/\*\*\*

CURSORS: S'utilitzen per consultes que retornen més d'una fila

Es fa un recorregut i s'accedeix fila a fila

DECLARE CURSOR nom\_cursor FOR SELECT...

OPEN nom\_cursor

PER CADA FILA

FETCH nom\_cursor INTO...

FPER

CLOSE nom\_cursor

\*\*\*/

**Exercici1.** Mitjançant un cursor fes un llistat dels empleats que cobren més de 6000, ordenat per salari.

**Utilitza l’estructura iterativa LOOP.**

/\*\*\*EXERCICI 1. AMB LOOP\*\*\*/

DO $$

DECLARE

--Declarar el cursor (qualsevol SELECT és acceptada)

cur\_employees CURSOR FOR

SELECT employee\_id, first\_name, salary

FROM employees

WHERE salary > 6000

ORDER BY salary;

v\_employee\_id EMPLOYEES.EMPLOYEE\_ID%TYPE;

v\_first\_name EMPLOYEES.FIRST\_NAME%TYPE;

v\_salary EMPLOYEES.SALARY%TYPE;

BEGIN

--Obrir cursor. Es reserva memoria i el posiciona a la primera fila

OPEN cur\_employees;

LOOP

--Carrega les dades en les variables i es posiciona en la següent fila

FETCH cur\_employees INTO v\_employee\_id, v\_first\_name, v\_salary;

--Per detectar fi de cursor

EXIT WHEN NOT FOUND;

RAISE NOTICE ‘% % %’, v\_employee\_id ,v\_first\_name ,v\_salary;

END LOOP;

--Allibera recursos

CLOSE cur\_employees;

END $$;

**Exercici2.** Mitjançant un cursor fes un llistat dels empleats que cobren més de 6000, ordenat per salari.

**Utilitza l’estructura iterativa FOR.**

/\*\*\*EXERCICI 2. AMB FOR. En aquest cas OBRIR/FETCH i TANCAR cursor es fa automatic\*\*\*/

DO $$

DECLARE

--Declarar el cursor (qualsevol SELECT és acceptada)

cur\_employees CURSOR FOR

SELECT employee\_id, first\_name, salary

FROM employees

WHERE salary > 6000

ORDER BY salary;

BEGIN

FOR v\_curEmployees\_fila IN cur\_employees LOOP

RAISE NOTICE ‘% % % ‘,v\_curEmployees\_fila.employee\_id ,v\_curEmployees\_fila.first\_name , v\_curEmployees\_fila.salary;

END LOOP;

--Acabat el FOR el cursor s'ha tancat

END $$;

**Exercici3.** Mitjançant un cursor fes un llistat dels empleats que cobren més de 6000, ordenat per salari.

**Utilitza l’estructura iterativa FOR sense declarar cap cursor.**

/\*\*\*EXERCICI 3. AMB FOR més simplificat encara\*\*\*/

DO $$

DECLARE

fila\_Empleat record;

BEGIN

FOR fila\_Empleat IN (SELECT employee\_id, first\_name, salary

FROM employees

WHERE salary > 6000

ORDER BY salary) LOOP

RAISE NOTICE '% % %' , fila\_Empleat.employee\_id, fila\_Empleat.first\_name, fila\_Empleat.salary;

END LOOP;

END $$;

**Exercici4.** Mitjançant un cursor fes un llistat dels empleats que cobren més de X (on X és un valor que informarà el programador mitjançant un paràmetre), ordenat per salari.

**Utilitza l’estructura iterativa LOOP i fes servir un paràmetre en la variable cursor per indicar la quantitat amb la que comparar el salari.**

/\*\*\*EXERCICI 4. CURSOR AMB PARAMETRES\*\*\*/

do $$

DECLARE

cur\_employees CURSOR(p\_salary EMPLOYEES.SALARY%TYPE) FOR

SELECT employee\_id, first\_name, salary

FROM employees

WHERE salary > p\_salary

ORDER BY salary;

v\_employee\_id EMPLOYEES.EMPLOYEE\_ID%TYPE;

v\_first\_name EMPLOYEES.FIRST\_NAME%TYPE;

v\_salary EMPLOYEES.SALARY%TYPE;

p\_salary EMPLOYEES.SALARY%TYPE := 6000;

BEGIN

OPEN cur\_employees(p\_salary);

LOOP

FETCH cur\_employees INTO v\_employee\_id, v\_first\_name, v\_salary;

EXIT WHEN NOT FOUND;

RAISE NOTICE '% % %' , v\_employee\_id, v\_first\_name, v\_salary;

END LOOP;

CLOSE cur\_employees;

--Utilitzar el mateix cursor pero amb FOR

FOR v\_curEmployees\_fila IN cur\_employees(p\_salary) LOOP

RAISE NOTICE '% % %' , v\_CurEmployees\_fila.employee\_id, v\_CurEmployees\_fila.first\_name, v\_CurEmployees\_fila.salary;

END LOOP;

END $$;

**Exercici 5.** Fes un bloc anònim en el que s’actualitzin els salaris dels empleats que cobren més de 6000, segons els següents criteris:

- Si treballa en el departament 5 s’incrementa en un 10%.

- Si treballa en els departaments 6,7 i 8 s’incrementa en un 15%.

- En cas contrari no s’ha d’actualitzar.

Has d’anar informant dels canvis fets, mostra el codi d’empleat, nom, cognom i nou salari.

/\*\*\*EXERCICI 5. ACTUALITZACIONS EN RECÓRRER REGISTRES\*\*\*/

DO $$

DECLARE

cur\_employees CURSOR FOR

SELECT employee\_id, first\_name, last\_name, salary, department\_id

FROM employees

WHERE salary > 6000;

v\_nousalari numeric;

v\_actualitza boolean:=false;

BEGIN

FOR v\_curEmployees\_fila IN cur\_employees LOOP

--Depen del departament increntem el salari un % determinat

v\_actualitza := false;

IF v\_curEmployees\_fila.department\_id = 5 THEN

v\_nousalari := v\_curEmployees\_fila.salary \* 1.10;

v\_actualitza := true;

ELSIF v\_curEmployees\_fila.department\_id BETWEEN 6 AND 8 THEN

v\_nousalari := v\_curEmployees\_fila.salary \*1.15;

v\_actualitza := true;

END IF;

IF v\_actualitza THEN

UPDATE employees

SET salary = v\_nousalari

WHERE CURRENT of cur\_employees;

RAISE NOTICE '% % % %' , v\_CurEmployees\_fila.employee\_id, v\_CurEmployees\_fila.first\_name, v\_CurEmployees\_fila.last\_name, round(v\_nousalari);

END IF;

END LOOP;

END $$;

**Exercici 6.** Fes un bloc anònim que mostri els 3 empleats que guanyen menys. (Només 3)

-- EXERCICI 6 – AQUEST EXERCICI TAMBÉ ES POT RESOLDRE UTILITZANT LA CLÀUSULA LIMIT LLAVORS NO FA FALTA UTILITZAR CAP CONTADOR (PROVA-HO)

do $$

DECLARE

curEmpleats CURSOR FOR

select employee\_id, first\_name, salary

from employees

order by salary asc;

regEmpleats record;

rowcount int :=0;

BEGIN

OPEN curEmpleats;

LOOP

fetch curEmpleats into regEmpleats;

rowcount := rowcount + 1;

exit when not found OR rowcount = 4;

raise notice '% % %', regEmpleats.employee\_id , regEmpleats.first\_name, regEmpleats.salary;

END LOOP;

CLOSE curEmpleats;

END $$;