLOSE c1;
END;
```

```
CREATE OR REPLACE PROCEDURE ver_emple_en_dept (dep number) AS
    v_dept NUMBER(2);
    CURSOR c1 IS SELECT last_name FROM employees WHERE department_id = dep;
BEGIN
    for varreg in c1 loop
        DBMS_OUTPUT.PUT_LINE(varreg.last_name);
    END LOOP;
END;
```

--Cursos con parametros

```
CREATE OR REPLACE PROCEDURE ver_emple_por_dept AS
  CURSOR c1 IS SELECT * FROM departments
  CURSOR c2(dpto number) IS SELECT * FROM employees department_id = dpto;
BEGIN
  for r1 in c1 loop
    DBMS_OUTPUT.PUT_LINE(r1.department_id);
    for r2 in c2(r1.department_id) loop
      DBMS_OUTPUT.PUT_LINE(r2.last_name);
    end loop;
  END LOOP;
END;
```

-- Cursos para actualización

```
CREATE OR REPLACE PROCEDURE subir_salario_dpto(vp_num_dpto NUMBER, vp_pct_subida
NUMBER) AS
  CURSOR c_emple IS SELECT oficio, salario FROM emple WHERE dept_no = vp_num_dpto
FOR UPDATE;
  vc_reg_emple c_emple%ROWTYPE;
  v_inc NUMBER(8,2);
BEGIN
  OPEN c_emple;
  FETCH c_emple INTO vc_reg_emple;
  WHILE c_emple%FOUND LOOP
    v_inc := (vc_reg_emple.salario / 100) *
    vp_pct_subida;
    UPDATE emple SET salario = salario + v_inc WHERE CURRENT OF c_emple; -- (al
actual)
    FETCH c_emple INTO vc_reg_emple;
  END LOOP;
END subir_salario_dpto;
```

```
CREATE OR REPLACE PROCEDURE subir_salario_dpto(num NUMBER, pct_subida
NUMBER, maximo number) AS
```

```
  CURSOR c1 IS SELECT * FROM employees WHERE dept_no = num FOR UPDATE;
BEGIN
  for r1 in c1 loop
    if (r1.salary*pct_subida <= maximo) then
      UPDATE employees SET salary = salary+salary*pct_subida WHERE
CURRENT OF c_emple; -- (al actual)
      end if;
    END LOOP;
END subir_salario_dpto;
```

Explicaciones alternativas al tema y videos en:
<https://elbauldelprogramador.com/plsql-cursos/>