**Tema 5**

/\*

1. Pentru fiecare job (titlu - care va fi afisat o singura data) obtineti

lista angajatilor (nume si salariu) care lucreaza in prezent pe jobul

respectiv. Tratati cazul in care nu exista angajati care sa lucreze

in prezent pe un anumit job. Rezolvati problema folosind:

\*/

-- a. cursoare clasice

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary

FROM employees

WHERE job\_id = jobId;

v\_jobs\_job\_id jobs.job\_id%TYPE;

v\_jobs\_job\_title jobs.job\_title%TYPE;

v\_emp\_last\_name employees.last\_name%TYPE;

v\_emp\_first\_name employees.first\_name%TYPE;

v\_emp\_salary employees.salary%TYPE;

v\_nr\_emp NUMBER;

BEGIN

OPEN c\_jobs;

LOOP

FETCH c\_jobs INTO v\_jobs\_job\_id, v\_jobs\_job\_title;

EXIT WHEN c\_jobs%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Job ' || v\_jobs\_job\_title || ': ');

v\_nr\_emp := 0;

OPEN c\_employees(v\_jobs\_job\_id);

LOOP

FETCH c\_employees INTO v\_emp\_last\_name, v\_emp\_first\_name, v\_emp\_salary;

EXIT WHEN c\_employees%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || v\_emp\_first\_name || ' ' || v\_emp\_last\_name || ' are salariul ' || v\_emp\_salary);

v\_nr\_emp := v\_nr\_emp + 1;

END LOOP;

CLOSE c\_employees;

IF v\_nr\_emp = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

END IF;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

CLOSE c\_jobs;

END;

/

-- b. ciclu cursoare

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary

FROM employees

WHERE job\_id = jobId;

v\_nr\_emp NUMBER;

BEGIN

FOR i in c\_jobs LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

v\_nr\_emp := 0;

FOR j in c\_employees(i.job\_id) LOOP

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

v\_nr\_emp := v\_nr\_emp + 1;

END LOOP;

IF v\_nr\_emp = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

END IF;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

END;

/

-- c. ciclu cursoare cu subcereri

DECLARE

v\_nr\_emp NUMBER;

BEGIN

FOR i in (SELECT job\_id, job\_title

FROM jobs) LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

v\_nr\_emp := 0;

FOR j in (SELECT last\_name, first\_name, salary

FROM employees

WHERE job\_id = i.job\_id) LOOP

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

v\_nr\_emp := v\_nr\_emp + 1;

END LOOP;

IF v\_nr\_emp = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

END IF;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

END;

/

-- d. expresii cursor

DECLARE

TYPE refcursor IS REF CURSOR;

CURSOR c IS

SELECT job\_title, CURSOR (SELECT first\_name, last\_name, salary

FROM employees e

WHERE e.job\_id = j.job\_id)

FROM jobs j;

TYPE emp IS RECORD (last\_name employees.last\_name%TYPE,

first\_name employees.first\_name%TYPE,

salary employees.salary%TYPE);

TYPE tbl IS TABLE OF emp

INDEX BY binary\_integer;

v\_cursor refcursor;

emp\_tbl tbl;

titlu jobs.job\_title%TYPE;

BEGIN

OPEN c;

LOOP

FETCH c INTO titlu, v\_cursor;

EXIT WHEN c%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Job ' || titlu || ': ');

LOOP

FETCH v\_cursor BULK COLLECT INTO emp\_tbl;

IF emp\_tbl.count <> 0 THEN

FOR i in emp\_tbl.FIRST..emp\_tbl.LAST LOOP

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || emp\_tbl(i).first\_name || ' ' || emp\_tbl(i).last\_name || ' are salariul ' || emp\_tbl(i).salary);

END LOOP;

ELSE

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

END IF;

EXIT WHEN v\_cursor%NOTFOUND;

END LOOP;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

CLOSE c;

END;

/

-- 2. Modificati exercitiul anterior astfel incat sa obtineti si urmatoarele informatii:

-- - un numar de ordine pentru fiecare angajat care va fi resetat pentru fiecare job

-- - pentru fiecare job:

-- o numarul de angajati

-- o valoarea lunara a veniturilor angajatilor

-- o valoarea medie a veniturilor angajatilor

-- - indiferent job

-- o numarul total de angajati

-- o valoarea totala lunara a veniturilor angajatilor

-- o valoarea medie a veniturilor angajatilor

-- cursoare clasice

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary, commission\_pct

FROM employees

WHERE job\_id = jobId;

v\_jobs\_job\_id jobs.job\_id%TYPE;

v\_jobs\_job\_title jobs.job\_title%TYPE;

v\_emp\_last\_name employees.last\_name%TYPE;

v\_emp\_first\_name employees.first\_name%TYPE;

v\_emp\_salary employees.salary%TYPE;

v\_emp\_commission\_pct employees.commission\_pct%TYPE;

v\_nr\_emp\_job NUMBER;

v\_sal\_job NUMBER;

v\_nr\_emp\_total NUMBER := 0;

v\_sal\_total NUMBER := 0;

BEGIN

OPEN c\_jobs;

LOOP

FETCH c\_jobs INTO v\_jobs\_job\_id, v\_jobs\_job\_title;

EXIT WHEN c\_jobs%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Job ' || v\_jobs\_job\_title || ': ');

v\_nr\_emp\_job := 0;

v\_sal\_job := 0;

OPEN c\_employees(v\_jobs\_job\_id);

LOOP

FETCH c\_employees INTO v\_emp\_last\_name, v\_emp\_first\_name, v\_emp\_salary, v\_emp\_commission\_pct;

EXIT WHEN c\_employees%NOTFOUND;

v\_nr\_emp\_job := v\_nr\_emp\_job + 1;

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || v\_emp\_first\_name || ' ' || v\_emp\_last\_name || ' are salariul ' || v\_emp\_salary);

v\_sal\_job := v\_sal\_job + v\_emp\_salary + v\_emp\_salary \* nvl(v\_emp\_commission\_pct, 0);

END LOOP;

CLOSE c\_employees;

IF v\_nr\_emp\_job = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

ELSE

DBMS\_OUTPUT.PUT('Acest job are ' || v\_nr\_emp\_job || ' angajati');

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_job);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_job / v\_nr\_emp\_job);

END IF;

v\_nr\_emp\_total := v\_nr\_emp\_total + v\_nr\_emp\_job;

v\_sal\_total := v\_sal\_total + v\_sal\_job;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

CLOSE c\_jobs;

DBMS\_OUTPUT.PUT('Numarul total de angajati este ' || v\_nr\_emp\_total);

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_total);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_total / v\_nr\_emp\_total);

END;

/

-- b. ciclu cursoare

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary, commission\_pct

FROM employees

WHERE job\_id = jobId;

v\_nr\_emp\_job NUMBER;

v\_sal\_job NUMBER;

v\_nr\_emp\_total NUMBER := 0;

v\_sal\_total NUMBER := 0;

BEGIN

FOR i in c\_jobs LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

v\_nr\_emp\_job := 0;

v\_sal\_job := 0;

FOR j in c\_employees(i.job\_id) LOOP

v\_nr\_emp\_job := v\_nr\_emp\_job + 1;

DBMS\_OUTPUT.PUT\_LINE(v\_nr\_emp\_job || '. Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

v\_sal\_job := v\_sal\_job + j.salary + j.salary \* nvl(j.commission\_pct, 0);

END LOOP;

IF v\_nr\_emp\_job = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

ELSE

DBMS\_OUTPUT.PUT('Acest job are ' || v\_nr\_emp\_job || ' angajati');

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_job);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_job / v\_nr\_emp\_job);

END IF;

v\_nr\_emp\_total := v\_nr\_emp\_total + v\_nr\_emp\_job;

v\_sal\_total := v\_sal\_total + v\_sal\_job;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

DBMS\_OUTPUT.NEW\_LINE;

DBMS\_OUTPUT.PUT('Numarul total de angajati este ' || v\_nr\_emp\_total);

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_total);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_total / v\_nr\_emp\_total);

END;

/

-- c. ciclu cursoare cu subcereri

DECLARE

v\_nr\_emp\_job NUMBER;

v\_sal\_job NUMBER;

v\_nr\_emp\_total NUMBER := 0;

v\_sal\_total NUMBER := 0;

BEGIN

FOR i in (SELECT job\_id, job\_title

FROM jobs) LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

v\_nr\_emp\_job := 0;

v\_sal\_job := 0;

FOR j in (SELECT last\_name, first\_name, salary, commission\_pct

FROM employees

WHERE job\_id = i.job\_id) LOOP

v\_nr\_emp\_job := v\_nr\_emp\_job + 1;

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

v\_sal\_job := v\_sal\_job + j.salary + j.salary \* nvl(j.commission\_pct, 0);

END LOOP;

IF v\_nr\_emp\_job = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

ELSE

DBMS\_OUTPUT.PUT('Acest job are ' || v\_nr\_emp\_job || ' angajati');

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_job);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_job / v\_nr\_emp\_job);

END IF;

v\_nr\_emp\_total := v\_nr\_emp\_total + v\_nr\_emp\_job;

v\_sal\_total := v\_sal\_total + v\_sal\_job;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

DBMS\_OUTPUT.NEW\_LINE;

DBMS\_OUTPUT.PUT('Numarul total de angajati este ' || v\_nr\_emp\_total);

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_total);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_total / v\_nr\_emp\_total);

END;

/

-- d. expresii cursor

DECLARE

TYPE refcursor IS REF CURSOR;

CURSOR c IS

SELECT job\_title, CURSOR (SELECT first\_name, last\_name, salary, commission\_pct

FROM employees e

WHERE e.job\_id = j.job\_id)

FROM jobs j;

TYPE emp IS RECORD (last\_name employees.last\_name%TYPE,

first\_name employees.first\_name%TYPE,

salary employees.salary%TYPE,

commission\_pct employees.commission\_pct%TYPE);

TYPE tbl IS TABLE OF emp

INDEX BY binary\_integer;

v\_cursor refcursor;

emp\_tbl tbl;

titlu jobs.job\_title%TYPE;

v\_sal\_job NUMBER := 0;

v\_nr\_emp\_total NUMBER := 0;

v\_sal\_total NUMBER := 0;

BEGIN

OPEN c;

LOOP

FETCH c INTO titlu, v\_cursor;

EXIT WHEN c%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Job ' || titlu || ': ');

LOOP

FETCH v\_cursor BULK COLLECT INTO emp\_tbl;

v\_sal\_job := 0;

IF emp\_tbl.count <> 0 THEN

FOR i in emp\_tbl.FIRST..emp\_tbl.LAST LOOP

DBMS\_OUTPUT.PUT\_LINE('Angajatul cu numele ' || emp\_tbl(i).first\_name || ' ' || emp\_tbl(i).last\_name || ' are salariul ' || emp\_tbl(i).salary);

v\_sal\_job := v\_sal\_job + emp\_tbl(i).salary + emp\_tbl(i).salary \* nvl(emp\_tbl(i).commission\_pct, 0);

END LOOP;

DBMS\_OUTPUT.PUT('Acest job are ' || emp\_tbl.count || ' angajati');

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_job);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_job / emp\_tbl.count);

ELSE

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

END IF;

v\_nr\_emp\_total := v\_nr\_emp\_total + emp\_tbl.count;

v\_sal\_total := v\_sal\_total + v\_sal\_job;

DBMS\_OUTPUT.NEW\_LINE;

EXIT WHEN v\_cursor%NOTFOUND;

END LOOP;

END LOOP;

CLOSE c;

DBMS\_OUTPUT.NEW\_LINE;

DBMS\_OUTPUT.PUT('Numarul total de angajati este ' || v\_nr\_emp\_total);

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_total);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_total / v\_nr\_emp\_total);

END;

/

/\*

3. Modificati exercitiul anterior astfel incat sa obtineti suma totala

alocata lunar pentru plata salariilor si a comisioanelor tuturor

angajatilor, iar pentru fiecare angajat, cat la suta din aceasta

suma castiga lunar.

\*/

set serveroutput on;

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary, commission\_pct

FROM employees

WHERE job\_id = jobId;

CURSOR c\_sal\_com IS

SELECT sum(salary) sal, sum(nvl(commission\_pct, 0)) com

FROM employees;

v\_nr\_emp\_job NUMBER;

v\_sal\_job NUMBER;

v\_nr\_emp\_total NUMBER := 0;

v\_sal\_total NUMBER := 0;

v\_com\_total NUMBER := 0;

BEGIN

FOR i in c\_jobs LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

FOR k in c\_sal\_com LOOP

v\_sal\_total := k.sal;

v\_com\_total := k.com;

END LOOP;

v\_sal\_total := v\_sal\_total + v\_sal\_total \* v\_com\_total;

v\_nr\_emp\_job := 0;

v\_sal\_job := 0;

FOR j in c\_employees(i.job\_id) LOOP

v\_nr\_emp\_job := v\_nr\_emp\_job + 1;

DBMS\_OUTPUT.PUT(v\_nr\_emp\_job || '. Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

DBMS\_OUTPUT.PUT\_LINE(' care reprezinta ' || round(j.salary \* 100/v\_sal\_total, 2)|| '% din total');

v\_sal\_job := v\_sal\_job + j.salary + j.salary \* nvl(j.commission\_pct, 0);

END LOOP;

IF v\_nr\_emp\_job = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista angajati cu acest job');

ELSE

DBMS\_OUTPUT.PUT('Acest job are ' || v\_nr\_emp\_job || ' angajati');

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_job);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_job / v\_nr\_emp\_job);

END IF;

v\_nr\_emp\_total := v\_nr\_emp\_total + v\_nr\_emp\_job;

v\_sal\_total := v\_sal\_total + v\_sal\_job;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

DBMS\_OUTPUT.NEW\_LINE;

DBMS\_OUTPUT.PUT('Numarul total de angajati este ' || v\_nr\_emp\_total);

DBMS\_OUTPUT.PUT(', totalul veniturilor lunare ' || v\_sal\_total);

DBMS\_OUTPUT.PUT\_LINE(' si media veniturilor lunare ' || v\_sal\_total / v\_nr\_emp\_total);

END;

/

/\*

4. Modificati exercitiul anterior astfel incat sa obtineti

pentru fiecare job primii 5 angajati care castiga cel mai

mare salariu lunar. Specificati daca pentru un job sunt mai

putin de 5 angajati.

\*/

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary, commission\_pct

FROM employees

WHERE job\_id = jobId;

v\_nr\_emp\_job NUMBER;

BEGIN

FOR i in c\_jobs LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

v\_nr\_emp\_job := 0;

FOR j in c\_employees(i.job\_id) LOOP

v\_nr\_emp\_job := v\_nr\_emp\_job + 1;

DBMS\_OUTPUT.PUT\_LINE(v\_nr\_emp\_job || '. Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

EXIT WHEN v\_nr\_emp\_job = 5;

END LOOP;

IF v\_nr\_emp\_job < 5 THEN

DBMS\_OUTPUT.PUT\_LINE('Sunt mai putin de 5 angajati cu acest job.');

END IF;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

END;

/

/\*

5. Modificati exercitiul anterior astfel incat sa obtineti

pentru fiecare job top 5 angajati. Daca exista mai multi

angajati care respecta criteriul de selectie care au acelasi

salariu, atunci acestia vor ocupa aceeasi pozitie in top 5.

\*/

DECLARE

CURSOR c\_jobs IS

SELECT job\_id, job\_title

FROM jobs;

CURSOR c\_employees (jobId jobs.job\_id%TYPE) IS

SELECT last\_name, first\_name, salary, commission\_pct

FROM employees

WHERE job\_id = jobId;

v\_nr\_emp\_job NUMBER;

v\_sal\_vechi employees.salary%TYPE := 0;

BEGIN

FOR i in c\_jobs LOOP

DBMS\_OUTPUT.PUT\_LINE('Job ' || i.job\_title || ': ');

v\_nr\_emp\_job := 0;

FOR j in c\_employees(i.job\_id) LOOP

IF v\_sal\_vechi != j.salary THEN

v\_nr\_emp\_job := v\_nr\_emp\_job + 1;

v\_sal\_vechi := j.salary;

END IF;

DBMS\_OUTPUT.PUT\_LINE(v\_nr\_emp\_job || '. Angajatul cu numele ' || j.first\_name || ' ' || j.last\_name || ' are salariul ' || j.salary);

EXIT WHEN v\_nr\_emp\_job = 5;

END LOOP;

IF v\_nr\_emp\_job < 5 THEN

DBMS\_OUTPUT.PUT\_LINE('Sunt mai putin de 5 angajati cu acest job.');

END IF;

DBMS\_OUTPUT.NEW\_LINE;

END LOOP;

END;

/