Huja Petru Alexandru - Proiect SGBD

1.

Prezentați pe scurt baza de date

-- baza de date care permite crearea de utilizatori care la randul lor pot crea formulare

-- parola utilizatorilor este pastrata separat si este encrypted

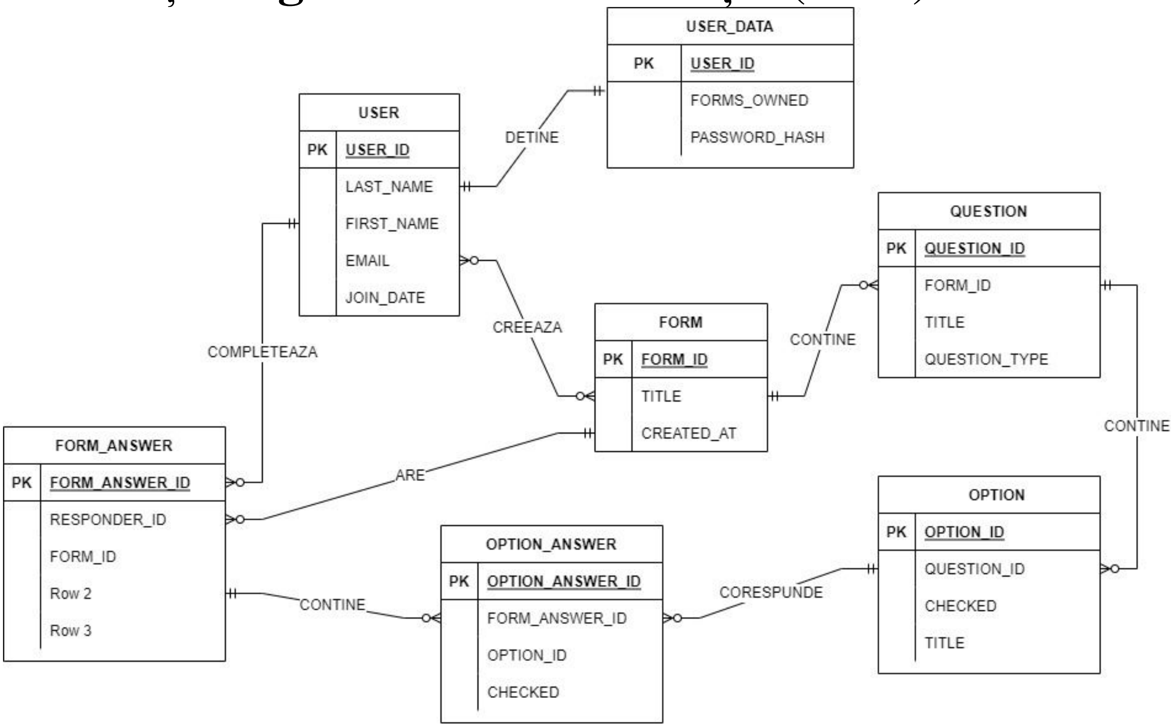
-- formularele pot avea oricate intrebari de tip radio si checkbox

-- intrebarile poat avea oricate variante de raspuns

-- utilizatorii pot raspunde la formulare create de alti utilizatori

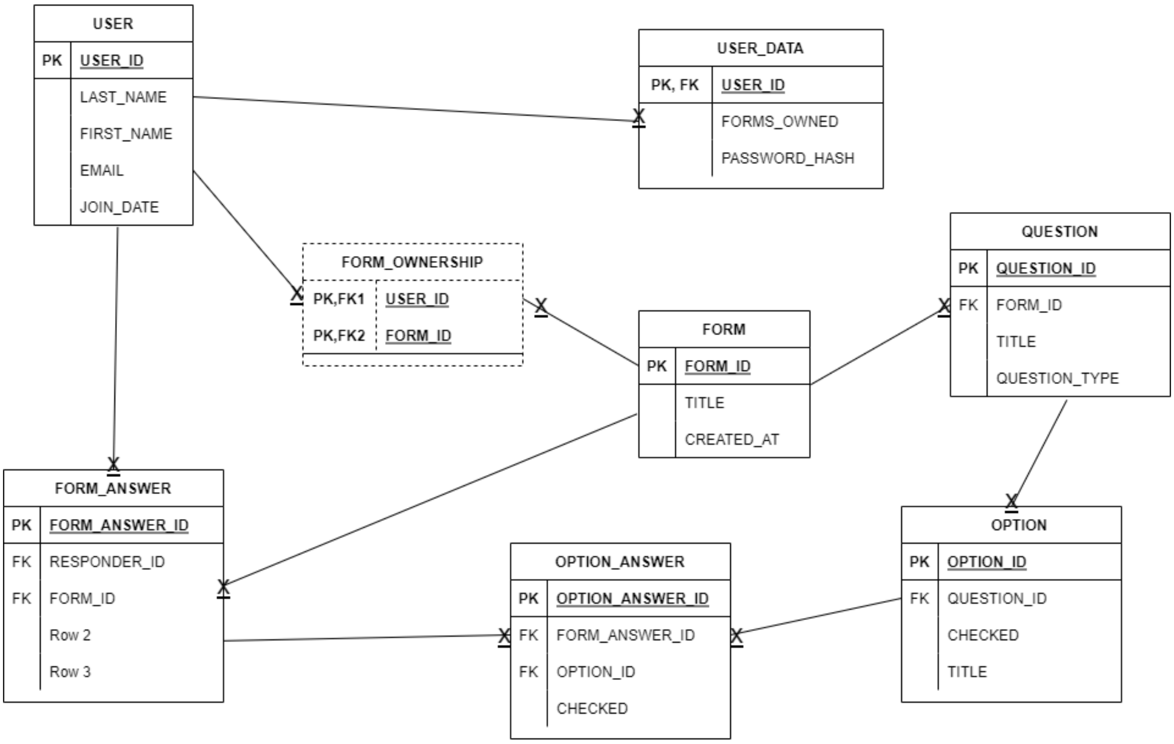
2.

Realizați **diagrama entitate-relație** (ERD).



3.

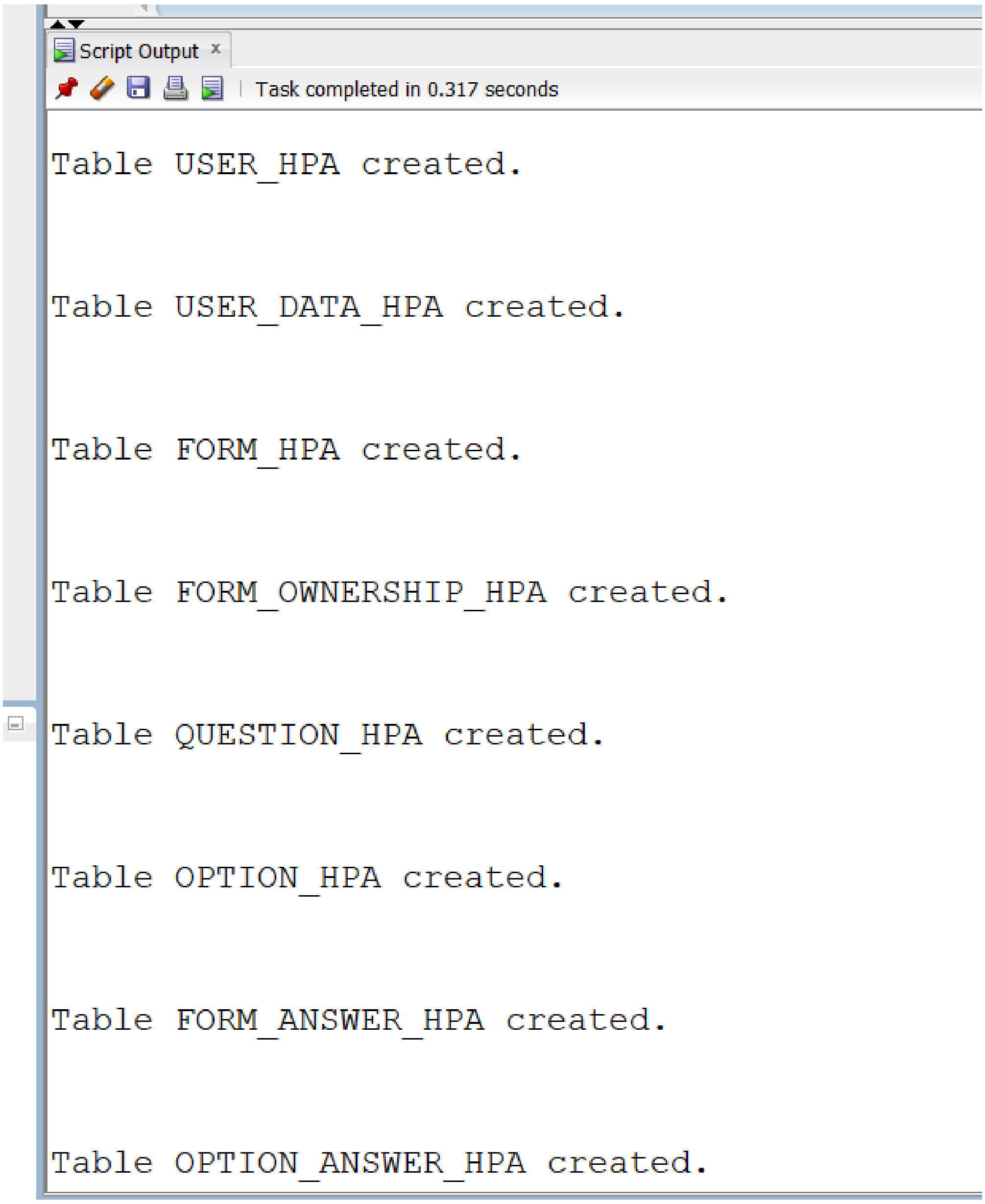
Pornind de la diagrama entitate-relație realizați **diagrama conceptuală** a modelului propus, integrând toate atributele necesare.



4.

Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, implementând toate constrângerile de integritate necesare (chei primare, cheile externe etc).

```
create table user_hpa (

   user_id varchar2(30) primary key,

   last_name varchar2(30) not null,

   first_name varchar2(30) not null,
   email varchar2(30) unique not null,
   join_date date default sysdate
);

create table user_data_hpa (
   user_id varchar2(30) primary key,
   forms_owned number(10) default 0,
   password_hash varchar2(100) not null,
   foreign key (user_id) references user_hpa(user_id) on delete cascade
);

create table form_hpa (
   form_id varchar2(30) primary key,
   title varchar2(100),
   created_at date default sysdate
);

create table form_ownership_hpa (
   form_id varchar2(30),
   user_id varchar2(30),
   primary key (form_id, user_id),
   foreign key (form_id) references form_hpa(form_id) on delete cascade,
   foreign key (user_id) references user_hpa(user_id) on delete cascade
);

create table question_hpa (
   question_id varchar2(30) primary key,
   title varchar2(100),
   form_id varchar2(30),
   q_type varchar2(10),
   foreign key (form_id) references form_hpa(form_id) on delete cascade
);

create table option_hpa (
   option_id varchar2(30) primary key,
   question_id varchar2(30),
   checked number(1),
   title varchar2(100),
   foreign key (question_id) references question_hpa(question_id) on delete cascade
);

create table form_answer_hpa (
   form_answer_id varchar2(30) primary key,
   responder_id varchar2(30),
   form_id varchar2(30),
   answered_at date default sysdate,
   foreign key (responder_id) references user_hpa(user_id) on delete cascade,
   foreign key (form_id) references form_hpa(form_id) on delete cascade
);

create table option_answer_hpa (
   option_answer_id varchar2(30) primary key,
   form_answer_id varchar2(30),
   option_id varchar2(30),
   checked number(1),
   foreign key (form_answer_id) references form_answer_hpa(form_answer_id) on delete cascade,
   foreign key (option_id) references option_hpa(option_id) on delete cascade

);
```



Task completed in 0.317 seconds

```
Table USER_HPA created.


Table USER_DATA_HPA created.


Table FORM_HPA created.


Table FORM_OWNERSHIP_HPA created.


Table QUESTION_HPA created.


Table OPTION_HPA created.


Table FORM_ANSWER_HPA created.


Table OPTION_ANSWER_HPA created.
```

5.

Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

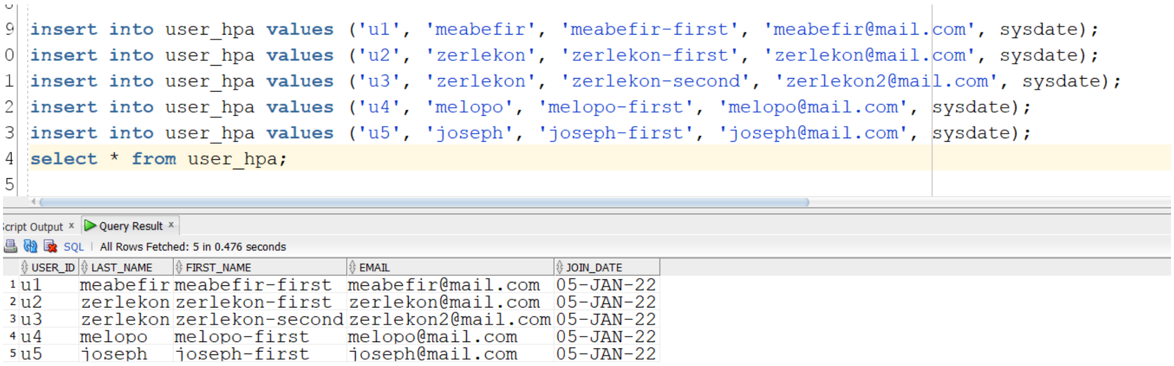
insert into user_hpa values ('u1', 'meabefir', 'meabefir-first', 'meabefir@mail.com', sysdate);

insert into user_hpa values ('u2', 'zerlekon', 'zerlekon-first', 'zerlekon@mail.com', sysdate);

insert into user_hpa values ('u3', 'zerlekon', 'zerlekon-second', 'zerlekon2@mail.com', sysdate);

insert into user_hpa values ('u4', 'melopo', 'melopo-first', 'melopo@mail.com', sysdate);

insert into user_hpa values ('u5', 'joseph', 'joseph-first', 'joseph@mail.com', sysdate);



```
insert into user_hpa values ('u1', 'meabefir', 'meabefir-first', 'meabefir@mail.com', sysdate);
insert into user_hpa values ('u2', 'zerlekon', 'zerlekon-first', 'zerlekon@mail.com', sysdate);
insert into user_hpa values ('u3', 'zerlekon', 'zerlekon-second', 'zerlekon2@mail.com', sysdate);
insert into user_hpa values ('u4', 'melopo', 'melopo-first', 'melopo@mail.com', sysdate);
insert into user_hpa values ('u5', 'joseph', 'joseph-first', 'joseph@mail.com', sysdate);
select * from user_hpa;
```

All Rows Fetched: 5 in 0.476 seconds

```
  USER_ID LAST_NAME FIRST_NAME      EMAIL              JOIN_DATE
1 u1      meabefir  meabefir-first  meabefir@mail.com  05-JAN-22
2 u2      zerlekon  zerlekon-first  zerlekon@mail.com  05-JAN-22
3 u3      zerlekon  zerlekon-second zerlekon2@mail.com 05-JAN-22
4 u4      melopo    melopo-first    melopo@mail.com    05-JAN-22
5 u5      joseph    joseph-first    joseph@mail.com    05-JAN-22
```

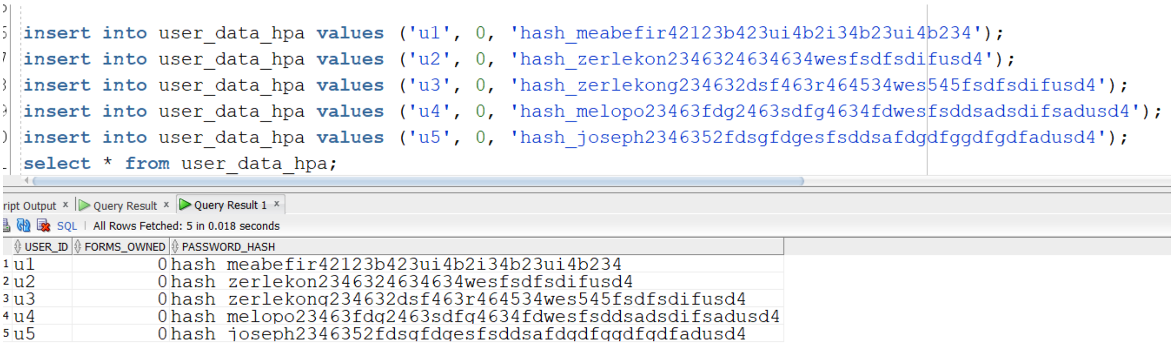
insert into user_data_hpa values ('u1', 0, 'hash_meabefir42123b423ui4b2i34b23ui4b234');

insert into user_data_hpa values ('u2', 0, 'hash_zerlekon2346324634634wesfsdfsdifusd4');

insert into user_data_hpa values ('u3', 0, 'hash_zerlekong234632dsf463r464534wes545fsdfsdifusd4');

insert into user_data_hpa values ('u4', 0, 'hash_melopo23463fdg2463sdfg4634fdwesfsddsadsdifsadusd4');

insert into user_data_hpa values ('u5', 0, 'hash_joseph2346352fdsgfdgesfsddsafdgdfggdfgdfadusd4');



```
insert into user_data_hpa values ('u1', 0, 'hash_meabefir42123b423ui4b2i34b23ui4b234');
insert into user_data_hpa values ('u2', 0, 'hash_zerlekon2346324634634wesfsdfsdifusd4');
insert into user_data_hpa values ('u3', 0, 'hash_zerlekong234632dsf463r464534wes545fsdfsdifusd4');
insert into user_data_hpa values ('u4', 0, 'hash_melopo23463fdg2463sdfg4634fdwesfsddsadsdifsadusd4');
insert into user_data_hpa values ('u5', 0, 'hash_joseph2346352fdsgfdgesfsddsafdgdfggdfgdfadusd4');
select * from user_data_hpa;
```

All Rows Fetched: 5 in 0.018 seconds

```
  USER_ID FORMS_OWNED PASSWORD_HASH
1 u1                0 hash_meabefir42123b423ui4b2i34b23ui4b234
2 u2                0 hash_zerlekon2346324634634wesfsdfsdifusd4
3 u3                0 hash_zerlekong234632dsf463r464534wes545fsdfsdifusd4
4 u4                0 hash_melopo23463fdg2463sdfg4634fdwesfsddsadsdifsadusd4
5 u5                0 hash_joseph2346352fdsgfdgesfsddsafdgdfggdfgdfadusd4
```

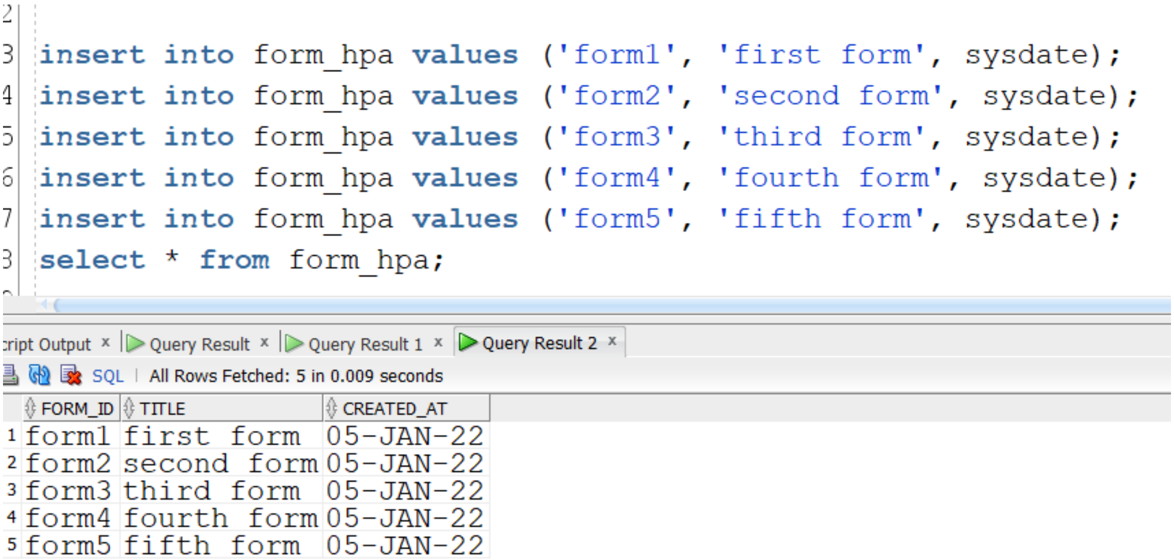
insert into form_hpa values ('form1', 'first form', sysdate);

insert into form_hpa values ('form2', 'second form', sysdate);

insert into form_hpa values ('form3', 'third form', sysdate);

insert into form_hpa values ('form4', 'fourth form', sysdate);

insert into form_hpa values ('form5', 'fifth form', sysdate);



```
insert into form_hpa values ('form1', 'first form', sysdate);
insert into form_hpa values ('form2', 'second form', sysdate);
insert into form_hpa values ('form3', 'third form', sysdate);
insert into form_hpa values ('form4', 'fourth form', sysdate);
insert into form_hpa values ('form5', 'fifth form', sysdate);
select * from form_hpa;
```

All Rows Fetched: 5 in 0.009 seconds

	FORM_ID	TITLE	CREATED_AT
1	form1	first form	05-JAN-22
2	form2	second form	05-JAN-22
3	form3	third form	05-JAN-22
4	form4	fourth form	05-JAN-22
5	form5	fifth form	05-JAN-22

insert into form_ownership_hpa values ('form1', 'u1');

insert into form_ownership_hpa values ('form2', 'u1');

insert into form_ownership_hpa values ('form1', 'u2');

insert into form_ownership_hpa values ('form2', 'u2');

insert into form_ownership_hpa values ('form1', 'u3');

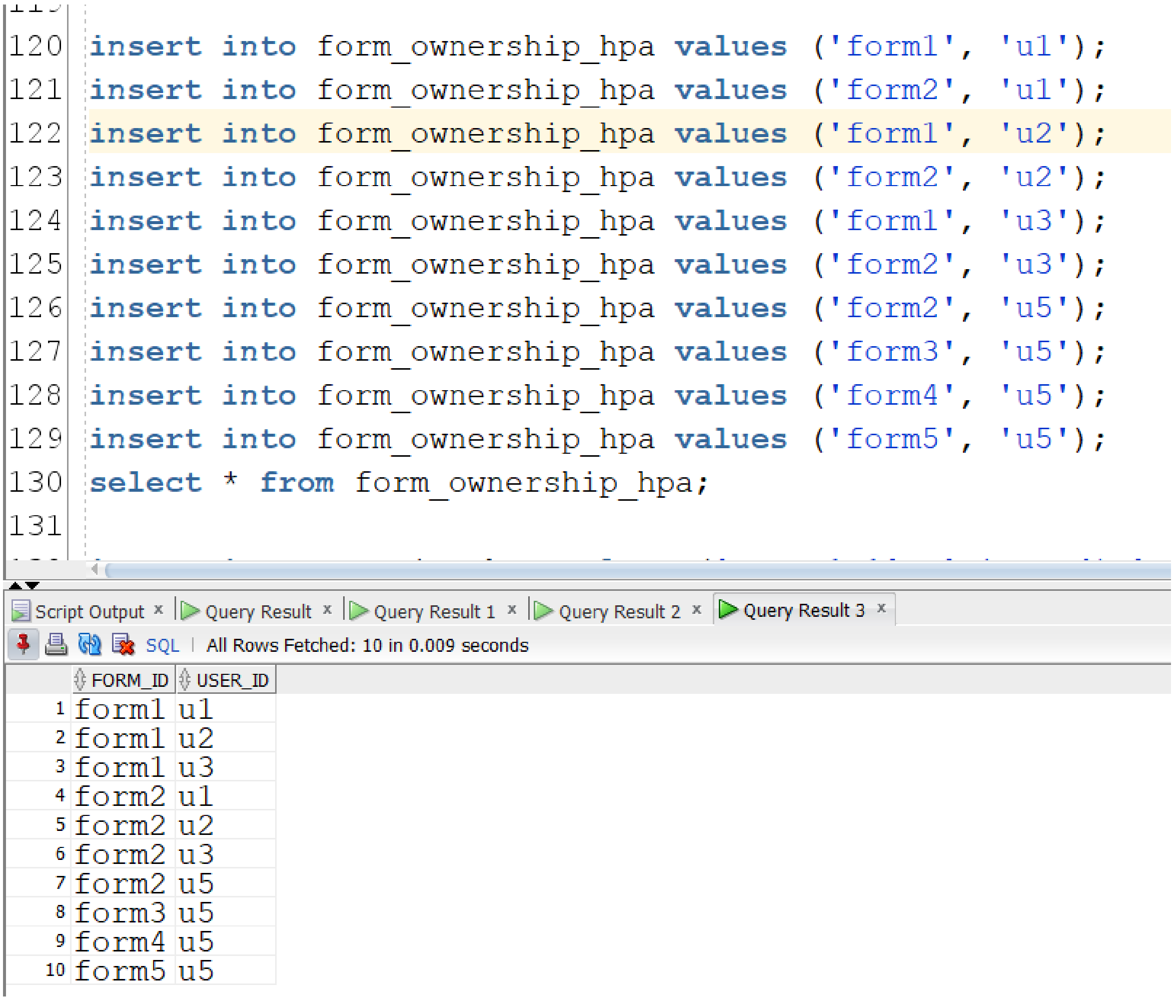
insert into form_ownership_hpa values ('form2', 'u3');

insert into form_ownership_hpa values ('form2', 'u5');

insert into form_ownership_hpa values ('form3', 'u5');

insert into form_ownership_hpa values ('form4', 'u5');

insert into form_ownership_hpa values ('form5', 'u5');



```
insert into form_ownership_hpa values ('form1', 'u1');
insert into form_ownership_hpa values ('form2', 'u1');
insert into form_ownership_hpa values ('form1', 'u2');
insert into form_ownership_hpa values ('form2', 'u2');
insert into form_ownership_hpa values ('form1', 'u3');
insert into form_ownership_hpa values ('form2', 'u3');
insert into form_ownership_hpa values ('form2', 'u5');
insert into form_ownership_hpa values ('form3', 'u5');
insert into form_ownership_hpa values ('form4', 'u5');
insert into form_ownership_hpa values ('form5', 'u5');
select * from form_ownership_hpa;
```

All Rows Fetched: 10 in 0.009 seconds

	FORM_ID	USER_ID
1	form1	u1
2	form1	u2
3	form1	u3
4	form2	u1
5	form2	u2
6	form2	u3
7	form2	u5
8	form3	u5
9	form4	u5
10	form5	u5

insert into question_hpa values ('quest1-1', 'tip radio', 'form1', 'RADIO');

insert into question_hpa values ('quest2-1', 'tip checkbox', 'form1', 'CHECKBOX');

insert into question_hpa values ('quest3-1', 'intrebare fara optiuni', 'form1', 'CHECKBOX');

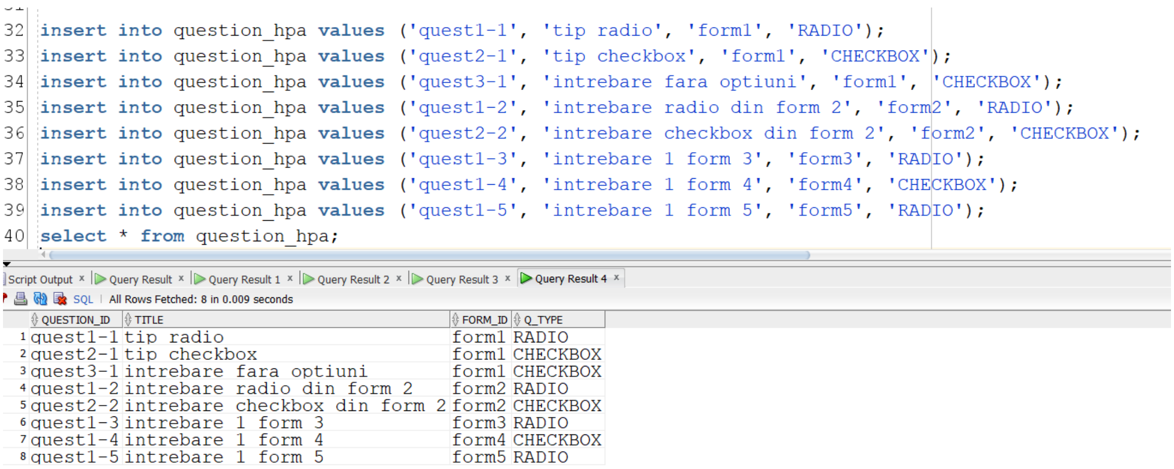
insert into question_hpa values ('quest1-2', 'intrebare radio din form 2', 'form2', 'RADIO');

insert into question_hpa values ('quest2-2', 'intrebare checkbox din form 2', 'form2', 'CHECKBOX');

insert into question_hpa values ('quest1-3', 'intrebare 1 form 3', 'form3', 'RADIO');

insert into question_hpa values ('quest1-4', 'intrebare 1 form 4', 'form4', 'CHECKBOX');

insert into question_hpa values ('quest1-5', 'intrebare 1 form 5', 'form5', 'RADIO');

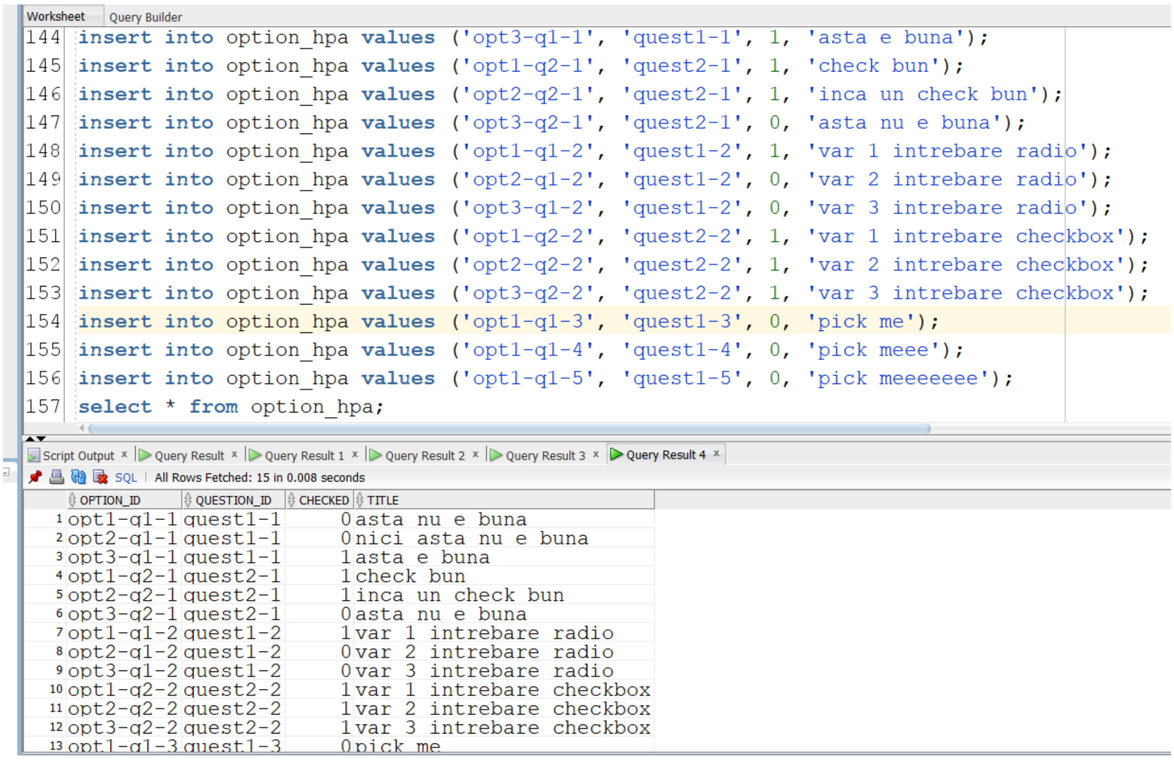


```sql
insert into question_hpa values ('quest1-1', 'tip radio', 'form1', 'RADIO');
insert into question_hpa values ('quest2-1', 'tip checkbox', 'form1', 'CHECKBOX');
insert into question_hpa values ('quest3-1', 'intrebare fara optiuni', 'form1', 'CHECKBOX');
insert into question_hpa values ('quest1-2', 'intrebare radio din form 2', 'form2', 'RADIO');
insert into question_hpa values ('quest2-2', 'intrebare checkbox din form 2', 'form2', 'CHECKBOX');
insert into question_hpa values ('quest1-3', 'intrebare 1 form 3', 'form3', 'RADIO');
insert into question_hpa values ('quest1-4', 'intrebare 1 form 4', 'form4', 'CHECKBOX');
insert into question_hpa values ('quest1-5', 'intrebare 1 form 5', 'form5', 'RADIO');
select * from question_hpa;
```

All Rows Fetched: 8 in 0.009 seconds

```
  QUESTION_ID TITLE                          FORM_ID Q_TYPE
1 quest1-1    tip radio                      form1   RADIO
2 quest2-1    tip checkbox                   form1   CHECKBOX
3 quest3-1    intrebare fara optiuni         form1   CHECKBOX
4 quest1-2    intrebare radio din form 2     form2   RADIO
5 quest2-2    intrebare checkbox din form 2  form2   CHECKBOX
6 quest1-3    intrebare 1 form 3             form3   RADIO
7 quest1-4    intrebare 1 form 4             form4   CHECKBOX
8 quest1-5    intrebare 1 form 5             form5   RADIO
```

```sql
insert into option_hpa values ('opt1-q1-1', 'quest1-1', 0, 'asta nu e buna');

insert into option_hpa values ('opt2-q1-1', 'quest1-1', 0, 'nici asta nu e buna');

insert into option_hpa values ('opt3-q1-1', 'quest1-1', 1, 'asta e buna');

insert into option_hpa values ('opt1-q2-1', 'quest2-1', 1, 'check bun');

insert into option_hpa values ('opt2-q2-1', 'quest2-1', 1, 'inca un check bun');

insert into option_hpa values ('opt3-q2-1', 'quest2-1', 0, 'asta nu e buna');

insert into option_hpa values ('opt1-q1-2', 'quest1-2', 1, 'var 1 intrebare radio');

insert into option_hpa values ('opt2-q1-2', 'quest1-2', 0, 'var 2 intrebare radio');

insert into option_hpa values ('opt3-q1-2', 'quest1-2', 0, 'var 3 intrebare radio');

insert into option_hpa values ('opt1-q2-2', 'quest2-2', 1, 'var 1 intrebare checkbox');

insert into option_hpa values ('opt2-q2-2', 'quest2-2', 1, 'var 2 intrebare checkbox');

insert into option_hpa values ('opt3-q2-2', 'quest2-2', 1, 'var 3 intrebare checkbox');

insert into option_hpa values ('opt1-q1-3', 'quest1-3', 0, 'pick me');

insert into option_hpa values ('opt1-q1-4', 'quest1-4', 0, 'pick meee');

insert into option_hpa values ('opt1-q1-5', 'quest1-5', 0, 'pick meeeeeee');
```



```
insert into option_hpa values ('opt3-q1-1', 'quest1-1', 1, 'asta e buna');
insert into option_hpa values ('opt1-q2-1', 'quest2-1', 1, 'check bun');
insert into option_hpa values ('opt2-q2-1', 'quest2-1', 1, 'inca un check bun');
insert into option_hpa values ('opt3-q2-1', 'quest2-1', 0, 'asta nu e buna');
insert into option_hpa values ('opt1-q1-2', 'quest1-2', 1, 'var 1 intrebare radio');
insert into option_hpa values ('opt2-q1-2', 'quest1-2', 0, 'var 2 intrebare radio');
insert into option_hpa values ('opt3-q1-2', 'quest1-2', 0, 'var 3 intrebare radio');
insert into option_hpa values ('opt1-q2-2', 'quest2-2', 1, 'var 1 intrebare checkbox');
insert into option_hpa values ('opt2-q2-2', 'quest2-2', 1, 'var 2 intrebare checkbox');
insert into option_hpa values ('opt3-q2-2', 'quest2-2', 1, 'var 3 intrebare checkbox');
insert into option_hpa values ('opt1-q1-3', 'quest1-3', 0, 'pick me');
insert into option_hpa values ('opt1-q1-4', 'quest1-4', 0, 'pick meee');
insert into option_hpa values ('opt1-q1-5', 'quest1-5', 0, 'pick meeeeeee');
select * from option_hpa;
```

All Rows Fetched: 15 in 0.008 seconds

	OPTION_ID	QUESTION_ID	CHECKED	TITLE
1	opt1-q1-1	quest1-1	0	asta nu e buna
2	opt2-q1-1	quest1-1	0	nici asta nu e buna
3	opt3-q1-1	quest1-1	1	asta e buna
4	opt1-q2-1	quest2-1	1	check bun
5	opt2-q2-1	quest2-1	1	inca un check bun
6	opt3-q2-1	quest2-1	0	asta nu e buna
7	opt1-q1-2	quest1-2	1	var 1 intrebare radio
8	opt2-q1-2	quest1-2	0	var 2 intrebare radio
9	opt3-q1-2	quest1-2	0	var 3 intrebare radio
10	opt1-q2-2	quest2-2	1	var 1 intrebare checkbox
11	opt2-q2-2	quest2-2	1	var 2 intrebare checkbox
12	opt3-q2-2	quest2-2	1	var 3 intrebare checkbox
13	opt1-q1-3	quest1-3	0	pick me

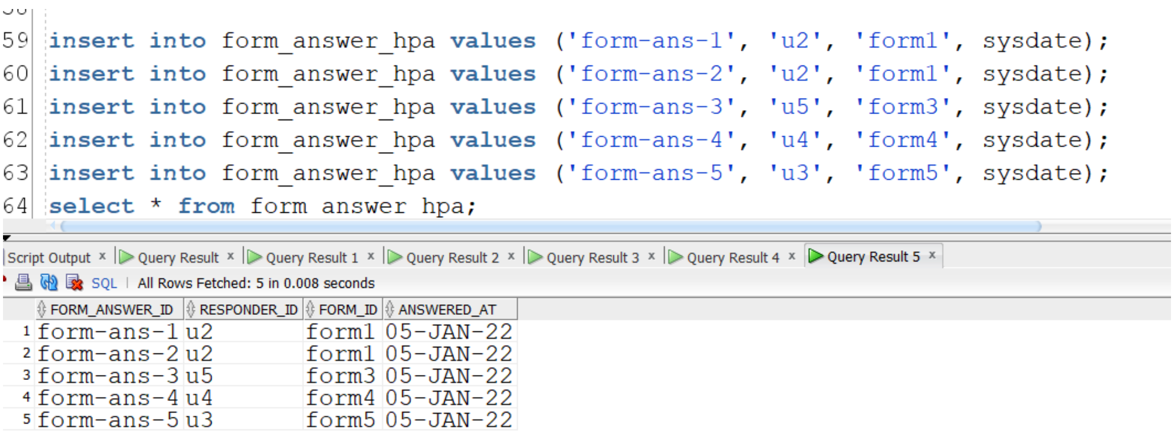
insert into form_answer_hpa values ('form-ans-1', 'u2', 'form1', sysdate);

insert into form_answer_hpa values ('form-ans-2', 'u2', 'form1', sysdate);

insert into form_answer_hpa values ('form-ans-3', 'u5', 'form3', sysdate);

insert into form_answer_hpa values ('form-ans-4', 'u4', 'form4', sysdate);

insert into form_answer_hpa values ('form-ans-5', 'u3', 'form5', sysdate);



```
insert into form_answer_hpa values ('form-ans-1', 'u2', 'form1', sysdate);
insert into form_answer_hpa values ('form-ans-2', 'u2', 'form1', sysdate);
insert into form_answer_hpa values ('form-ans-3', 'u5', 'form3', sysdate);
insert into form_answer_hpa values ('form-ans-4', 'u4', 'form4', sysdate);
insert into form_answer_hpa values ('form-ans-5', 'u3', 'form5', sysdate);
select * from form answer hpa;
```

All Rows Fetched: 5 in 0.008 seconds

	FORM_ANSWER_ID	RESPONDER_ID	FORM_ID	ANSWERED_AT
1	form-ans-1	u2	form1	05-JAN-22
2	form-ans-2	u2	form1	05-JAN-22
3	form-ans-3	u5	form3	05-JAN-22
4	form-ans-4	u4	form4	05-JAN-22
5	form-ans-5	u3	form5	05-JAN-22

insert into option_answer_hpa values ('fa1-opt1-q1', 'form-ans-1', 'opt1-q1-1', 0);

insert into option_answer_hpa values ('fa1-opt2-q1', 'form-ans-1', 'opt2-q1-1', 0);

insert into option_answer_hpa values ('fa1-opt3-q1', 'form-ans-1', 'opt3-q1-1', 1);

insert into option_answer_hpa values ('fa1-opt1-q2', 'form-ans-1', 'opt1-q2-1', 1);

insert into option_answer_hpa values ('fa1-opt2-q2', 'form-ans-1', 'opt2-q2-1', 1);

insert into option_answer_hpa values ('fa1-opt3-q2', 'form-ans-1', 'opt3-q2-1', 1);

insert into option_answer_hpa values ('fa2-opt1-q1', 'form-ans-2', 'opt1-q1-1', 0);

insert into option_answer_hpa values ('fa2-opt2-q1', 'form-ans-2', 'opt2-q1-1', 0);

insert into option_answer_hpa values ('fa2-opt3-q1', 'form-ans-2', 'opt3-q1-1', 1);

insert into option_answer_hpa values ('fa2-opt1-q2', 'form-ans-2', 'opt1-q2-1', 1);

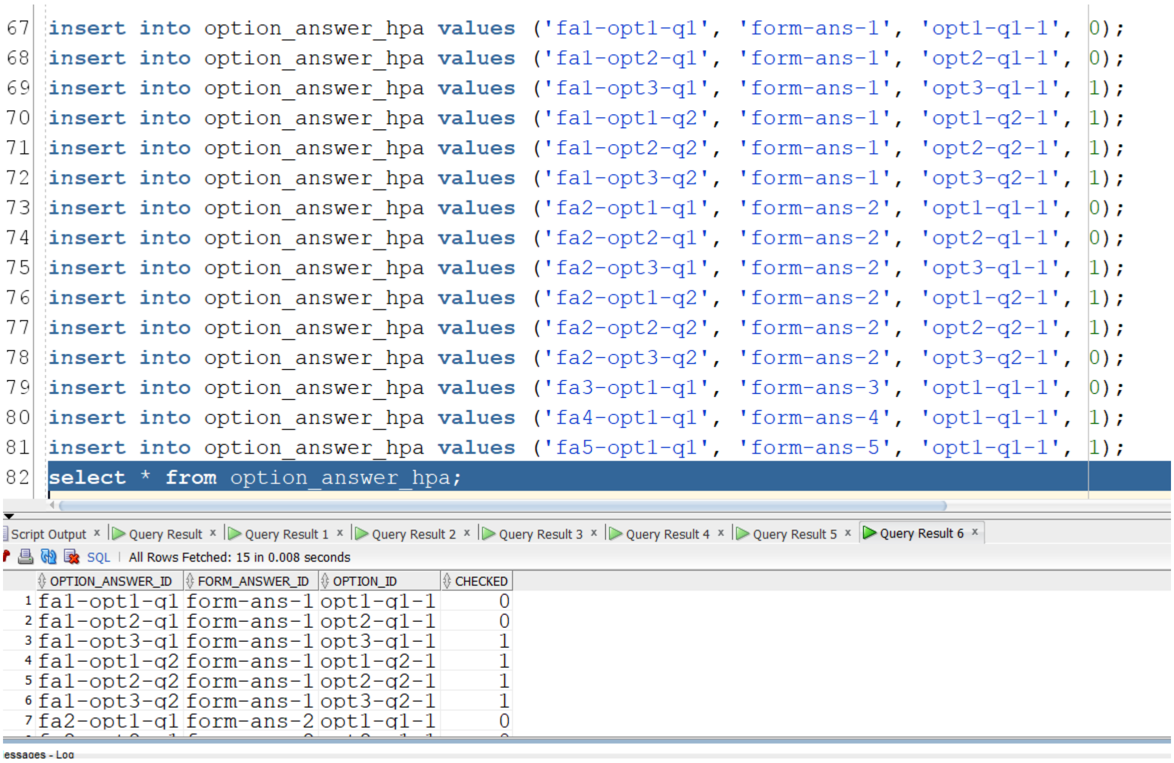
insert into option_answer_hpa values ('fa2-opt2-q2', 'form-ans-2', 'opt2-q2-1', 1);

insert into option_answer_hpa values ('fa2-opt3-q2', 'form-ans-2', 'opt3-q2-1', 0);

insert into option_answer_hpa values ('fa3-opt1-q1', 'form-ans-3', 'opt1-q1-1', 0);

insert into option_answer_hpa values ('fa4-opt1-q1', 'form-ans-4', 'opt1-q1-1', 1);

insert into option_answer_hpa values ('fa5-opt1-q1', 'form-ans-5', 'opt1-q1-1', 1);



```
insert into option_answer_hpa values ('fa1-opt1-q1', 'form-ans-1', 'opt1-q1-1', 0);
insert into option_answer_hpa values ('fa1-opt2-q1', 'form-ans-1', 'opt2-q1-1', 0);
insert into option_answer_hpa values ('fa1-opt3-q1', 'form-ans-1', 'opt3-q1-1', 1);
insert into option_answer_hpa values ('fa1-opt1-q2', 'form-ans-1', 'opt1-q2-1', 1);
insert into option_answer_hpa values ('fa1-opt2-q2', 'form-ans-1', 'opt2-q2-1', 1);
insert into option_answer_hpa values ('fa1-opt3-q2', 'form-ans-1', 'opt3-q2-1', 1);
insert into option_answer_hpa values ('fa2-opt1-q1', 'form-ans-2', 'opt1-q1-1', 0);
insert into option_answer_hpa values ('fa2-opt2-q1', 'form-ans-2', 'opt2-q1-1', 0);
insert into option_answer_hpa values ('fa2-opt3-q1', 'form-ans-2', 'opt3-q1-1', 1);
insert into option_answer_hpa values ('fa2-opt1-q2', 'form-ans-2', 'opt1-q2-1', 1);
insert into option_answer_hpa values ('fa2-opt2-q2', 'form-ans-2', 'opt2-q2-1', 1);
insert into option_answer_hpa values ('fa2-opt3-q2', 'form-ans-2', 'opt3-q2-1', 0);
insert into option_answer_hpa values ('fa3-opt1-q1', 'form-ans-3', 'opt1-q1-1', 0);
insert into option_answer_hpa values ('fa4-opt1-q1', 'form-ans-4', 'opt1-q1-1', 1);
insert into option_answer_hpa values ('fa5-opt1-q1', 'form-ans-5', 'opt1-q1-1', 1);
select * from option_answer_hpa;
```

All Rows Fetched: 15 in 0.008 seconds

```
   OPTION_ANSWER_ID FORM_ANSWER_ID OPTION_ID CHECKED
 1 fa1-opt1-q1      form-ans-1     opt1-q1-1       0
 2 fa1-opt2-q1      form-ans-1     opt2-q1-1       0
 3 fa1-opt3-q1      form-ans-1     opt3-q1-1       1
 4 fa1-opt1-q2      form-ans-1     opt1-q2-1       1
 5 fa1-opt2-q2      form-ans-1     opt2-q2-1       1
 6 fa1-opt3-q2      form-ans-1     opt3-q2-1       1
 7 fa2-opt1-q1      form-ans-2     opt1-q1-1       0
```

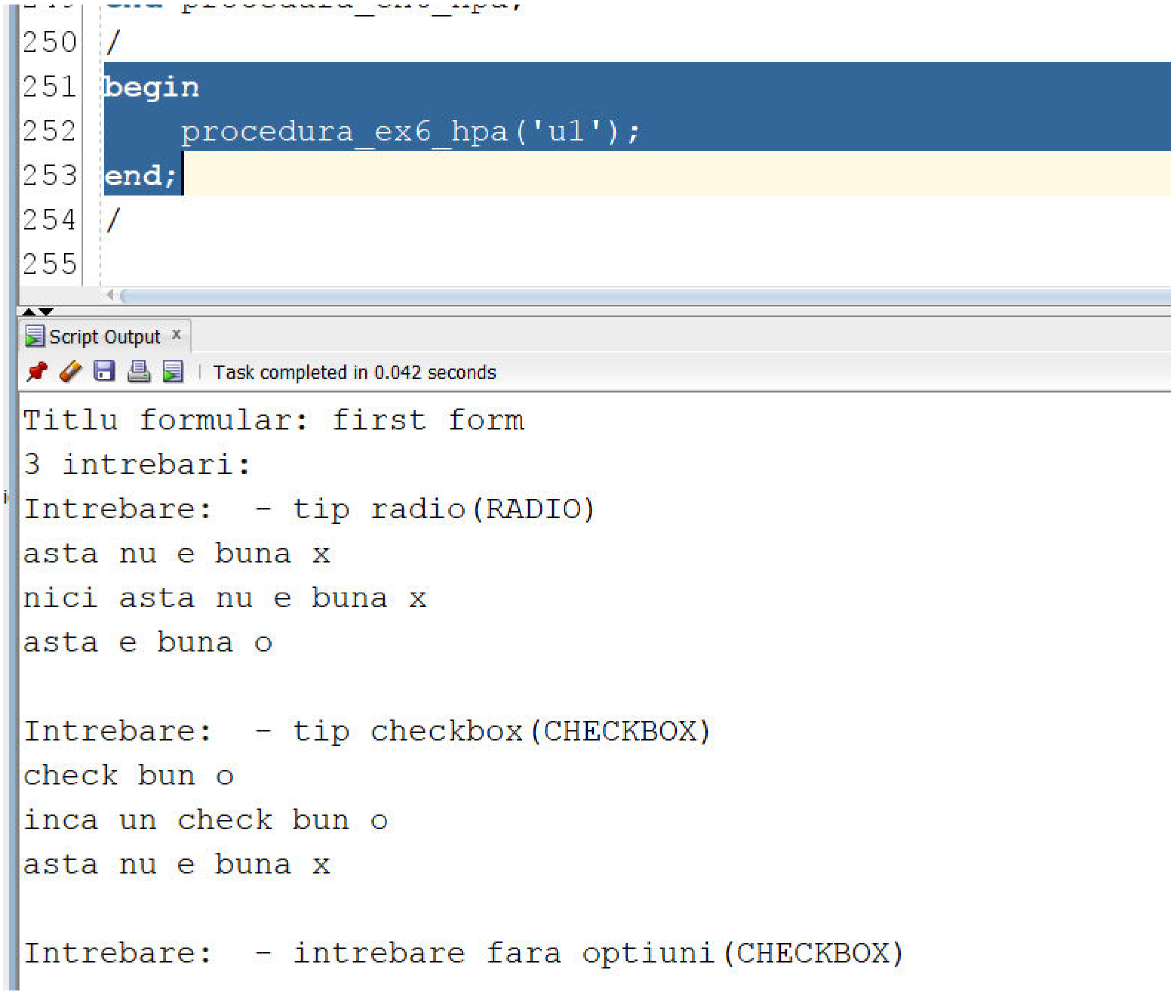
6.

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat** care să utilizeze două tipuri de colecție studiate. Apelați subprogramul.

```
-- afisati intr-un format usor de citit primul formular creat de un user
-- si intrebarile acestuia, cu variante de raspuns (id-ul user-ului este primit ca parametru)
/
create or replace type str_vec is varray(100) of varchar2(100);
/
create or replace type num_vec is varray(100) of number(30);
/

create or replace procedure procedura_ex6_hpa (id_user user_hpa.user_id%type)
is
    type str_tix is table of varchar2(100) index by binary_integer;
    type number_tix is table of number(30) index by binary_integer;

    id_formular form_hpa.form_id%type;
    titlu_formular form_hpa.title%type;
    titlu_intrebare question_hpa.title%type;
    tip_intrebare question_hpa.q_type%type;

    nume_optiune option_hpa.title%type;
    option_checked option_hpa.checked%type;

    question_ids str_vec := str_vec();
    option_ids str_tix;
begin
    select f.form_id into id_formular
    from form_ownership_hpa fo join form_hpa f on fo.form_id = f.form_id
    where id_user = user_id
    and rownum <= 1
    order by f.created_at;
```

```
select title into titlu_formular
from form_hpa
where form_id = id_formular;

select question_id
bulk collect into question_ids
from question_hpa
where form_id = id_formular;

dbms_output.put_line('Titlu formular: ' || titlu_formular);
dbms_output.put_line(question_ids.count || ' intrebari: ');

for q_idx in question_ids.first..question_ids.last loop
   select title, q_type into titlu_intrebare, tip_intrebare
   from question_hpa
   where question_ids(q_idx) = question_id;

   dbms_output.put_line('Intrebare: ' || ' - ' || titlu_intrebare || '(' || tip_intrebare || ')');
   -- optiunile de raspuns
   select option_id bulk collect into option_ids
   from option_hpa
   where question_id = question_ids(q_idx);

   if option_ids.count = 0 then
      continue;
   end if;
   for o_idx in option_ids.first..option_ids.last loop
      select title, checked into nume_optiune, option_checked
      from option_hpa
```

```
            where option_id = option_ids(o_idx);

            dbms_output.put(nume_optiune || ' ');
            if option_checked = 1 then
                dbms_output.put_line('o');
            else
                dbms_output.put_line('x');
            end if;
        end loop;
        dbms_output.put_line('');
    end loop;
end procedura_ex6_hpa;
/
begin
    procedura_ex6_hpa('u1');
end;
/
```



```
end procedura_ex6_hpa;
/
begin
    procedura_ex6_hpa('u1');
end;
/
```

Task completed in 0.042 seconds

```
Titlu formular: first form
3 intrebari:
Intrebare:  - tip radio(RADIO)
asta nu e buna x
nici asta nu e buna x
asta e buna o

Intrebare:  - tip checkbox(CHECKBOX)
check bun o
inca un check bun o
asta nu e buna x

Intrebare:  - intrebare fara optiuni(CHECKBOX)
```

7.

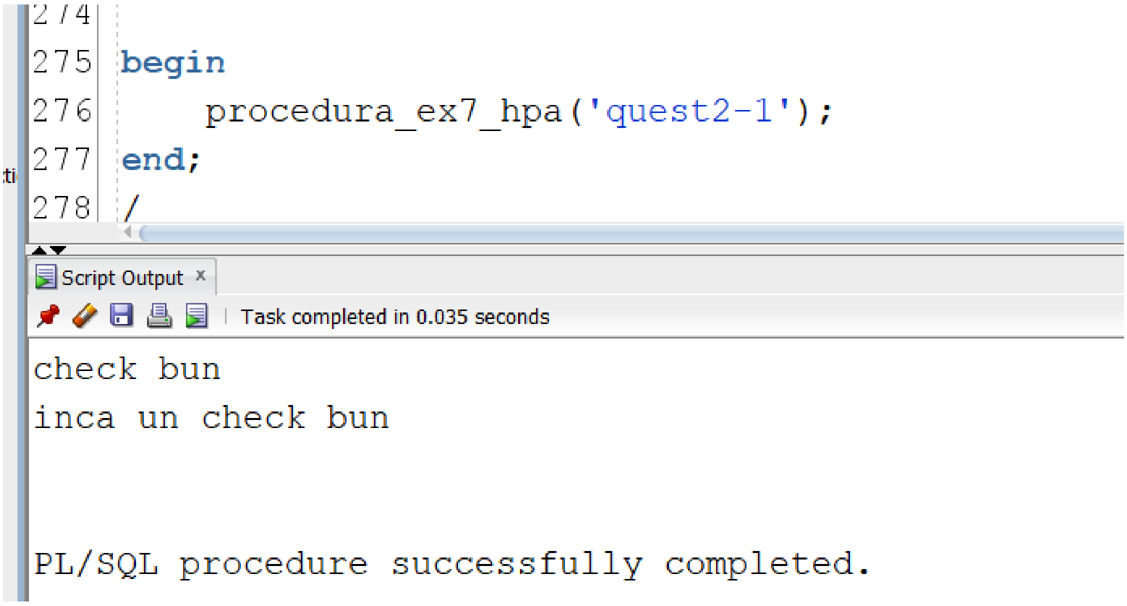
Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat** care să utilizeze un tip de cursor studiat. Apelați subprogramul.

-- afisati toate optiunile de raspuns care provin dintr-o intrebare al carui id este dat ca parametru si care

-- este varianta corecta de raspuns

```
create or replace procedure procedura_ex7_hpa (id_question question_hpa.question_id%type)
is
  cursor c_question (id_ques question_hpa.question_id%type) is
    select title
```

```
        from option_hpa
        where question_id = id_ques and checked = 1;
begin
    for q in c_question(id_question) loop
        dbms_output.put_line(q.title);
    end loop;
end procedura_ex7_hpa;


begin
    procedura_ex7_hpa('quest2-1');
end;
/
```



```
begin
    procedura_ex7_hpa('quest2-1');
end;
/
```

Task completed in 0.035 seconds

```
check bun
inca un check bun


PL/SQL procedure successfully completed.
```

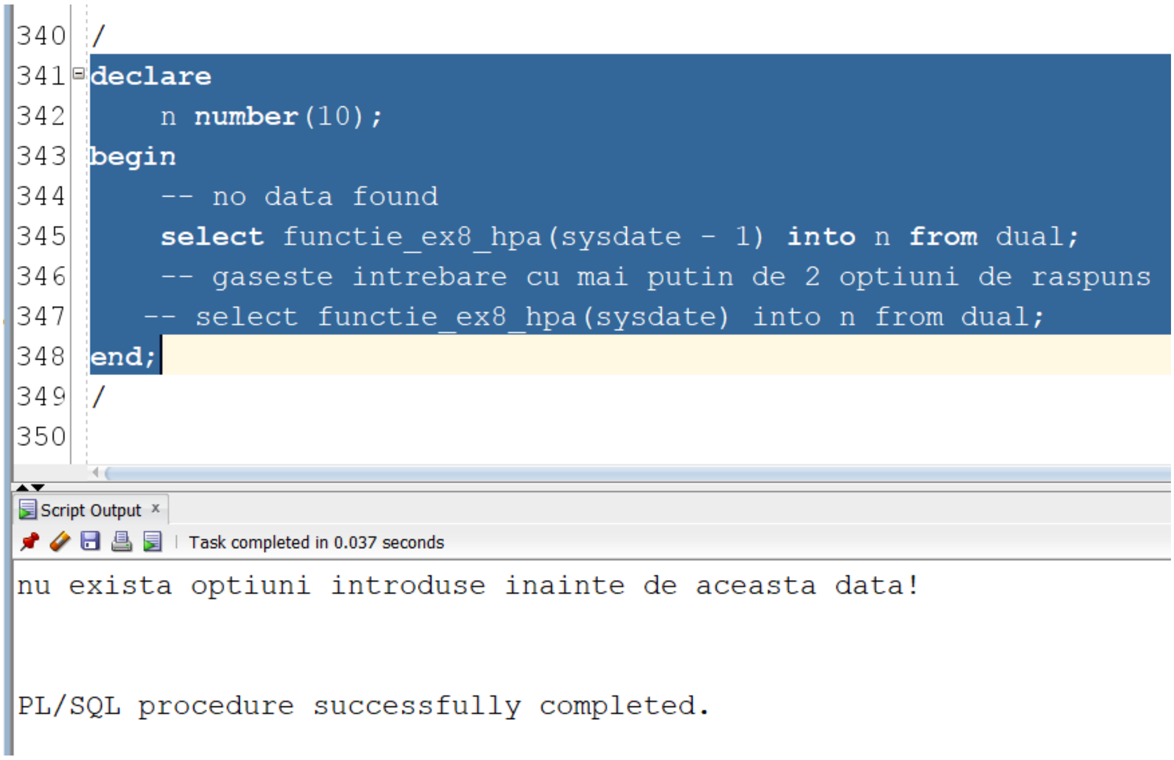
8.

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat de tip funcție** care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite. Tratați toate excepțiile care pot apărea. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

```
-- afisati, pentru fiecare optiune care apartine de un formular creat inainte de o data transmisa ca paramtetru,
-- id-ul intrebarii si id-ul formularului de care depinde
-- verifica ca toate intrebarile sa aiba cel putin 2 optiuni
-- sa se returneze numarul randurilor gasite


create or replace function functie_ex8_hpa (data_max date)
return number is
   rows_fetched number(20) := 0;
   nr_optiuni number(20);

   ex_no_data exception;
   ex_too_few_options exception;
begin
   select count(*) into rows_fetched
   from option_hpa o join question_hpa q on o.question_id = q.question_id join form_hpa f on q.form_id = f.form_id
   where f.created_at <= data_max;

   if rows_fetched = 0 then
      raise ex_no_data;
   end if;

   for q in (select ff.form_id, qq.question_id
            from question_hpa qq join form_hpa ff on ff.form_id = qq.form_id
            where ff.created_at <= data_max )
   loop
```

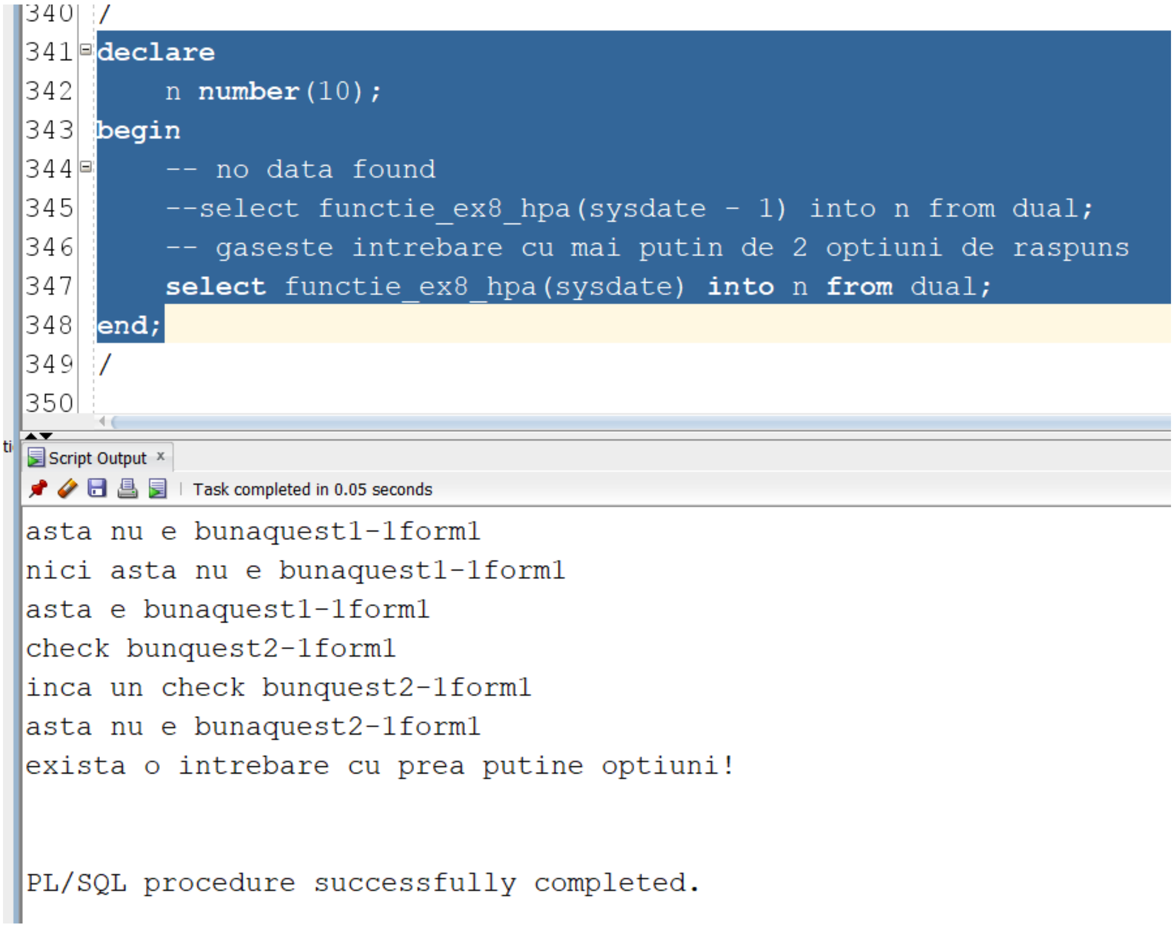
```
        select count(*) into nr_optiuni

        from option_hpa

        where question_id = q.question_id;



        if nr_optiuni < 2 then

            raise ex_too_few_options;

        end if;



        for o in (select * from option_hpa where question_id = q.question_id)

        loop

            dbms_output.put_line(o.title || o.question_id || q.form_id);

        end loop;



    end loop;



    return rows_fetched;



exception

    when ex_no_data then

        dbms_output.put_line('nu exista optiuni introduse inainte de aceasta data!');

        return -1;

    when ex_too_few_options then

        dbms_output.put_line('exista o intrebare cu prea putine optiuni!');

        return -1;

    when others then

        dbms_output.put_line('alta eroare!');

        return -1;

end functie_ex8_hpa;



/
```

```
declare
   n number(10);
begin
   -- no data found
   select functie_ex8_hpa(sysdate - 1) into n from dual;
   -- gaseste intrebare cu mai putin de 2 optiuni de raspuns
  -- select functie_ex8_hpa(sysdate) into n from dual;
end;
/
```



```
/
declare
     n number(10);
begin
     -- no data found
     select functie_ex8_hpa(sysdate - 1) into n from dual;
     -- gaseste intrebare cu mai putin de 2 optiuni de raspuns
   -- select functie_ex8_hpa(sysdate) into n from dual;
end;
/
```

```
nu exista optiuni introduse inainte de aceasta data!


PL/SQL procedure successfully completed.
```



```
/
declare
    n number(10);
begin
    -- no data found
    --select functie_ex8_hpa(sysdate - 1) into n from dual;
    -- gaseste intrebare cu mai putin de 2 optiuni de raspuns
    select functie_ex8_hpa(sysdate) into n from dual;
end;
/
```

Task completed in 0.05 seconds

```
asta nu e bunaquest1-1form1
nici asta nu e bunaquest1-1form1
asta e bunaquest1-1form1
check bunquest2-1form1
inca un check bunquest2-1form1
asta nu e bunaquest2-1form1
exista o intrebare cu prea putine optiuni!


PL/SQL procedure successfully completed.
```

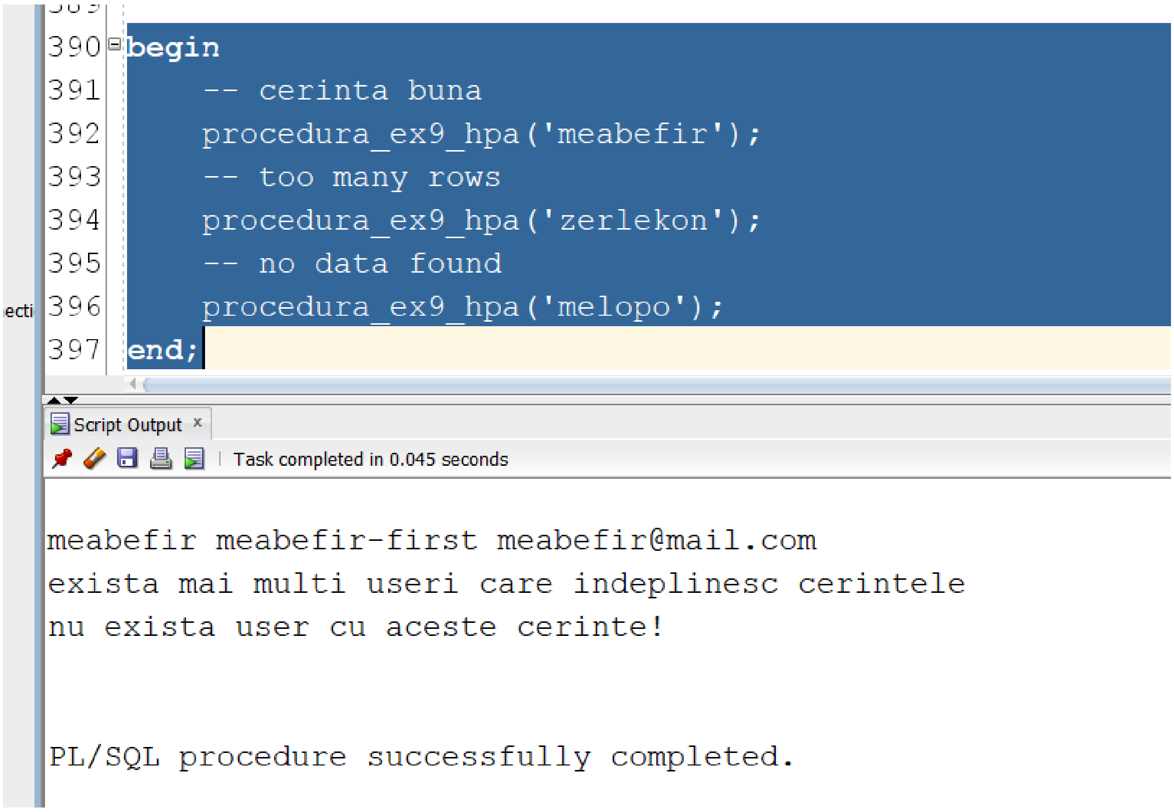
9.

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat de tip procedură** care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO_DATA_FOUND și TOO_MANY_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

-- afisati date despre userul care se numeste (nume dat parametru), care a creat cel putin 2 formulare si care are

-- cel putin 3 intrebari cu 3 sau mai multe optiuni de raspuns in total

```
create or replace procedure procedura_ex9_hpa (nume_user user_hpa.last_name%type)
is
   id_user user_hpa.user_id%type;
   type user_row is record (data user_hpa%rowtype);
   user_dataa user_row;
begin
   select u.user_id into id_user
   from user_hpa u join user_data_hpa ud on u.user_id = ud.user_id
   where ud.forms_created >= 2 and lower(u.last_name) = lower(nume_user)
   and 3 <= (select count(*)
          from question_hpa q join form_hpa f on q.form_id = f.form_id
          join form_ownership_hpa fo on fo.form_id = f.form_id
          where fo.user_id = u.user_id
          and 3 <= (select count(*)
                 from option_hpa
                 where question_id = q.question_id));

   select * into user_dataa.data
   from user_hpa
   where user_id = id_user;
   dbms_output.put_line(user_dataa.data.last_name || ' ' || user_dataa.data.first_name || ' ' || user_dataa.data.email);
exception
   when no_data_found then
      dbms_output.put_line('nu exista user cu aceste cerinte!');
   when too_many_rows then
      dbms_output.put_line('exista mai multi useri care indeplinesc cerintele');
   when others then
      dbms_output.put_line('exceptie neprevazuta!');
end procedura_ex9_hpa;
```



```
begin
    -- cerinta buna
    procedura_ex9_hpa('meabefir');
    -- too many rows
    procedura_ex9_hpa('zerlekon');
    -- no data found
    procedura_ex9_hpa('melopo');
end;
```

Task completed in 0.045 seconds

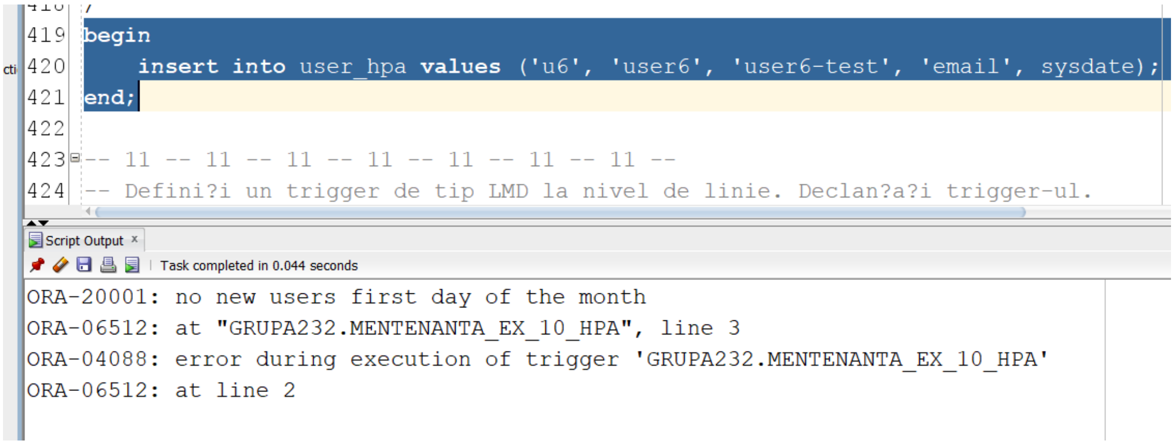
```
meabefir meabefir-first meabefir@mail.com
exista mai multi useri care indeplinesc cerintele
nu exista user cu aceste cerinte!


PL/SQL procedure successfully completed.
```

10.

Definiți un *trigger* de tip LMD la nivel de comandă. Declanșați *trigger*-ul

```
-- nu permite inregistrarea de useri noi prima zi din luna intre orele 22 si 24


create or replace trigger mentenanta_ex_10_hpa

before insert on user_hpa


begin

   if to_char(sysdate, 'dd') = '01' or (to_char(sysdate,'hh24') between 22 AND 24) then

      RAISE_APPLICATION_ERROR(-20001, 'no new users first day of the month');

   end if;

end;

begin
```

```
   insert into user_hpa values ('u6', 'user6', 'user6-test', 'email', sysdate);
end;
```



```
/
begin
    insert into user_hpa values ('u6', 'user6', 'user6-test', 'email', sysdate);
end;

-- 11 -- 11 -- 11 -- 11 -- 11 -- 11 -- 11 --
-- Defini?i un trigger de tip LMD la nivel de linie. Declan?a?i trigger-ul.
```

Task completed in 0.044 seconds

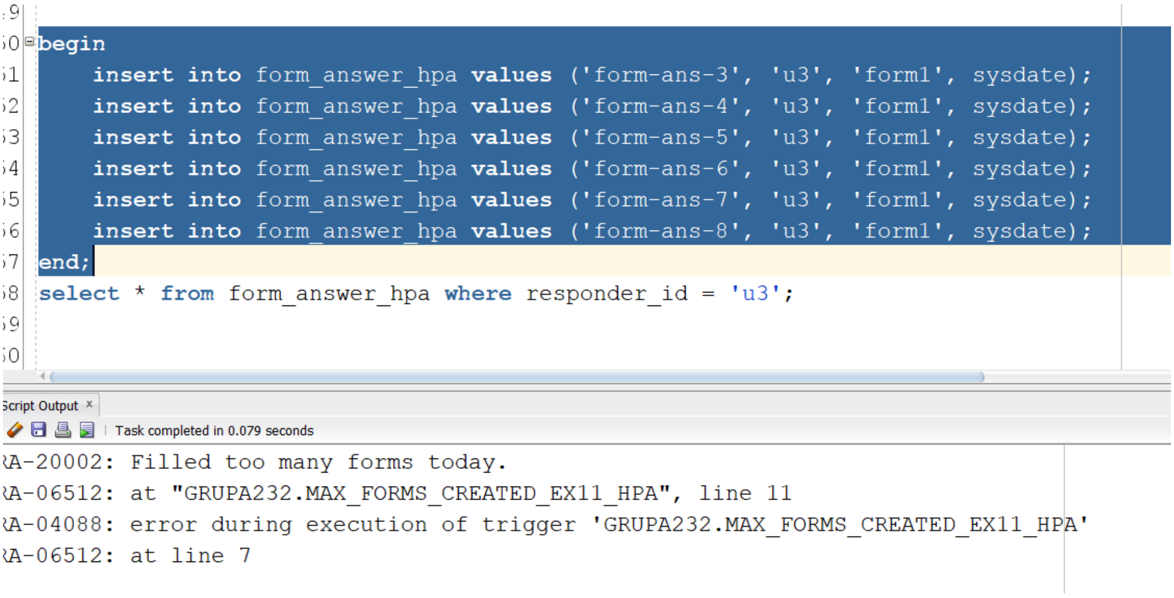
```
ORA-20001: no new users first day of the month
ORA-06512: at "GRUPA232.MENTENANTA_EX_10_HPA", line 3
ORA-04088: error during execution of trigger 'GRUPA232.MENTENANTA_EX_10_HPA'
ORA-06512: at line 2
```

11.

Definiți un *trigger* de tip LMD la nivel de linie. Declanșați *trigger*-ul.

```
-- trigger care nu mai permite completarea formularelor de catre
-- useri care au completat 5 formulare in ziua respectiva

create or replace trigger max_forms_created_ex11_hpa
before insert on form_answer_hpa
for each row
declare
   zi_azi date := to_date(sysdate, 'yyyy-mm-dd');
   forms_today number(10);
   id_responder user_hpa.user_id%type;
begin
   select count(*) into forms_today
   from form_answer_hpa
   where responder_id = :NEW.responder_id and to_date(answered_at, 'yyyy-mm-dd') = zi_azi;
```

```
    if forms_today >= 5 then
        RAISE_APPLICATION_ERROR(-20002, 'Filled too many forms today.');
    end if;
    dbms_output.put_line(:NEW.responder_id || ' filled in a form');
end;
/


begin
    insert into form_answer_hpa values ('form-ans-3', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-4', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-5', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-6', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-7', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-8', 'u3', 'form1', sysdate);
end;
```



```
begin
    insert into form_answer_hpa values ('form-ans-3', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-4', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-5', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-6', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-7', 'u3', 'form1', sysdate);
    insert into form_answer_hpa values ('form-ans-8', 'u3', 'form1', sysdate);
end;
select * from form_answer_hpa where responder_id = 'u3';
```

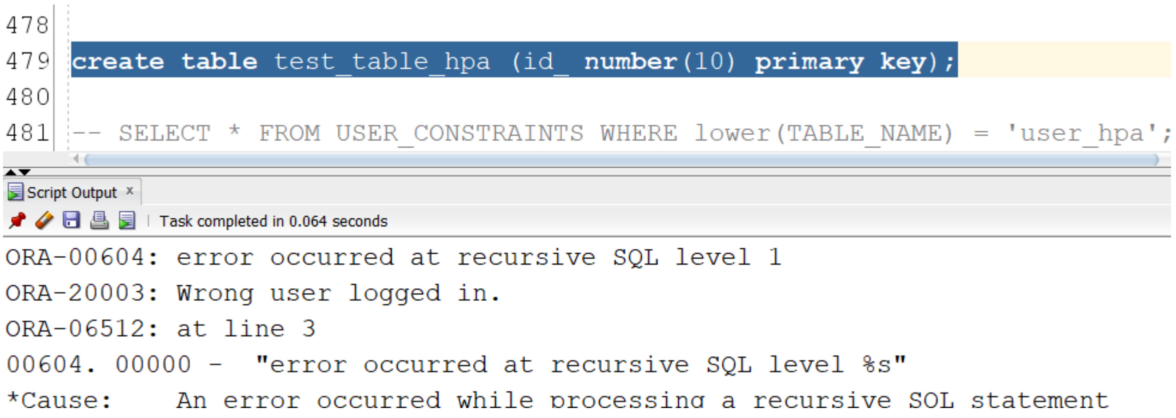
Task completed in 0.079 seconds

```
RA-20002: Filled too many forms today.
RA-06512: at "GRUPA232.MAX_FORMS_CREATED_EX11_HPA", line 11
RA-04088: error during execution of trigger 'GRUPA232.MAX_FORMS_CREATED_EX11_HPA'
RA-06512: at line 7
```

12.

Definiți un *trigger* de tip LDD. Declanșați *trigger*-ul.

```
-- daca userul logat nu este cel corect, sa dea eroare

create or replace trigger trigger_ex12_hpa

before create or drop or alter on schema


begin

   if SYS.LOGIN_USER != 'HPA' then

       RAISE_APPLICATION_ERROR(-20003, 'Wrong user logged in.');

   end if;

end;

/


create table test_table_hpa (id_ number(10) primary key);
```



```

create table test_table_hpa (id_ number(10) primary key);

-- SELECT * FROM USER_CONSTRAINTS WHERE lower(TABLE_NAME) = 'user_hpa';
```

```
ORA-00604: error occurred at recursive SQL level 1
ORA-20003: Wrong user logged in.
ORA-06512: at line 3
00604. 00000 -  "error occurred at recursive SQL level %s"
*Cause:    An error occurred while processing a recursive SQL statement
```

13.

Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.

```
create or replace package pachet_ex13_hpa as
```

```
type str_vec is varray(100) of varchar2(100);
type num_vec is varray(100) of number(30);


    -- ex 6 procedure
    procedure procedura_ex6_hpa (id_user user_hpa.user_id%type);
    -- ex 7 procedure
    procedure procedura_ex7_hpa (id_question question_hpa.question_id%type);
    -- ex 8 function
    function functie_ex8_hpa (data_max date) return number;
    -- ex 9 procedure
    procedure procedura_ex9_hpa (nume_user user_hpa.last_name%type);

end pachet_ex13_hpa;

create or replace package body pachet_ex13_hpa is


-- 6 -- 6 -- 6 -- 6 -- 6 -- 6 -- 6 -- 6 --
-- Formula?i în limbaj natural o problem? pe care s? o rezolva?i folosind un subprogram stocat care
-- s? utilizeze dou? tipuri de colec?ie studiate. Apela?i subprogramul.
--
-- afisati intr-un format usor de citit primul formular creat de un user
-- si intrebarile acestuia, cu variante de raspuns (id-ul user-ului este primit ca parametru)
procedure procedura_ex6_hpa (id_user user_hpa.user_id%type)
is
    type str_tix is table of varchar2(100) index by binary_integer;
    type number_tix is table of number(30) index by binary_integer;


    id_formular form_hpa.form_id%type;
    titlu_formular form_hpa.title%type;
```

```
   titlu_intrebare question_hpa.title%type;
   tip_intrebare question_hpa.q_type%type;

   nume_optiune option_hpa.title%type;
   option_checked option_hpa.checked%type;

   question_ids str_vec := str_vec();
   option_ids str_tix;
begin
   select f.form_id into id_formular
   from form_ownership_hpa fo join form_hpa f on fo.form_id = f.form_id
   where id_user = user_id
   and rownum <= 1
   order by f.created_at;

   select title into titlu_formular
   from form_hpa
   where form_id = id_formular;

   select question_id
   bulk collect into question_ids
   from question_hpa
   where form_id = id_formular;

   dbms_output.put_line('Titlu formular: ' || titlu_formular);
   dbms_output.put_line(question_ids.count || ' intrebari: ');

   for q_idx in question_ids.first..question_ids.last loop
      select title, q_type into titlu_intrebare, tip_intrebare
      from question_hpa
      where question_ids(q_idx) = question_id;

      dbms_output.put_line('Intrebare: ' || ' - ' || titlu_intrebare || '(' || tip_intrebare || ')');
      -- optiunile de raspuns
      select option_id bulk collect into option_ids
      from option_hpa
      where question_id = question_ids(q_idx);

      if option_ids.count = 0 then
         continue;
      end if;
      for o_idx in option_ids.first..option_ids.last loop
         select title, checked into nume_optiune, option_checked
         from option_hpa
         where option_id = option_ids(o_idx);

         dbms_output.put(nume_optiune || ' ');
         if option_checked = 1 then
            dbms_output.put_line('o');
         else
            dbms_output.put_line('x');
         end if;
      end loop;
      dbms_output.put_line('');
   end loop;
end procedura_ex6_hpa;

-- 7 -- 7 -- 7 -- 7 -- 7 -- 7 -- 7 -- 7 --
-- Formula?i în limbaj natural o problem? pe care s? o rezolva?i folosind un subprogram stocat care
-- s? utilizeze un tip de cursor studiat. Apela?i subprogramul.
```

```
--
-- afisati toate optiunile de raspuns care provin dintr-o intrebare al carui id este dat ca parametru si care
-- este considerat raspuns corect

procedure procedura_ex7_hpa (id_question question_hpa.question_id%type)
is
   cursor c_question (id_ques question_hpa.question_id%type) is
      select title
      from option_hpa
      where question_id = id_ques and checked = 1;
begin
   for q in c_question(id_question) loop
      dbms_output.put_line(q.title);
   end loop;
end procedura_ex7_hpa;

-- 8 -- 8 -- 8 -- 8 -- 8 -- 8 -- 8 -- 8 -- 8 --
-- Formula?i în limbaj natural o problem? pe care s? o rezolva?i folosind un
-- subprogram stocat de tip func?ie care s? utilizeze într-o singur? comand? SQL 3 dintre
-- tabelele definite. Trata?i toate excep?iile care pot ap?rea. Apela?i subprogramul astfel încât
-- s? eviden?ia?i toate cazurile tratate.
--
-- afisati, pentru fiecare optiune care apartine de un formular creat inainte de o data transmisa ca paramtetru,
-- id-ul intrebarii si id-ul formularului de care depinde
-- verifica ca toate intrebarile sa aiba cel putin 2 optiuni
-- sa se returneze numarul randurilor gasite

function functie_ex8_hpa (data_max date)
return number is
```

```
   rows_fetched number(20) := 0;
   nr_optiuni number(20);

   ex_no_data exception;
   ex_too_few_options exception;
begin
   select count(*) into rows_fetched
   from option_hpa o join question_hpa q on o.question_id = q.question_id join form_hpa f on q.form_id = f.form_id
   where f.created_at <= data_max;

   if rows_fetched = 0 then
      raise ex_no_data;
   end if;

   for q in (select ff.form_id, qq.question_id
            from question_hpa qq join form_hpa ff on ff.form_id = qq.form_id
            where ff.created_at <= data_max )
   loop
      select count(*) into nr_optiuni
      from option_hpa
      where question_id = q.question_id;

      if nr_optiuni < 2 then
         raise ex_too_few_options;
      end if;

      for o in (select * from option_hpa where question_id = q.question_id)
      loop
         dbms_output.put_line(o.title || o.question_id || q.form_id);
      end loop;
```

```
    end loop;

    return rows_fetched;

exception
    when ex_no_data then
        dbms_output.put_line('nu exista optiuni introduse inainte de aceasta data!');
        return -1;
    when ex_too_few_options then
        dbms_output.put_line('exista o intrebare cu prea putine optiuni!');
        return -1;
    when others then
        dbms_output.put_line('alta eroare!');
        return -1;
end functie_ex8_hpa;

-- 9 -- 9 -- 9 -- 9 -- 9 -- 9 -- 9 -- 9 --
-- Formula?i în limbaj natural o problem? pe care s? o rezolva?i folosind un
-- subprogram stocat de tip procedur? care s? utilizeze într-o singur? comand? SQL 5 dintre tabelele definite. Trata?i toate excep?iile care pot ap?rea, incluzând excep?iile NO_DATA_FOUND ?i TOO_MANY_ROWS.
-- Apela?i subprogramul astfel încât s? eviden?ia?i toate cazurile tratate.
--
-- afisati date despre userul care se numeste (nume dat parametru), care a creat cel putin 2 formulare si care are
-- cel putin 3 intrebari cu 3 sau mai multe optiuni de raspuns in total

procedure procedura_ex9_hpa (nume_user user_hpa.last_name%type)
is
    id_user user_hpa.user_id%type;
    type user_row is record (data user_hpa%rowtype);
```

```
   user_dataa user_row;
begin
   select u.user_id into id_user
   from user_hpa u join user_data_hpa ud on u.user_id = ud.user_id
   where ud.forms_owned >= 2 and lower(u.last_name) = lower(nume_user)
   and 3 <= (select count(*)
            from question_hpa q join form_hpa f on q.form_id = f.form_id
            join form_ownership_hpa fo on fo.form_id = f.form_id
            where fo.user_id = u.user_id
            and 3 <= (select count(*)
                     from option_hpa
                     where question_id = q.question_id));

   select * into user_dataa.data
   from user_hpa
   where user_id = id_user;
   dbms_output.put_line(user_dataa.data.last_name || ' ' || user_dataa.data.first_name || ' ' ||
user_dataa.data.email);
exception
   when no_data_found then
      dbms_output.put_line('nu exista user cu aceste cerinte!');
   when too_many_rows then
      dbms_output.put_line('exista mai multi useri care indeplinesc cerintele');
   when others then
      dbms_output.put_line('exceptie neprevazuta!');
end procedura_ex9_hpa;

end pachet_ex13_hpa;
```

14.

Definiți un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri).

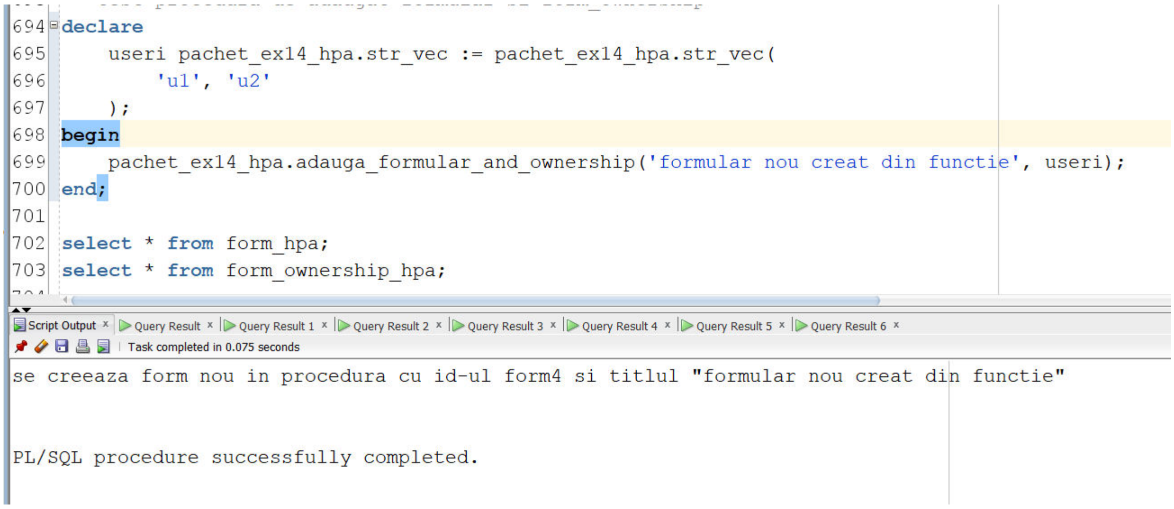
```
create or replace package pachet_ex14_hpa as


type info_optiune is record (title option_hpa.title%type,
                             checked option_hpa.checked%type);
type str_vec is varray(100) of varchar2(100);
type num_vec is varray(100) of number(30);
type info_optiune_vec is varray(100) of info_optiune;


-- creeaza o procedura care permite adaugarea de formulare si
-- oferirea mai multor utilizatori ownership catre acest formular
procedure adauga_formular_and_ownership (titlu_formular form_hpa.title%type,
                                         useri str_vec);


-- creeaza o procedura care permite sa adaugi o intrebare noua + optiuni penbtru aceasta
-- pt un formular al carui id este dat ca parametru
-- de asemenea se transmit ca parametrii titlul intrebarii si variantele ce raspuns
procedure adauga_intrebare_cu_optiuni (id_formular form_hpa.form_id%type,
                            titlu_intrebare question_hpa.title%type,
                            tip_intrebare question_hpa.q_type%type,
                            info_optiuni info_optiune_vec);


-- functie care ret 1 daca raspunsul de formular primit ca parametru este corect si 0 daca nu este
function form_answer_corect (id_rasp_form form_answer_hpa.form_answer_id%type) return
number;
```

```
-- functie care returneaza cate formulare a completat corect un utilizator dat ca parametru

function nr_formulare_corecte (id_user user_hpa.user_id%type) return number;


-- creeaza o procedura care sterge utilizatorii care nu au completat niciun formular corect

procedure st_ut_fara_rasp_corect;


end pachet_ex14_hpa;

/


-----------------------------------------------------------------------------------------------

-----------------------------------------------------------------------------------------------

-----------------------------------------------------------------------------------------------

create or replace package body pachet_ex14_hpa is


---------------------------------------------------------------------------

---------------------------------------------------------------------------

procedure adauga_formular_and_ownership (titlu_formular form_hpa.title%type,

                              useri str_vec)

is

   nr_forms_old number(10);

   new_form_id form_hpa.form_id%type;

begin

   select count(*) into nr_forms_old

   from form_hpa;

   nr_forms_old := nr_forms_old + 1;


   new_form_id := 'form' || to_char(nr_forms_old);


   dbms_output.put_line('se creeaza form nou in procedura cu id-ul ' || new_form_id || ' si titlul "'
|| titlu_formular || '"');
```

```
   insert into form_hpa values (new_form_id, titlu_formular, sysdate);

   for idx in useri.first..useri.last loop
      insert into form_ownership_hpa values (new_form_id, useri(idx));
   end loop;

end adauga_formular_and_ownership;

-----------------------------------------------------------------------------
---------------------------------------------------------------------------
procedure adauga_intrebare_cu_optiuni (id_formular form_hpa.form_id%type,
                     titlu_intrebare question_hpa.title%type,
                     tip_intrebare question_hpa.q_type%type,
                     info_optiuni info_optiune_vec)
is
   numar_formular varchar2(30);
   nr_intrebari_alr number(10);
   id_quest question_hpa.question_id%type;
   current_option_id option_hpa.option_id%type;
begin
   -- iau nr formularului pt a construi id-ul intrebarii
   numar_formular := substr(id_formular,5,30);
   select count(*) into nr_intrebari_alr
   from question_hpa
   where form_id = id_formular;
   nr_intrebari_alr := nr_intrebari_alr + 1;

   id_quest := 'quest' || to_char(nr_intrebari_alr) || '-' || numar_formular;

   dbms_output.put_line('se creeaza intrebare noua cu id-ul ' || id_quest || ' pt formularul ' || id_formular);
   insert into question_hpa values (id_quest, titlu_intrebare, id_formular, tip_intrebare);


   for opt_idx in info_optiuni.first..info_optiuni.last loop

      -- option id template

      -- opt1-q1-1

      current_option_id := 'opt' || to_char(opt_idx) || '-q' || to_char(nr_intrebari_alr) || '-' || numar_formular;

      insert into option_hpa values (current_option_id, id_quest,

                     info_optiuni(opt_idx).checked, info_optiuni(opt_idx).title);

      -- dbms_output.put_line(current_option_id);

   end loop;

end adauga_intrebare_cu_optiuni;


----------------------------------------------------------------------------

----------------------------------------------------------------------------

function form_answer_corect (id_rasp_form form_answer_hpa.form_answer_id%type) return number

is

   id_formular form_hpa.form_id%type;

   is_correct_option_checked number(1);

   cursor options_in_form_answer is select * from

                           option_answer_hpa

                           where form_answer_id = id_rasp_form;

begin

   select form_id into id_formular from form_answer_hpa where form_answer_id = id_rasp_form;


   -- cursor pt fiecare optiune in raspuns

   for o in options_in_form_answer loop

      select checked into is_correct_option_checked

      from option_hpa where option_id = o.option_id;

      if is_correct_option_checked != o.checked then
```

```
            return 0;
        end if;
    end loop;

    return 1;
end form_answer_corect;

----------------------------------------------------------------------------
----------------------------------------------------------------------------
function nr_formulare_corecte(id_user user_hpa.user_id%type) return number
is
    nr_ret number(10) := 0;
    cursor c_form_completate is select * from form_answer_hpa where responder_id = id_user;
begin
    for fa in c_form_completate loop
        dbms_output.put_line(fa.form_answer_id || ' completat de ' || fa.responder_id);
        if form_answer_corect(fa.form_answer_id) = 1 then
            nr_ret := nr_ret + 1;
        end if;
    end loop;
    dbms_output.put_line(id_user || ' a completat ' || nr_ret || ' formulare corect!');
    return nr_ret;
end nr_formulare_corecte;

----------------------------------------------------------------------------
----------------------------------------------------------------------------
procedure st_ut_fara_rasp_corect
is
    users_to_delete str_vec := str_vec();
    cursor c_users is select user_id from user_hpa;
begin

    for u in c_users loop
        if nr_formulare_corecte(u.user_id) = 0 then
            users_to_delete.extend();
            users_to_delete(users_to_delete.last) := u.user_id;
        end if;
    end loop;

    if users_to_delete.count != 0 then
        for idx in users_to_delete.first..users_to_delete.last loop
            dbms_output.put_line('s-a sters userul cu id-ul ' || users_to_delete(idx));
            delete from user_hpa where user_id = users_to_delete(idx);
        end loop;
    end if;

end st_ut_fara_rasp_corect;

end pachet_ex14_hpa;
/

-- test pachet_ex14
------------------------------ TESTE -------------------------------------
--------------------$$$----------------------------------------$$$--------
-- test procedura de adaugat formular si form_ownership
declare
    useri pachet_ex14_hpa.str_vec := pachet_ex14_hpa.str_vec(
        'u1', 'u2'
    );
```

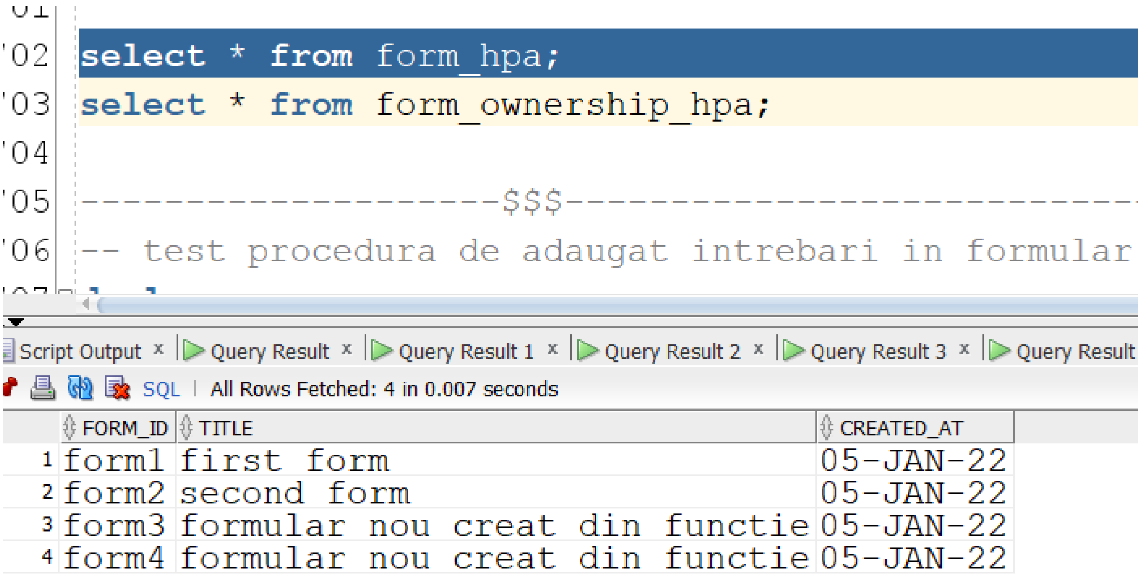
```
begin
    pachet_ex14_hpa.adauga_formular_and_ownership('formular nou creat din functie', useri);
end;


select * from form_hpa;
select * from form_ownership_hpa;
```



```
declare
    useri pachet_ex14_hpa.str_vec := pachet_ex14_hpa.str_vec(
        'u1', 'u2'
    );
begin
    pachet_ex14_hpa.adauga_formular_and_ownership('formular nou creat din functie', useri);
end;

select * from form_hpa;
select * from form_ownership_hpa;
```

Task completed in 0.075 seconds

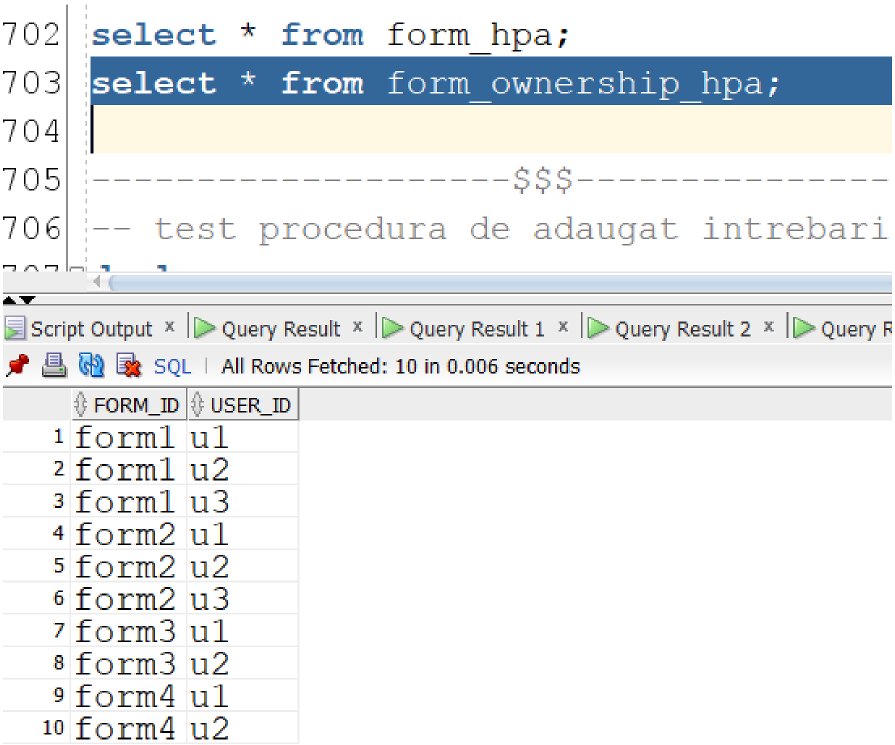
```
se creeaza form nou in procedura cu id-ul form4 si titlul "formular nou creat din functie"


PL/SQL procedure successfully completed.
```



```
select * from form_hpa;
select * from form_ownership_hpa;

-------------------$$$---------------------------
-- test procedura de adaugat intrebari in formular
```

All Rows Fetched: 4 in 0.007 seconds

	FORM_ID	TITLE	CREATED_AT
1	form1	first form	05-JAN-22
2	form2	second form	05-JAN-22
3	form3	formular nou creat din functie	05-JAN-22
4	form4	formular nou creat din functie	05-JAN-22



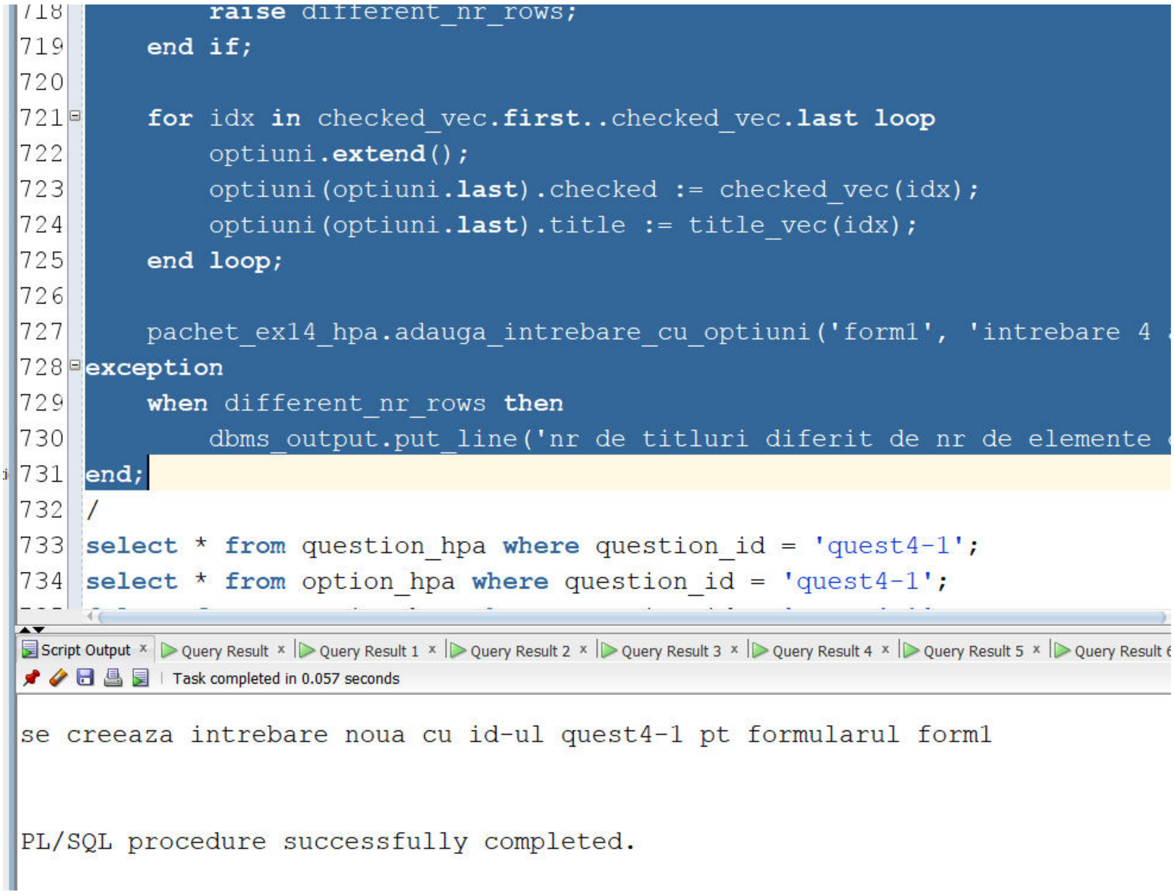
```
select * from form_hpa;
select * from form_ownership_hpa;

--------------------$$$---------------
-- test procedura de adaugat intrebari
```

All Rows Fetched: 10 in 0.006 seconds

	FORM_ID	USER_ID
1	form1	u1
2	form1	u2
3	form1	u3
4	form2	u1
5	form2	u2
6	form2	u3
7	form3	u1
8	form3	u2
9	form4	u1
10	form4	u2

```
--------------------$$$--------------------------------------$$$--------------

-- test procedura de adaugat intrebari in formular

declare

   optiuni pachet_ex14_hpa.info_optiune_vec := pachet_ex14_hpa.info_optiune_vec();

   checked_vec pachet_ex14_hpa.num_vec := pachet_ex14_hpa.num_vec(

      1, 0, 1

   );

   title_vec pachet_ex14_hpa.str_vec := pachet_ex14_hpa.str_vec(

      'prima optiune in intrebare prin functie', 'a doua optiune in intrebare prin functie', 'a  treia optiune in intrebare prin functie'

   );

   different_nr_rows exception;

begin

   if title_vec.count != checked_vec.count then

      raise different_nr_rows;

   end if;
```

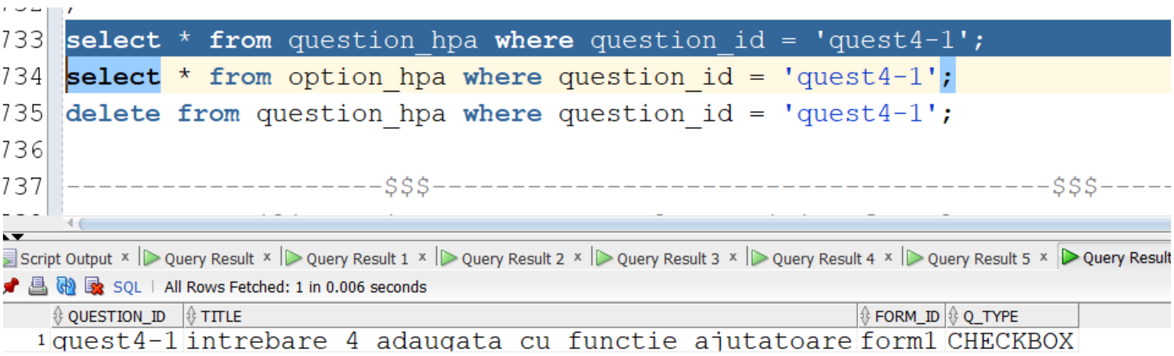
```
    for idx in checked_vec.first..checked_vec.last loop
        optiuni.extend();
        optiuni(optiuni.last).checked := checked_vec(idx);
        optiuni(optiuni.last).title := title_vec(idx);
    end loop;

    pachet_ex14_hpa.adauga_intrebare_cu_optiuni('form1', 'intrebare 4 adaugata cu functie ajutatoare', 'CHECKBOX', optiuni);
exception
    when different_nr_rows then
        dbms_output.put_line('nr de titluri diferit de nr de elemente din checked!');
end;
/
select * from question_hpa where question_id = 'quest4-1';
select * from option_hpa where question_id = 'quest4-1';
delete from question_hpa where question_id = 'quest4-1';
```



```
            raise different_nr_rows;
        end if;

        for idx in checked_vec.first..checked_vec.last loop
            optiuni.extend();
            optiuni(optiuni.last).checked := checked_vec(idx);
            optiuni(optiuni.last).title := title_vec(idx);
        end loop;

        pachet_ex14_hpa.adauga_intrebare_cu_optiuni('form1', 'intrebare 4
exception
        when different_nr_rows then
            dbms_output.put_line('nr de titluri diferit de nr de elemente
end;
/
select * from question_hpa where question_id = 'quest4-1';
select * from option_hpa where question_id = 'quest4-1';
```

Task completed in 0.057 seconds

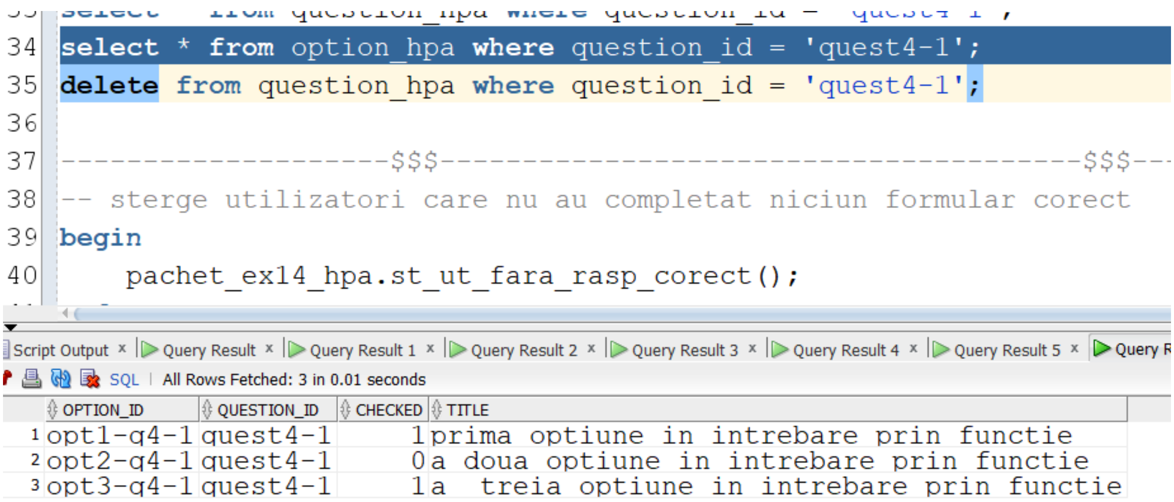
```
se creeaza intrebare noua cu id-ul quest4-1 pt formularul form1


PL/SQL procedure successfully completed.
```



```
select * from question_hpa where question_id = 'quest4-1';
select * from option_hpa where question_id = 'quest4-1';
delete from question_hpa where question_id = 'quest4-1';

--------------------$$$---------------------------------------$$$-----
```

All Rows Fetched: 1 in 0.006 seconds

	QUESTION_ID	TITLE	FORM_ID	Q_TYPE
1	quest4-1	intrebare 4 adaugata cu functie ajutatoare	form1	CHECKBOX

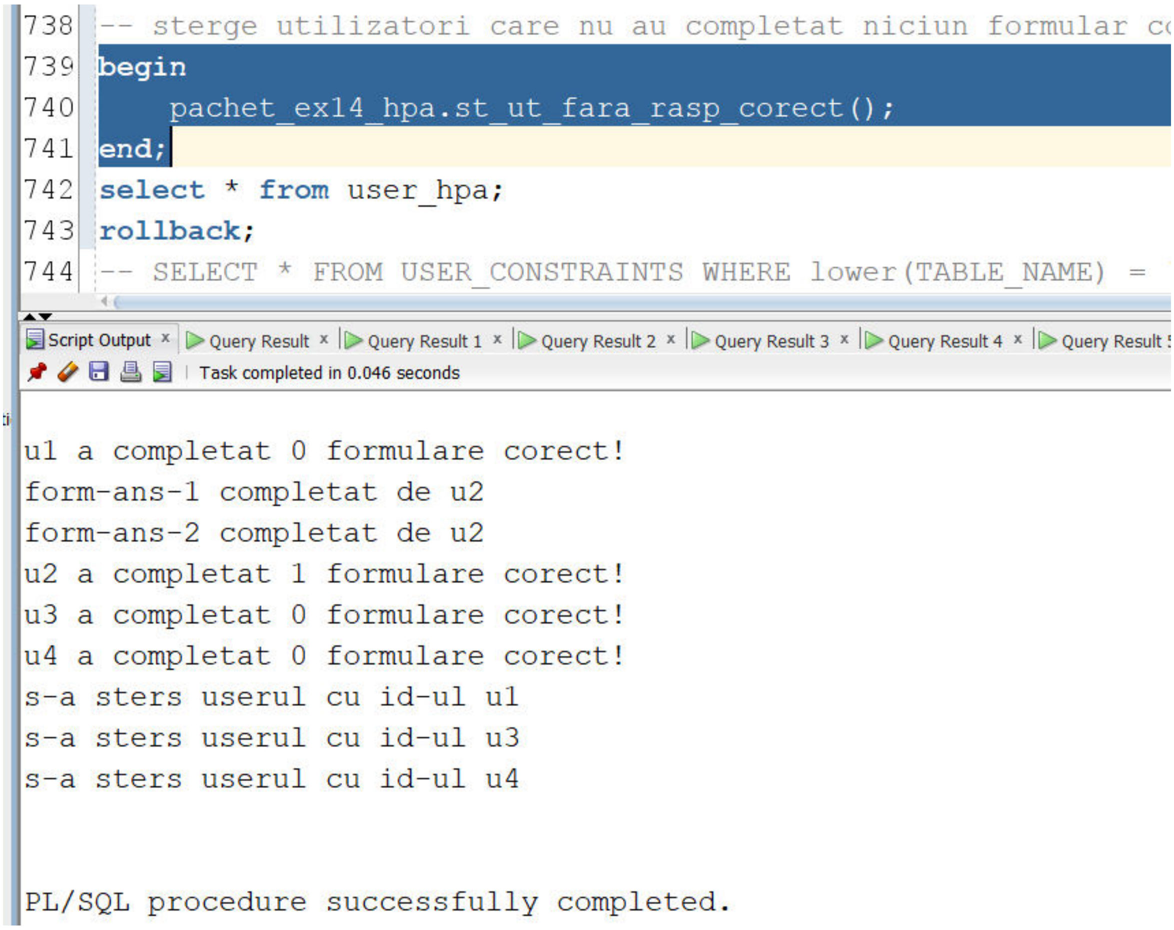


```
select * from option_hpa where question_id = 'quest4-1';
delete from question_hpa where question_id = 'quest4-1';

--------------------$$$---------------------------------------$$$--
-- sterge utilizatori care nu au completat niciun formular corect
begin
    pachet_ex14_hpa.st_ut_fara_rasp_corect();
```

All Rows Fetched: 3 in 0.01 seconds

	OPTION_ID	QUESTION_ID	CHECKED	TITLE
1	opt1-q4-1	quest4-1	1	prima optiune in intrebare prin functie
2	opt2-q4-1	quest4-1	0	a doua optiune in intrebare prin functie
3	opt3-q4-1	quest4-1	1	a treia optiune in intrebare prin functie

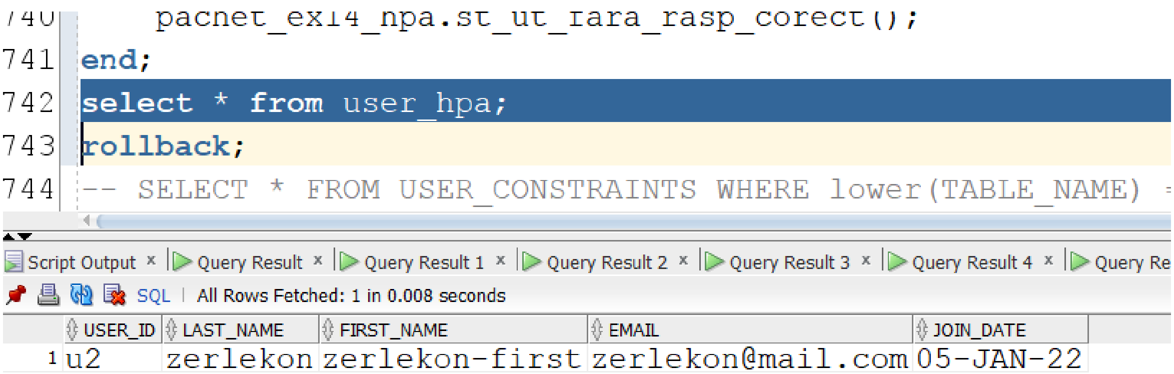
```
--------------------$$$---------------------------------------$$$--------------

-- sterge utilizatori care nu au completat niciun formular corect

begin

   pachet_ex14_hpa.st_ut_fara_rasp_corect();

end;

select * from user_hpa;
```



```sql
-- sterge utilizatori care nu au completat niciun formular co
begin
    pachet_ex14_hpa.st_ut_fara_rasp_corect();
end;
select * from user_hpa;
rollback;
-- SELECT * FROM USER_CONSTRAINTS WHERE lower(TABLE_NAME) =
```

Task completed in 0.046 seconds

```
u1 a completat 0 formulare corect!
form-ans-1 completat de u2
form-ans-2 completat de u2
u2 a completat 1 formulare corect!
u3 a completat 0 formulare corect!
u4 a completat 0 formulare corect!
s-a sters userul cu id-ul u1
s-a sters userul cu id-ul u3
s-a sters userul cu id-ul u4


PL/SQL procedure successfully completed.
```



```sql
    pachet_ex14_hpa.st_ut_fara_rasp_corect();
end;
select * from user_hpa;
rollback;
-- SELECT * FROM USER_CONSTRAINTS WHERE lower(TABLE_NAME) =
```

All Rows Fetched: 1 in 0.008 seconds

	USER_ID	LAST_NAME	FIRST_NAME	EMAIL	JOIN_DATE
1	u2	zerlekon	zerlekon-first	zerlekon@mail.com	05-JAN-22