

Tema 3 SGBD - Oprea Tudor (241)

```
/
--drop table excursie_OT;
--drop type tip_orase_ot;
CREATE OR REPLACE TYPE tip_orase_OT IS VARRAY(30) of VARCHAR2(30);

/

create table excursie_ot ( cod_excursie NUMBER(2) CONSTRAINT verifica_id_exc
PRIMARY KEY,
                        denumire VARCHAR2(20),
                        orase tip_orase_OT,
                        status VARCHAR2(15) CONSTRAINT verifica_status CHECK(
                        upper(STATUS) in ('DISPONIBIL','ANULAT')) );

--alter table excursie_ot ADD (orase tip_orase_ot)
--NESTED TABLE orase STORE as tabel_orase_ot;
/
select * from excursie_ot;

/

INSERT into excursie_OT VALUES(1,'Bucuresti', tip_orase_OT('Iasi'),'DISPONIBIL');
INSERT into excursie_OT VALUES(2,'Cluj', tip_orase_OT('Iasi,Braila'),'DISPONIBIL');
INSERT into excursie_OT VALUES(3,'Timisoara', tip_orase_OT('Bucuresti,
Ploiesti'),'ANULAT');
INSERT into excursie_OT VALUES(4,'Brasov',
tip_orase_OT('Alexandria,Constanta'),'DISPONIBIL');
INSERT into excursie_OT VALUES(5,'Constanta', tip_orase_OT('Buzau, Pitesti'),'ANULAT');
INSERT into excursie_OT VALUES(18,'Constanta', tip_orase_OT('Buzau,
Pitesti'),'ANULAT');

/

DECLARE
    i          integer;

    v_cod_exc  excursie_OT.cod_excursie%TYPE:=&p_cod;
    v_orase    excursie_OT.orase %TYPE;

    v_o1       varchar2(30):='&p_nume_oras1';
    v_o2       varchar2(30):='&p_nume_oras2';

    v_id1      integer(6);
```

```

v_id2    integer(6);
v_aux    varchar2(30);

v_oe     varchar2(30):='&p_nume_oraselim';
v_poz    integer(6);

--pentru punctul d: (pt a folosi si tablou indexat , punem orasele
--                din fiecare excursie in tablou cu BULK COLLECT)
TYPE tabindex_OT is TABLE of excursie_ot.cod_excursie%TYPE
INDEX BY BINARY_INTEGER;
sir_excursie  tabindex_OT;

orase_c      excursie_ot.orase%type;
BEGIN
  -- b
  select orase into v_orase from excursie_ot
  where cod_excursie = v_cod_exc;

  --adaugam noul oras in lista de orase vizitate
  v_orase.EXTEND(); v_orase(v_orase.COUNT):='Noul oras';

  -- ca sa inseram pe poz 2, mutam totul mai spre dreapta
  v_orase.EXTEND();
  for i in reverse 2.. v_orase.COUNT loop
    v_orase(i):=v_orase(i-1);
  end loop;
  v_orase(2):='Oras inserat pe poz 2';

  -- specificam doua orase: v_o1, v_o2
  -- le cautam id urile in vector
  for i in 1 .. v_orase.COUNT loop
    if v_orase(i) = v_o1 then v_id1:=i; end if;
    if v_orase(i) = v_o2 then v_id2:=i; end if;
  end loop;
  -- le interschimbam
  v_aux:=v_orase(v_id1);
  v_orase(v_id1):= v_orase(v_id2);
  v_orase(v_id2):=v_aux;

  -- specificam un oras de eliminat: v_oe
  for i in 1.. v_orase.COUNT loop
    if v_orase(i) = v_oe then v_poz:=i; end if; end loop;
    --aplicam alg de eliminare
  for i in reverse v_poz..v_orase.COUNT loop
    v_orase(i):= v_orase(i-1);
  end loop;

```

```
v_orase.trim;
```

```
--update
```

```
update excursie_OT set orase=v_orase
```

```
where cod_excursie = v_cod_exc;
```

```
--c
```

```
--am folosit acelasi cod dat si la pct b
```

```
DBMS_OUTPUT.PUT_LINE('Nr orase: '|| v_orase.COUNT);
```

```
for i in 1..v_orase.COUNT loop
```

```
    DBMS_OUTPUT.PUT_LINE(v_orase(i)|| ' ');
```

```
end loop;
```

```
--d
```

```
select cod_excursie BULK COLLECT into sir_excursie
```

```
from excursie_ot; -- punem orasele in tabel indexat
```

– am incercat sa iterez prin sirul indexat cu while (sper ca este bine)

```
i:=sir_excursie.first;
```

```
while i<=sir_excursie.last LOOP
```

```
    select orase into orase_c from excursie_OT
```

```
    where cod_excursie=sir_excursie(i);
```

```
    -- punem in variabila orase_c orasele viz in excursia cu codul (1..n)
```

```
    DBMS_OUTPUT.PUT_LINE('Excursia cu id-ul ' || sir_excursie(i) ||':');
```

```
    -- orasele din excursie:
```

```
    for j in 1..orase_c.count loop
```

```
        DBMS_OUTPUT.PUT_LINE(orase_c(j)|| ' ');
```

```
    end loop;
```

```
    i:=sir_excursie.NEXT(i);
```

```
end loop;
```

```
END;
```

```
/
```

```
select * from excursie_OT;
```

- - nu am reusit sa testez si punctele b,c,d dar cred ca sunt corecte
- - nu am mai facut punctul e

Worksheet Query Builder

```
--NESTED TABLE orase STORE as tabel_orase_ot;
/
select * from excursie_ot;
/

INSERT into excursie_OT VALUES(1,'Bucuresti', tip_orase_OT('Iasi'),'DISPONIBIL');
INSERT into excursie_OT VALUES(2,'Cluj', tip_orase_OT('Iasi,Braila'),'DISPONIBIL');
INSERT into excursie_OT VALUES(3,'Timisoara', tip_orase_OT('Bucuresti, Ploiesti'),'ANULAT');
INSERT into excursie_OT VALUES(4,'Brasov', tip_orase_OT('Alexandria,Constanta'),'DISPONIBIL');
INSERT into excursie_OT VALUES(5,'Constanta', tip_orase_OT('Buzau, Pitesti'),'ANULAT');
INSERT into excursie_OT VALUES(18,'Constanta', tip_orase_OT('Buzau, Pitesti'),'ANULAT');
```

Script Output x Query Result x

SQL All Rows Fetched: 6 in 0,048 seconds

COD_EXCURSIE	DENUMIRE	ORASE	STATUS
1	1 Bucuresti	GRUPA241.TIP ORASE OT('Iasi')	DISPONIBIL
2	2 Cluj	GRUPA241.TIP ORASE OT('Iasi,Braila')	DISPONIBIL
3	3 Timisoara	GRUPA241.TIP ORASE OT('Bucuresti, Ploiesti')	ANULAT
4	4 Brasov	GRUPA241.TIP ORASE OT('Alexandria,Constanta')	DISPONIBIL
5	5 Constanta	GRUPA241.TIP ORASE OT('Buzau, Pitesti')	ANULAT
6	18 Constanta	GRUPA241.TIP ORASE OT('Buzau, Pitesti')	ANULAT

```
i:=sir_excursie.first;
while i<=sir_excursie.last LOOP
    select orase into orase_c from excursie_OT
    where cod_excursie=sir_excursie(i);
    -- punem in variabila orase_c orasele viz in excursia cu codul (1..n)
    DBMS_OUTPUT.PUT_LINE('Excursia cu id-ul ' || sir_excursie(i) || ':');
    -- orasele din excursie:
    for j in 1..orase_c.count loop
        DBMS_OUTPUT.PUT_LINE(orase_c(i) || ' ');
    end loop;

    i:=sir_excursie.NEXT(i);
end loop;
END;
/
select * from excursie_OT;
```

pt Output x Query Result x

SQL All Rows Fetched: 6 in 0,006 seconds

COD_EXCURSIE	DENUMIRE	ORASE	STATUS
1	1 Bucuresti	GRUPA241.TIP ORASE OT('Iasi')	DISPONIBIL
2	2 Cluj	GRUPA241.TIP ORASE OT('Iasi', 'Braila')	DISPONIBIL