**Tema laborator 6**

--Tema

SET SERVEROUTPUT ON;

CREATE type tip\_orase\_ido;

CREATE TABLE excursie\_ido (

cod\_excursie NUMBER(4) NOT NULL PRIMARY KEY,

denumire VARCHAR2(20),

orase VARCHAR(20),

status varchar(20)

);

--E2

DECLARE

orase\_curente excursie\_ido.orase%TYPE;

cod\_excursie\_specificat excursie\_ido.cod\_excursie%TYPE;

oras\_vizitat1 VARCHAR2(20);

oras\_vizitat2 VARCHAR2(20);

id\_oras\_1 INTEGER(5);

id\_oras\_2 INTEGER(5);

aux VARCHAR2(20);

oras\_specificat VARCHAR2(20);

pozitie\_sters INTEGER(5);

numar\_orase\_vizitate INTEGER(5);

curr\_orase excursie\_ido.orase%TYPE;

cod\_excursie excursie\_ido.cod\_excursie%TYPE;

nr\_excursii NUMBER;

TYPE vector IS varray(20) OF excursie\_ido.cod\_excursie%TYPE;

excursii vector;

BEGIN

DELETE FROM excursie\_ido;

-- a

for i IN 1..5 LOOP

INSERT INTO excursie\_ido (cod\_excursie, denumire, orase, status) VALUES (i, 'nume '|| i, tip\_orase\_ido('o1', 'o2'), 'aiic');

END LOOP;

-- b

cod\_excursie\_specificat := &cod\_excursie;

SELECT orase INTO orase\_curente

FROM excursie\_ido

WHERE cod\_excursie = cod\_excursie\_specificat;

-- b1

orase\_curente.extend();

orase\_curente(orase\_curente.COUNT) := 'nou';

-- b2

orase\_curente.extend();

for i IN REVERSE 2..orase\_curente.COUNT LOOP

orase\_curente(i) := orase\_curente(i-1);

END LOOP;

orase\_curente(2) := 'sec';

-- b3

oras\_vizitat1 := '\_o1';

oras\_vizitat2 := '\_o2';

id\_oras\_1 := 1;

id\_oras\_2 := 1;

FOR i IN 1..orase\_curente.COUNT LOOP

IF orase\_curente(i) = oras\_vizitat1 THEN

id\_oras\_1 := i;

END IF;

IF orase\_curente(i) = oras\_vizitat2 THEN

id\_oras\_2 := i;

END IF;

END LOOP;

aux := orase\_curente(id\_oras\_1);

orase\_curente(id\_oras\_1) := orase\_curente(id\_oras\_2);

orase\_curente(id\_oras\_2) := aux;

-- b4

oras\_specificat := 'spec';

FOR i IN 1..orase\_curente.COUNT LOOP

IF orase\_curente(i) = oras\_specificat THEN

pozitie\_sters := i;

END IF;

END LOOP;

FOR i IN REVERSE pozitie\_sters..orase\_curente.COUNT LOOP

orase\_curente(i) := orase\_curente(i - 1);

END LOOP;

orase\_curente.TRIM;

-- then update

UPDATE excursie\_ido SET orase = orase\_curente WHERE cod\_excursie = cod\_excursie\_specificat;

-- c

DBMS\_OUTPUT.put\_line('Numar orase: '|| orase\_curente.count ||', orase: ');

FOR i IN 1..orase\_curente.COUNT LOOP

DBMS\_OUTPUT.put\_line(orase\_curente(i) || ', ');

END LOOP;

-- d

SELECT cod\_excursie BULK COLLECT INTO excursii FROM excursie\_ido;

FOR i IN 1..excursii.COUNT LOOP

SELECT orase INTO curr\_orase

FROM excursie\_ido

WHERE cod\_excursie = excursii(i);

DBMS\_OUTPUT.put\_line('Excursia ' || excursii(i) || ': ');

FOR i IN 1..curr\_orase.COUNT LOOP

DBMS\_OUTPUT.put\_line(curr\_orase(i) || ', ');

END LOOP;

END LOOP;

-- e

END;

select \* from excursie\_ido;

--E1

-- a

DECLARE

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

SELECT e.last\_name, e.first\_name, e.salary

FROM employees e, jobs j

WHERE j.job\_id = e.job\_id and j.job\_id = job\_curent;

numar\_joburi NUMBER;

titlu\_job jobs.job\_title%TYPE;

nume\_angajat employees.first\_name%TYPE;

prenume\_angajat employees.last\_name%TYPE;

salariu employees.salary%TYPE;

TYPE tip\_joburi IS TABLE OF jobs.job\_id%TYPE;

joburi tip\_joburi := tip\_joburi();

counter NUMBER(5);

BEGIN

SELECT COUNT(\*) INTO numar\_joburi FROM jobs;

joburi.EXTEND(numar\_joburi);

SELECT j.job\_id BULK COLLECT INTO joburi FROM jobs j;

FOR i IN joburi.FIRST..joburi.LAST LOOP

SELECT job\_title

INTO titlu\_job

FROM jobs j

WHERE j.job\_id = joburi(i);

DBMS\_OUTPUT.put\_line(titlu\_job);

counter := 0;

OPEN c(joburi(i));

LOOP

FETCH c INTO nume\_angajat, prenume\_angajat, salariu;

EXIT WHEN c%NOTFOUND;

DBMS\_OUTPUT.put\_line(nume\_angajat || ' ' || prenume\_angajat || ' ' || salariu);

counter := counter + 1;

END LOOP;

CLOSE c;

IF counter = 0 THEN

DBMS\_OUTPUT.put\_line('Niciun anj');

END IF;

DBMS\_OUTPUT.new\_line();

END LOOP;

END;

/

-- b

DECLARE

TYPE tip\_joburi IS TABLE OF jobs.job\_id%TYPE;

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

SELECT e.last\_name l\_name, e.first\_name f\_name, e.salary salary

FROM employees e, jobs j

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

j.job\_id = job\_curent;

numar\_joburi NUMBER;

titlu\_job jobs.job\_title%TYPE;

nume\_angajat employees.first\_name%TYPE;

prenume\_angajat employees.last\_name%TYPE;

salariu employees.salary%TYPE;

joburi tip\_joburi := tip\_joburi();

counter NUMBER(5);

BEGIN

SELECT COUNT(\*) INTO numar\_joburi FROM jobs;

joburi.EXTEND(numar\_joburi);

SELECT j.job\_id BULK COLLECT INTO joburi FROM jobs j;

FOR i IN joburi.FIRST..joburi.LAST LOOP

SELECT job\_title

INTO titlu\_job

FROM jobs j

WHERE j.job\_id = joburi(i);

DBMS\_OUTPUT.put\_line(titlu\_job);

counter := 0;

FOR j IN c(joburi(i)) LOOP

EXIT WHEN c%NOTFOUND;

DBMS\_OUTPUT.put\_line(j.f\_name || ' ' || j.l\_name || ' ' || j.salary);

counter := counter + 1;

END LOOP;

IF counter = 0 THEN

DBMS\_OUTPUT.put\_line('Niciun ang.');

END IF;

DBMS\_OUTPUT.new\_line();

END LOOP;

END;

/

-- c

DECLARE

TYPE tip\_joburi IS TABLE OF jobs.job\_id%TYPE;

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

SELECT e.last\_name l\_name, e.first\_name f\_name, e.salary salary

FROM employees e, jobs j

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

j.job\_id = job\_curent;

numar\_joburi NUMBER;

titlu\_job jobs.job\_title%TYPE;

nume\_angajat employees.first\_name%TYPE;

prenume\_angajat employees.last\_name%TYPE;

salariu employees.salary%TYPE;

joburi tip\_joburi := tip\_joburi();

counter NUMBER(5);

BEGIN

SELECT COUNT(\*) INTO numar\_joburi FROM jobs;

joburi.EXTEND(numar\_joburi);

select j.job\_id BULK COLLECT INTO joburi FROM jobs j;

FOR i IN joburi.FIRST..joburi.LAST LOOP

SELECT job\_title

INTO titlu\_job

FROM jobs j

WHERE j.job\_id = joburi(i);

DBMS\_OUTPUT.put\_line(titlu\_job);

counter := 0;

FOR j IN (SELECT e.last\_name l\_name, e.first\_name f\_name, e.salary salary

FROM employees e, jobs j

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

j.job\_id = joburi(i))

LOOP

DBMS\_OUTPUT.put\_line(j.f\_name || ' ' || j.l\_name || ' ' || j.salary);

counter := counter + 1;

END LOOP;

IF counter = 0 THEN

DBMS\_OUTPUT.put\_line('Niciun ang.');

END IF;

DBMS\_OUTPUT.new\_line();

END LOOP;

END;

/

--E4

--9

CREATE OR REPLACE TYPE administratori\_ido AS VARRAY(10) OF NUMBER(4);

/

CREATE TABLE utilizatori\_ido (cod\_mgr NUMBER(10),

nume VARCHAR2(20),

lista utilizatori\_ido);

DECLARE

v\_sub administartori\_ido:= administratori\_ido(100,200,300);

v\_lista utilizatori\_ido.lista%TYPE;

BEGIN

INSERT INTO administratori\_ido

VALUES (1, 'Adm 1', v\_sub);

INSERT INTO administratori\_ido

VALUES (2, 'Adm 2', null);

INSERT INTO administratori\_ido

VALUES (3, 'Adm 3', utilizatori\_ido(4,5));

SELECT lista

INTO v\_lista

FROM administratori\_ido

WHERE cod\_mgr=1;

FOR j IN v\_lista.FIRST..v\_lista.LAST LOOP

DBMS\_OUTPUT.PUT\_LINE (v\_lista(j));

END LOOP;

END;

/

SELECT \* FROM manageri\_ido;

--10

CREATE TABLE emp\_test\_ido AS

SELECT id\_utilizator, nume FROM utilizatori

WHERE ROWNUM <= 2;

CREATE OR REPLACE TYPE tip\_telefon\_ido IS TABLE OF VARCHAR(12);

/

ALTER TABLE emp\_test\_ido

ADD (telefon tip\_telefon\_ido)

NESTED TABLE telefon STORE AS tabel\_telefon\_ido;

INSERT INTO emp\_test\_ido

VALUES (500, 'XYZ',tip\_telefon\_ido('074XXX', '0213XXX', '037XXX'));

UPDATE emp\_test\_ido

SET telefon = tip\_telefon\_ido('073XXX', '0214XXX')

WHERE employee\_id=100;

SELECT a.employee\_id, b.\*

FROM emp\_test\_ido a, TABLE (a.telefon) b;

DROP TABLE emp\_test\_ido;

DROP TYPE tip\_telefon\_ido;











