Tema laborator SGBD 4 extra

E2. Definiți un tip colecție denumit tip\_orase\_\*\*\*. Creați tabelul excursie\_\*\*\* cu următoarea structură: cod\_excursie NUMBER(4), denumire VARCHAR2(20), orase tip\_orase\_\*\*\* (ce va conține lista orașelor care se vizitează într-o excursie, într-o ordine stabilită; de exemplu, primul oraș din listă va fi primul oraș vizitat), status (disponibilă sau anulată).

Cod text:

create or replace type tip\_orase\_aho is table of varchar2(20);

/

create table excursie\_aho (

cod\_excursie number(4),

denumire varchar2(20),

orase tip\_orase\_aho,

status varchar2(20))

nested table orase store as tabel\_orase\_aho

/

a. Inserați 5 înregistrări în tabel.

Cod text:

insert into excursie\_aho

values (1, 'Norvegia', tip\_orase\_aho('Oslo', 'Bergen', 'Trondheim', 'Tromso'), 'disponibila');

insert into excursie\_aho

values (2, 'Munte', tip\_orase\_aho('Sinaia', 'Brasov', 'Predeal'), 'disponibila');

insert into excursie\_aho

values (3, 'Grecia', tip\_orase\_aho('Salonic', 'Atena'), 'anulata');

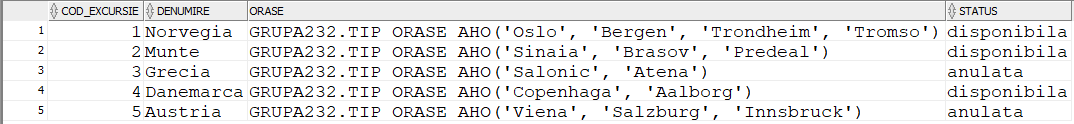
insert into excursie\_aho

values (4, 'Danemarca', tip\_orase\_aho('Copenhaga', 'Aalborg'), 'disponibila');

insert into excursie\_aho

values (5, 'Austria', tip\_orase\_aho('Viena', 'Salzburg', 'Innsbruck'), 'anulata');

Screenshot:



b. Actualizați coloana orase pentru o excursie specificată:

- adăugați un oraș nou în listă, ce va fi ultimul vizitat în excursia respectivă;

Cod text:

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

v\_orase.extend;

v\_orase(v\_orase.last) := 'Linz';

update excursie\_aho

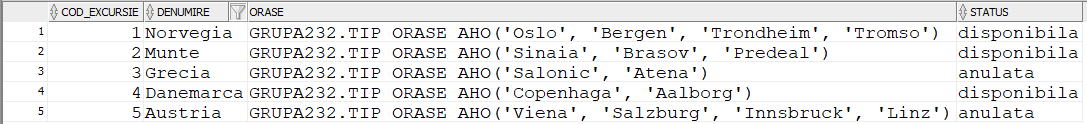
set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

Screenshot:



- adăugați un oraș nou în listă, ce va fi al doilea oraș vizitat în excursia respectivă;

Cod text:

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

v\_orase.extend;

for i in reverse 2..v\_orase.last-1 loop

v\_orase(i+1) := v\_orase(i);

end loop;

v\_orase(2) := 'Graz';

update excursie\_aho

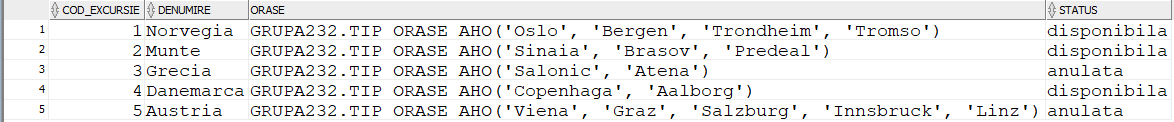
set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

Screenshot:



- inversați ordinea de vizitare a două dintre orașe al căror nume este specificat;

Cod text:

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

oras1 varchar2(20) := 'Salzburg';

oras2 varchar2(20) := 'Innsbruck';

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

for i in v\_orase.first..v\_orase.last loop

if v\_orase(i) = oras1 then

v\_orase(i) := oras2;

elsif v\_orase(i) = oras2 then

v\_orase(i) := oras1;

end if;

end loop;

update excursie\_aho

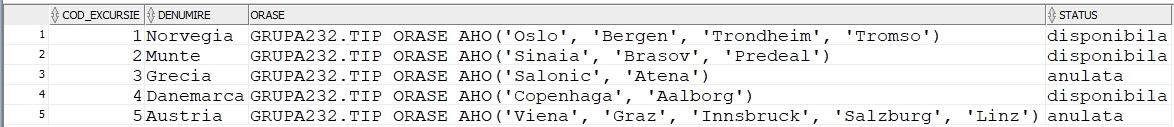
set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

Screenshot:



- eliminați din listă un oraș al cărui nume este specificat.

Cod text:

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

oras varchar2(20) := 'Graz';

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

for i in v\_orase.first..v\_orase.last loop

if v\_orase(i) = oras then

v\_orase.delete(i);

end if;

end loop;

update excursie\_aho

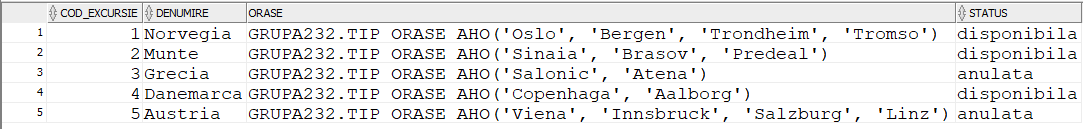
set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

Screenshot:



c. Pentru o excursie al cărui cod este dat, afișați numărul de orașe vizitate, respectiv numele orașelor.

Cod text:

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

dbms\_output.put\_line('Excursia viziteaza ' || v\_orase.count || ' orase: ');

for i in v\_orase.first..v\_orase.last loop

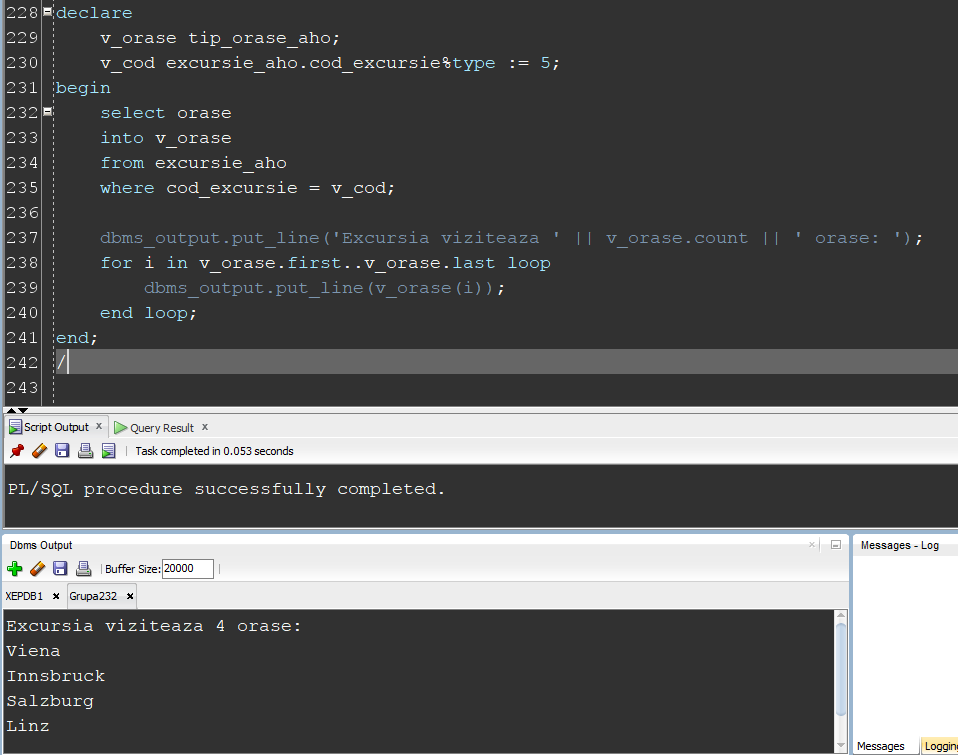
dbms\_output.put\_line(v\_orase(i));

end loop;

end;

/

Screenshot:



d. Pentru fiecare excursie afișați lista orașelor vizitate.

Cod text:

declare

v\_orase tip\_orase\_aho;

v\_nume excursie\_aho.denumire%type;

begin

for e in (select orase, denumire from excursie\_aho) loop

v\_orase := e.orase;

v\_nume := e.denumire;

dbms\_output.put\_line('Orase excursia ' || v\_nume || ': ');

for i in v\_orase.first..v\_orase.last loop

dbms\_output.put\_line(v\_orase(i));

end loop;

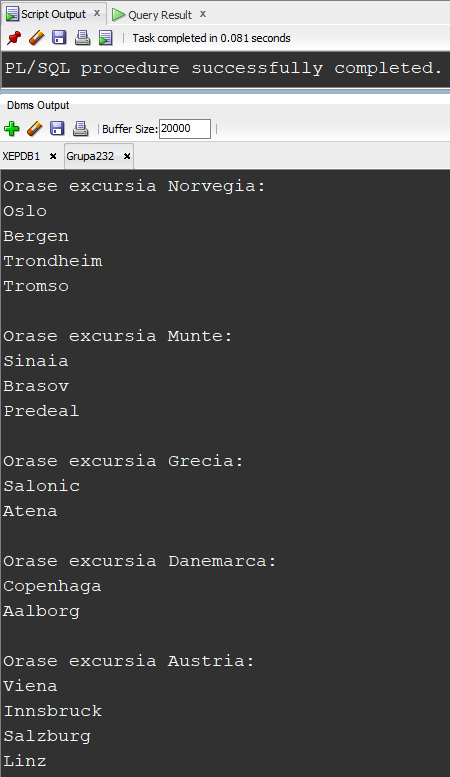
dbms\_output.new\_line;

end loop;

end;

/

Screenshot:



e. Anulați excursiile cu cele mai puține orașe vizitate.

Cod text:

declare

v\_nr\_min number(4) := 1000;

begin

for e in (select orase from excursie\_aho) loop

if e.orase.count < v\_nr\_min then

v\_nr\_min := e.orase.count;

end if;

end loop;

for e in (select orase, cod\_excursie from excursie\_aho) loop

if e.orase.count = v\_nr\_min then

update excursie\_aho

set status = 'anulata'

where cod\_excursie = e.cod\_excursie;

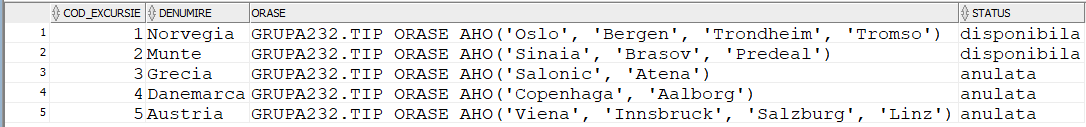
end if;

end loop;

end;

/

Screenshot:



E3. Rezolvați problema anterioară folosind un alt tip de colecție studiat. - varray

Definiți un tip colecție denumit tip\_orase\_\*\*\*. Creați tabelul excursie\_\*\*\* cu următoarea structură: cod\_excursie NUMBER(4), denumire VARCHAR2(20), orase tip\_orase\_\*\*\* (ce va conține lista orașelor care se vizitează într-o excursie, într-o ordine stabilită; de exemplu, primul oraș din listă va fi primul oraș vizitat), status (disponibilă sau anulată).

Cod text:

create or replace type tip\_orase\_aho is varray(10) of varchar2(20);

/

create table excursie\_aho (

cod\_excursie number(4),

denumire varchar2(20),

orase tip\_orase\_aho,

status varchar2(20));

a. Inserați 5 înregistrări în tabel.

insert into excursie\_aho

values (1, 'Norvegia', tip\_orase\_aho('Oslo', 'Bergen', 'Trondheim', 'Tromso'), 'disponibila');

insert into excursie\_aho

values (2, 'Munte', tip\_orase\_aho('Sinaia', 'Brasov', 'Predeal'), 'disponibila');

insert into excursie\_aho

values (3, 'Grecia', tip\_orase\_aho('Salonic', 'Atena'), 'anulata');

insert into excursie\_aho

values (4, 'Danemarca', tip\_orase\_aho('Copenhaga', 'Aalborg'), 'disponibila');

insert into excursie\_aho

values (5, 'Austria', tip\_orase\_aho('Viena', 'Salzburg', 'Innsbruck'), 'anulata');

b. Actualizați coloana orase pentru o excursie specificată:

- adăugați un oraș nou în listă, ce va fi ultimul vizitat în excursia respectivă;

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

v\_orase.extend;

v\_orase(v\_orase.last) := 'Linz';

update excursie\_aho

set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

- adăugați un oraș nou în listă, ce va fi al doilea oraș vizitat în excursia respectivă;

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

v\_orase.extend;

for i in reverse 2..v\_orase.last-1 loop

v\_orase(i+1) := v\_orase(i);

end loop;

v\_orase(2) := 'Graz';

update excursie\_aho

set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

- inversați ordinea de vizitare a două dintre orașe al căror nume este specificat;

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

oras1 varchar2(20) := 'Salzburg';

oras2 varchar2(20) := 'Innsbruck';

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

for i in v\_orase.first..v\_orase.last loop

if v\_orase(i) = oras1 then

v\_orase(i) := oras2;

elsif v\_orase(i) = oras2 then

v\_orase(i) := oras1;

end if;

end loop;

update excursie\_aho

set orase = v\_orase

where cod\_excursie = v\_cod;

end;

/

- eliminați din listă un oraș al cărui nume este specificat.

declare

v\_orase tip\_orase\_aho;

v\_orase2 tip\_orase\_aho := tip\_orase\_aho();

v\_cod excursie\_aho.cod\_excursie%type := 5;

oras varchar2(20) := 'Graz';

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

for i in v\_orase.first..v\_orase.last loop

if v\_orase(i) <> oras then

v\_orase2.extend;

v\_orase2(v\_orase2.last) := v\_orase(i);

end if;

end loop;

update excursie\_aho

set orase = v\_orase2

where cod\_excursie = v\_cod;

end;

/

c. Pentru o excursie al cărui cod este dat, afișați numărul de orașe vizitate, respectiv numele orașelor.

declare

v\_orase tip\_orase\_aho;

v\_cod excursie\_aho.cod\_excursie%type := 5;

begin

select orase

into v\_orase

from excursie\_aho

where cod\_excursie = v\_cod;

dbms\_output.put\_line('Excursia viziteaza ' || v\_orase.count || ' orase: ');

for i in v\_orase.first..v\_orase.last loop

dbms\_output.put\_line(v\_orase(i));

end loop;

end;

/

d. Pentru fiecare excursie afișați lista orașelor vizitate.

declare

v\_orase tip\_orase\_aho;

v\_nume excursie\_aho.denumire%type;

begin

for e in (select orase, denumire from excursie\_aho) loop

v\_orase := e.orase;

v\_nume := e.denumire;

dbms\_output.put\_line('Orase excursia ' || v\_nume || ': ');

for i in v\_orase.first..v\_orase.last loop

dbms\_output.put\_line(v\_orase(i));

end loop;

dbms\_output.new\_line;

end loop;

end;

/

e. Anulați excursiile cu cele mai puține orașe vizitate.

declare

v\_nr\_min number(4) := 1000;

begin

for e in (select orase from excursie\_aho) loop

if e.orase.count < v\_nr\_min then

v\_nr\_min := e.orase.count;

end if;

end loop;

for e in (select orase, cod\_excursie from excursie\_aho) loop

if e.orase.count = v\_nr\_min then

update excursie\_aho

set status = 'anulata'

where cod\_excursie = e.cod\_excursie;

end if;

end loop;

end;

/