

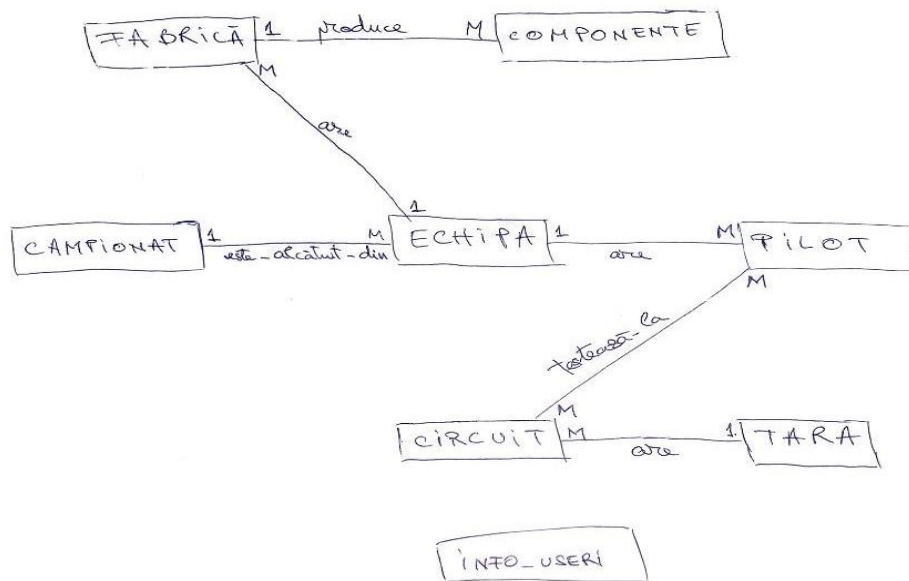
PROIECT SGBD MOTORSPORT

HERNEST MIHAI 242

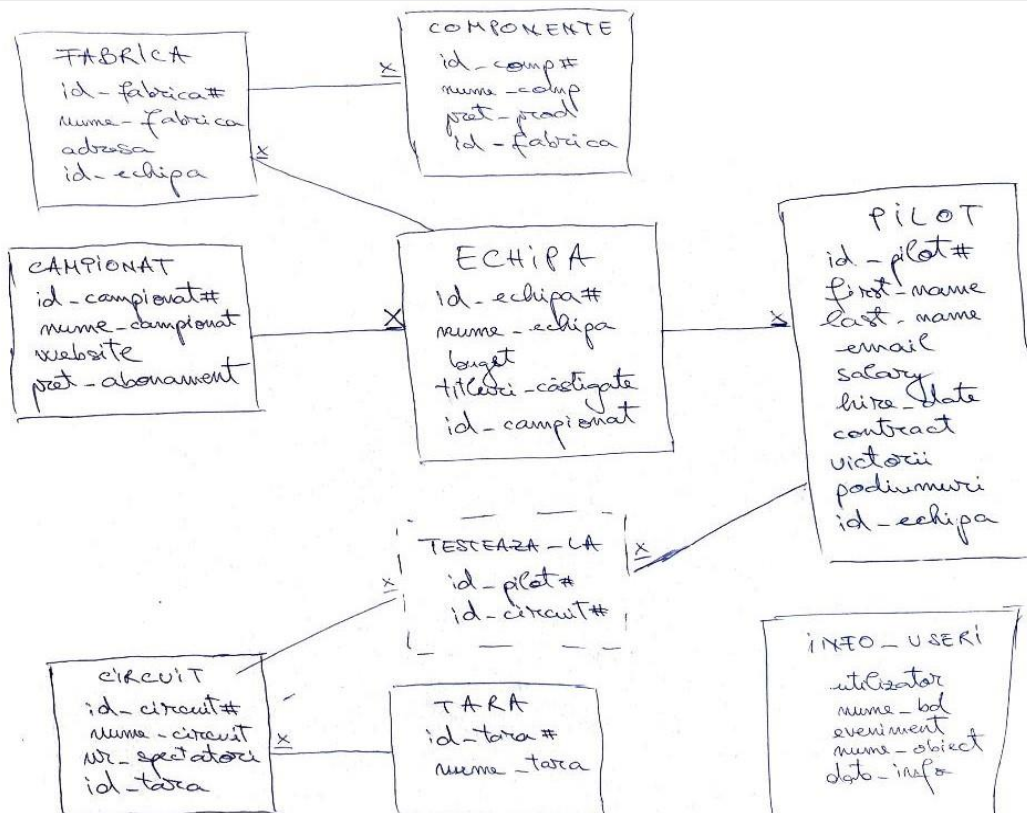
1.

Proiectul este realizat in jurul bazei de date FIA(Federatia Internationala de Automobilism) ce cuprinde campionate de motorsport precum Formula 1, Formula 2, Formula E etc. Aceasta baza de date s-a format pe baza momentului actual al desfasurarii acestor campionate si anume pre-sezonul, testarea masinilor si producerea noilor componente pentru sezonul ce urmeaza. Astfel, baza de date este alcatuita din urmatoarele entitati: campionat, ce cuprinde informatii legate de nume, website sau pretul unui abonament pentru fiecare campionat; echipa, cuprinde bugetul, cate titluri a castigat fiecare echipa; pilot contine informatii legate de palmaresul acestora, dar si informatii ce il leaga de echipa actuala(salariu, contract). Fiecare echipa are, de asemenea, fabrici in care poate produce componente pentru monoposturile din sezonul viitor. Pilotii au posibilitatea de a testa noile componente pe mai multe circuite, circuite ce fac parte din diferite tari ale lumii. Toate comenzile de tip LDD vor fi stocate in tabelul info_useri.

2.



3.



4.

```
CREATE TABLE campionat(  
    id_campionat number(3) PRIMARY KEY,  
    nume_campionat varchar2(50),  
    website varchar2(30),  
    pret_abonament number(4,3)  
);
```

```
CREATE TABLE echipa(  
    id echipa number(3) PRIMARY KEY,  
    nume_echipa varchar2(50),  
    buget number(10,3),  
    titluri_castigate number(3),  
    id_campionat number(3) REFERENCES campionat(id_campionat)  
on delete cascade  
);
```

```
CREATE TABLE pilot(  
    id_pilot number(4) PRIMARY KEY,  
    first_name varchar2(20),  
    last_name varchar2(20),  
    email varchar2(40),  
    salary number(10,3),  
    hire_date date,  
    contract number(2),
```

```
victorii number(4),  
podiumuri number(4),  
id echipa number(3) REFERENCES echipa(id echipa) on delete  
cascade  
);
```

```
CREATE TABLE tara(  
    id_tara number(3) primary key,  
    nume_tara varchar2(50)  
);
```

```
CREATE TABLE circuit(  
    id_circuit number(3) PRIMARY KEY,  
    nume_circuit varchar2(50),  
    nr_spectatori number(6),  
    id_tara number(3) references tara(id_tara) on delete cascade  
);
```

```
CREATE TABLE testeaza_la(  
    id_pilot number(4) NOT NULL,  
    id_circuit number(3) NOT NULL,  
    FOREIGN KEY (id_pilot) references pilot(id_pilot),  
    FOREIGN KEY (id_circuit) references circuit(id_circuit),  
    UNIQUE (id_pilot, id_circuit)  
);
```

```
CREATE TABLE fabrica(  
    id_fabrica number(3) PRIMARY KEY,  
    nume_fabrica varchar2(40),  
    adresa varchar2(40),  
    id echipa number(3) references echipa(id echipa) on delete  
cascade  
);
```

```
CREATE TABLE componente(  
    id_comp number(4) PRIMARY KEY,  
    nume_comp varchar2(30),  
    pret_prod number(10, 3),  
    id_fabrica number(3) references fabrica(id_fabrica) on delete  
cascade  
);
```

```
CREATE TABLE info_useri(  
    utilizator VARCHAR2(30),  
    nume_bd VARCHAR2(30),  
    eveniment VARCHAR2(20),  
    nume_obiect VARCHAR2(30),  
    data_info DATE  
);
```

Worksheet Query Builder

```
1 CREATE TABLE campionat(  
2     id_campionat number(3) PRIMARY KEY,  
3     nume_campionat varchar2(50),  
4     website varchar2(30),  
5     pret_abonament number(4,3)  
6 );  
7  
8 CREATE TABLE echipa(  
9     id echipa number(3) PRIMARY KEY,  
10    nume echipa varchar2(50),  
11    buget number(10,3),  
12    titluri_castigate number(3),  
13    id_campionat number(3) REFERENCES campionat(id_campionat) on delete cascade  
14 );  
15  
16 CREATE TABLE pilot(  
17     id_pilot number(4) PRIMARY KEY,  
18     first_name varchar2(20),  
19     last_name varchar2(20),  
20     email varchar2(40),  
21     salary number(10,3),  
22     hire_date date,  
23     contract number(2),  
24     victorii number(4),  
25     podiumuri number(4),  
26     id echipa number(3) REFERENCES echipa(id echipa) on delete cascade  
27 );
```

Script Output x

Task completed in 1.197 seconds

*Action:

Table CAMPIONAT created.

Table ECHIPA created.

```
16 CREATE TABLE pilot(  
17     id_pilot number(4) PRIMARY KEY,  
18     first_name varchar2(20),  
19     last_name varchar2(20),  
20     email varchar2(40),  
21     salary number(10,3),  
22     hire_date date,  
23     contract number(2),  
24     victorii number(4),  
25     podiumuri number(4),  
26     id echipa number(3) REFERENCES echipa(id echipa) on delete cascade  
27 );  
28  
29 CREATE TABLE tara(  
30     id_tara number(3) primary key,  
31     nume_tara varchar2(50)  
32 );  
33  
34 CREATE TABLE circuit(  
35     id_circuit number(3) PRIMARY KEY,  
36     nume_circuit varchar2(50),  
37     nr_spectatori number(6),  
38     id_tara number(3) references tara(id_tara) on delete cascade  
39 );  
40
```

Script Output x

Task completed in 1.197 seconds

Table PILOT created.

Table TARA created.

Table CIRCUIT created.

```
40
41 CREATE TABLE testeaza_la(
42     id_pilot number(4) NOT NULL,
43     id_circuit number(3) NOT NULL,
44     FOREIGN KEY (id_pilot) references pilot(id_pilot),
45     FOREIGN KEY (id_circuit) references circuit(id_circuit),
46     UNIQUE (id_pilot, id_circuit)
47 );
48
49 CREATE TABLE fabrica(
50     id_fabrica number(3) PRIMARY KEY,
51     nume_fabrica varchar2(40),
52     adresa varchar2(40),
53     id echipa number(3) references echipa(id echipa) on delete cascade
54 );
55
56 CREATE TABLE componente(
57     id_comp number(4) PRIMARY KEY,
58     nume_comp varchar2(30),
59     pret_prod number(10, 3),
60     id_fabrica number(3) references fabrica(id_fabrica) on delete cascade
61 );
62
63 CREATE TABLE info_useri(
64     utilizator VARCHAR2(30),
65     nume_bd VARCHAR2(30),
66     eveniment VARCHAR2(20),
67     nume_obiect VARCHAR2(30),
68     data_info DATE
69 );
```

Script Output x

Task completed in 1.197 seconds

Table TESTEAZA_LA created.

Table FABRICA created.

Table COMPONENTE created.

```
62
63 CREATE TABLE info_useri(
64     utilizator VARCHAR2(30),
65     nume_bd VARCHAR2(30),
66     eveniment VARCHAR2(20),
67     nume_obiect VARCHAR2(30),
68     data_info DATE
69 );
```

Script Output x

Task completed in 1.197 seconds

Table INFO_USERI created.

5.

----CAMPIONAT

insert into campionat values (1, 'Formula 1', 'www.formula1.com', 9.99);

insert into campionat values (2, 'Formula 2', 'www.formula2.com', 2.99);

insert into campionat values (3, 'Formula 3', 'www.formula3.com', 1.99);

insert into campionat values (4, 'Formula E', 'www.formulae.com', 4.99);

insert into campionat values (5, '24 Hours of Le Mans', 'www.lemans.com', 5.99);

insert into campionat values (6, 'Indycar', 'www.indycar.com', 6.95);

-----ECHIPA

insert into echipa values(30, 'Scuderia Ferrari', 300000, 16, 1);

insert into echipa values(31, 'Red Bull Racing', 150000, 4, 1);

insert into echipa values(32, 'Mercedes AMG Petronas', 190000, 7, 1);

insert into echipa values(40, 'Prema Racing', 50000, 1, 2);

insert into echipa values(60, 'Techeetah', 70000, 2, 4);

insert into echipa values(41, 'Uni-Virtuosi Racing', 40000, 0, 2);

insert into echipa values(33, 'Team Penske', 40000, 3, 6);

-----PILOT

insert into pilot values(5, 'Sebastian', 'Vettel', 'seb5@gmail.com',
100000, TO_DATE('03/05/2015', 'dd/mm/yyyy'),

5, 53, 121, 30);

insert into pilot values(16, 'Charles', 'Leclerc', 'leclerc1@sf.com',
20000, TO_DATE('11/09/2018', 'dd/mm/yyyy'),

3, 2, 12, 30);

insert into pilot values(44, 'Lewis', 'Hamilton',
'hamilton@yahoo.com', 50000, TO_DATE('07/10/2013',
'dd/mm/yyyy'),

1, 95, 165, 32);

insert into pilot values(33, 'Max', 'Verstappen', 'max@gmail.com',
25000, TO_DATE('04/09/2016', 'dd/mm/yyyy'),

3, 10, 42, 31);

insert into pilot values(13, 'Antonio', 'Da Costa', 'anto@yahoo.com',
10000, TO_DATE('11/08/2018', 'dd/mm/yyyy'),

2, 1, 4, 60);

insert into pilot values(47, 'Mick', 'Schumacher', 'mick47@f2.com',
15000, TO_DATE('22/06/2019', 'dd/mm/yyyy'),

2, 5, 10, 40);

-----TARA

insert into tara values(10, 'Italia');

```
insert into tara values(11, 'Ungaria');
insert into tara values(12, 'Belgia');
insert into tara values(13, 'Olanda');
insert into tara values(14, 'Turcia');
insert into tara values(15, 'America');
```

-----CIRCUIT

```
insert into circuit values(1, 'Autodromo Monza', 10000,10);
insert into circuit values(2, 'Hungaroring', 8000,11);
insert into circuit values(3, 'Spa-Francorchamps', 15000,12);
insert into circuit values(4, 'Park Zandvoort', 5000,13);
insert into circuit values(5, 'Mugello', 2000,10);
insert into circuit values(6, 'Circuit of the Americas', 5000, 15);
```

----TESTEAZA_LA

```
insert into testeaza_la values(5, 1);
insert into testeaza_la values(16, 1);
insert into testeaza_la values(44, 1);
insert into testeaza_la values(33, 1);
insert into testeaza_la values(5, 2);
insert into testeaza_la values(16, 2);
insert into testeaza_la values(44, 2);
```

insert into testeaza_la values(33, 2);
insert into testeaza_la values(13, 4);
insert into testeaza_la values(5, 3);
insert into testeaza_la values(16, 3);
insert into testeaza_la values(33, 3);

-----FABRICA

insert into fabrica values(10, 'Ferrari Factory', 'Via Abetone Inferiore', 30);
insert into fabrica values(11, 'Red Bull Racing Factory', 'Bradbourne Dr MK7 8BJ', 31);
insert into fabrica values(12, 'Mercedes AMG Petronas Factory', 'Operations Centre NN13 7BD', 32);
insert into fabrica values(13, 'Prema Racing Factory', 'Via Alcide de Gasperi', 40);
insert into fabrica values(14, 'Techeetah Factory', 'Jiajin Expy', 60);

-----COMPONENTE

insert into componente values(10, 'Motor Ferrari', 9999999, 10);
insert into componente values(20, 'Transmisie RB', 100000, 11);
insert into componente values(30, 'Sistem Suspensie MAMG', 400000, 12);
insert into componente values(11, 'Frane SF2000', 80000, 10);
insert into componente values(21, 'Motor RB', 8000000, 11);

```
83 insert into campionat values (1, 'Formula 1', 'www.formula1.com', 9.99);
84 insert into campionat values (2, 'Formula 2', 'www.formula2.com', 2.99);
85 insert into campionat values (3, 'Formula 3', 'www.formula3.com', 1.99);
86 insert into campionat values (4, 'Formula E', 'www.formulae.com', 4.99);
87 insert into campionat values (5, '24 Hours of Le Mans', 'www.lemans.com', 5.99);
88 insert into campionat values (6, 'Indycar', 'www.indycar.com', 6.95);
89
90 select * from campionat;
91
92
93 -----ECHIPA
94 insert into echipa values(30, 'Scuderia Ferrari', 300000, 16, 1);
95 insert into echipa values(31, 'Red Bull Racing', 150000, 4, 1);
96 insert into echipa values(32, 'Mercedes AMG Petronas', 190000, 7, 1);
```

Script Output x Query Result x

SQL | All Rows Fetched: 6 in 0.107 seconds

ID_CAMPIONAT	NUME_CAMPIONAT	WEBSITE	PRET_ABONAMENT
1	1 Formula 1	www.formula1.com	9.99
2	2 Formula 2	www.formula2.com	2.99
3	3 Formula 3	www.formula3.com	1.99
4	4 Formula E	www.formulae.com	4.99
5	5 24 Hours of Le Mans	www.lemans.com	5.99
6	6 Indycar	www.indycar.com	6.95

```
90
91 -----ECHIPA
92 insert into echipa values(30, 'Scuderia Ferrari', 300000, 16, 1);
93 insert into echipa values(31, 'Red Bull Racing', 150000, 4, 1);
94 insert into echipa values(32, 'Mercedes AMG Petronas', 190000, 7, 1);
95 insert into echipa values(40, 'Prema Racing', 50000, 1, 2);
96 insert into echipa values(60, 'Techeetah', 70000, 2, 4);
97 insert into echipa values(41, 'Uni-Virtuosi Racing', 40000, 0, 2);
98 insert into echipa values(33, 'Team Penske', 40000, 3, 6);
99 select * from echipa;
100
101
102 -----PILOT
103 insert into pilot values(5, 'Sebastian', 'Vettel', 'seb5@gmail.com', 100000,
```

Script Output x Query Result x

SQL | All Rows Fetched: 7 in 0.007 seconds

ID_ECHIPA	NUME_ECHIPA	BUGET	TITLURI_CASTIGATE	ID_CAMPIONAT
1	30 Scuderia Ferrari	300000	16	1
2	31 Red Bull Racing	150000	4	1
3	32 Mercedes AMG Petronas	190000	7	1
4	40 Prema Racing	50000	1	2
5	60 Techeetah	70000	2	4
6	41 Uni-Virtuosi Racing	40000	0	2
7	33 Team Penske	40000	3	6

```

103 -----PILOT
104 insert into pilot values(5, 'Sebastian', 'Vettel', 'seb5@gmail.com', 100000, TO_DATE('03/05/2015', 'dd/mm/yyyy'),
105 5, 53, 121, 30);
106 insert into pilot values(16, 'Charles', 'Leclerc', 'leclerc1@sf.com', 20000, TO_DATE('11/09/2018', 'dd/mm/yyyy'),
107 3, 2, 12, 30);
108 insert into pilot values(44, 'Lewis', 'Hamilton', 'hamilton@yahoo.com', 50000, TO_DATE('07/10/2013', 'dd/mm/yyyy'),
109 1, 95, 165, 32);
110 insert into pilot values(33, 'Max', 'Verstappen', 'max@gmail.com', 25000, TO_DATE('04/09/2016', 'dd/mm/yyyy'),
111 3, 10, 42, 31);
112 insert into pilot values(13, 'Antonio', 'Da Costa', 'anto@yahoo.com', 10000, TO_DATE('11/08/2018', 'dd/mm/yyyy'),
113 2, 1, 4, 60);
114 insert into pilot values(47, 'Mick', 'Schumacher', 'mick47@f2.com', 15000, TO_DATE('22/06/2019', 'dd/mm/yyyy'),
115 2, 5, 10, 40);
116 select * from pilot;
117
118 -----TARA
119 insert into tara values(10, 'Italia');
120 insert into tara values(11, 'Ungaria');

```

Script Output x Query Result x

SQL | All Rows Fetched: 6 in 0.005 seconds

ID_PILOT	FIRST_NAME	LAST_NAME	EMAIL	SALARY	HIRE_DATE	CONTRACT	VICTORIES	PODIUMS	WINS	POINTS
5	Sebastian	Vettel	seb5@gmail.com	100000	03-MAY-15	5	53	121	30	
16	Charles	Leclerc	leclerc1@sf.com	20000	11-SEP-18	3	2	12	30	
44	Lewis	Hamilton	hamilton@yahoo.com	50000	07-OCT-13	1	95	165	32	
33	Max	Verstappen	max@gmail.com	25000	04-SEP-16	3	10	42	31	
13	Antonio	Da Costa	anto@yahoo.com	10000	11-AUG-18	2	1	4	60	
47	Mick	Schumacher	mick47@f2.com	15000	22-JUN-19	2	5	10	40	

```

119 -----TARA
120 insert into tara values(10, 'Italia');
121 insert into tara values(11, 'Ungaria');
122 insert into tara values(12, 'Belgia');
123 insert into tara values(13, 'Olanda');
124 insert into tara values(14, 'Turcia');
125 insert into tara values(15, 'America');
126 select * from tara;
127
128 -----CIRCUIT
129 insert into circuit values(1, 'Autodromo Monza', 10000, 10);

```

Script Output x Query Result x

SQL | All Rows Fetched: 6 in 0.01 seconds

ID_TARA	NUME_TARA
10	Italia
11	Ungaria
12	Belgia
13	Olanda
14	Turcia
15	America

```

128
129 -----CIRCUIT
130 insert into circuit values(1, 'Autodromo Monza', 10000,10);
131 insert into circuit values(2, 'Hungaroring', 8000,11);
132 insert into circuit values(3, 'Spa-Francorchamps', 15000,12);
133 insert into circuit values(4, 'Park Zandvoort', 5000,13);
134 insert into circuit values(5, 'Mugello', 2000,10);
135 insert into circuit values(6, 'Circuit of the Americas', 5000, 15);
136 select * from circuit;
137 -----
138
139 ----TESTEAZA_LA
140 insert into testeaza_la values(5, 1);
141 insert into testeaza_la values(16, 1);

```

Script Output x Query Result x Query Result 1 x

SQL All Rows Fetched: 6 in 0.033 seconds

ID_CIRCUIT	NUME_CIRCUIT	NR_SPECTATORI	ID_TARA
1	1 Autodromo Monza	10000	10
2	2 Hungaroring	8000	11
3	3 Spa-Francorchamps	15000	12
4	4 Park Zandvoort	5000	13
5	5 Mugello	2000	10
6	6 Circuit of the Americas	5000	15

```

139 ----TESTEAZA_LA
140 insert into testeaza_la values(5, 1);
141 insert into testeaza_la values(16, 1);
142 insert into testeaza_la values(44, 1);
143 insert into testeaza_la values(33, 1);
144 insert into testeaza_la values(5, 2);
145 insert into testeaza_la values(16, 2);
146 insert into testeaza_la values(44, 2);
147 insert into testeaza_la values(33, 2);
148 insert into testeaza_la values(13, 4);
149 insert into testeaza_la values(5, 3);
150 insert into testeaza_la values(16, 3);
151 insert into testeaza_la values(33, 3);
152 select * from testeaza_la;
153 -----
154
155 -----FABRICA
156 insert into fabrica values(10, 'Ferrari Factory', 'Via Abetone Inferio

```

Script Output x Query Result x Query Result 1 x Query Result 2 x

SQL All Rows Fetched: 12 in 0.006 seconds

ID_PILOT	ID_CIRCUIT
1	5
2	16
3	44
4	33
5	5
6	16
7	44
8	33
9	13
10	5
11	16
12	33

```

155 -----FABRICA
156 insert into fabrica values(10, 'Ferrari Factory', 'Via Abetone Inferiore', 30);
157 insert into fabrica values(11, 'Red Bull Racing Factory', 'Bradbourne Dr MK7 8BJ', 31);
158 insert into fabrica values(12, 'Mercedes AMG Petronas Factory', 'Operations Centre NN13 7BD', 32);
159 insert into fabrica values(13, 'Prema Racing Factory', 'Via Alcide de Gasperi', 40);
160 insert into fabrica values(14, 'Techeetah Factory', 'Jiajin Expy', 60);
161 select * from fabrica;
162 -----
163
164 -----COMPONENTE
165 insert into componente values(10, 'Motor Ferrari', 9999999, 10);
166 insert into componente values(20, 'Transmisie RB', 100000, 11);
167 insert into componente values(30, 'Sistem Suspensie MAMG', 400000, 12);
168 insert into componente values(11, 'Frane SF2000', 80000, 10);
169 insert into componente values(21, 'Motor RB', 8000000, 11);
170 -----
171

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x

All Rows Fetched: 5 in 0.043 seconds

ID_FABRICA	NUME_FABRICA	ADRESA	ID_ECHIPA
1	10 Ferrari Factory	Via Abetone Inferiore	30
2	11 Red Bull Racing Factory	Bradbourne Dr MK7 8BJ	31
3	12 Mercedes AMG Petronas Factory	Operations Centre NN13 7BD	32
4	13 Prema Racing Factory	Via Alcide de Gasperi	40
5	14 Techeetah Factory	Jiajin Expy	60

```

163 -----COMPONENTE
164
165 insert into componente values(10, 'Motor Ferrari', 9999999, 10);
166 insert into componente values(20, 'Transmisie RB', 100000, 11);
167 insert into componente values(30, 'Sistem Suspensie MAMG', 400000, 12);
168 insert into componente values(11, 'Frane SF2000', 80000, 10);
169 insert into componente values(21, 'Motor RB', 8000000, 11);
170 select * from componente;
171 -----
172

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x

All Rows Fetched: 5 in 0.102 seconds

ID_COMP	NUME_COMP	PRET_PROD	ID_FABRICA
1	10 Motor Ferrari	9999999	10
2	20 Transmisie RB	100000	11
3	30 Sistem Suspensie MAMG	400000	12
4	11 Frane SF2000	80000	10
5	21 Motor RB	8000000	11

6.

Pentru campionatul cu cele mai putine echipe(numar diferit de zero), inserati o noua echipa cu bugetul 10000 si afisati echipa cu cel mai mic buget din campionatul cu cele mai multe echipe. Afisati modificarile.

CREATE SEQUENCE id_echipe START WITH 105;

create or replace procedure ex6

is

**type camp is table of campionat.id_campionat%TYPE INDEX BY
PLS_INTEGER;**

lista camp;

min_buget echipa.buget%type;

min_cnt number:=10;

max_cnt number:=0;

cnt number;

id_min_camp campionat.id_campionat%type;

id_max_camp campionat.id_campionat%type;

id_e echipa.id_echipa%type;

begin

select id_campionat

bulk collect into lista

from campionat;


```
for i in 1..lista.count loop
    select count(*) into cnt
    from echipa
    where id_campionat = lista(i);

    if cnt < min_cnt and cnt != 0 then
        min_cnt := cnt;
        id_min_camp := lista(i);
    end if;

    if cnt > max_cnt then
        max_cnt := cnt;
        id_max_camp := lista(i);
        select min(buget) into min_buget
        from echipa
        where id_campionat = lista(i);
    end if;
end loop;
```

```
insert into echipa values(id_echipe.nextval, 'EX6 RACING',
10000, 0, id_min_camp);
```

```
dbms_output.put_line('A fost inserata echipa in campionatul
cu id ' || id_min_camp);
```

```
select id_echipa into id_e
```

```

from echipa

where id_campionat = id_max_camp and buget = min_buget;

dbms_output.put_line('Echipa cu cel mai mic buget are id-ul '
|| id_e);

end ex6;

begin
    ex6;
end;

/

```

```

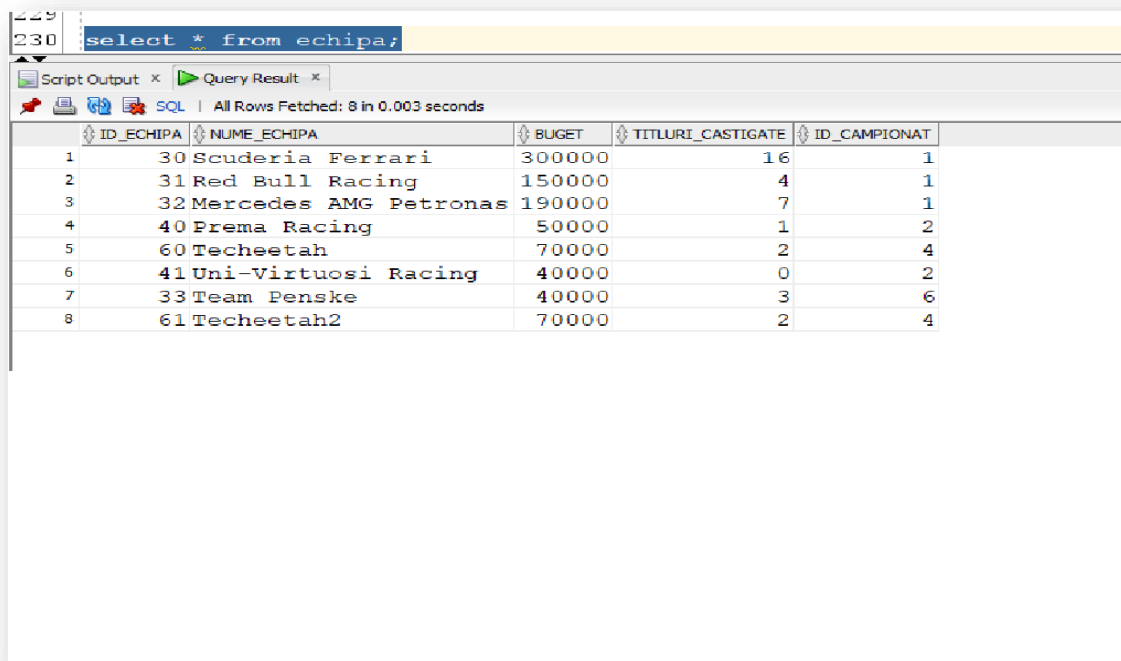
177 create or replace procedure ex6
178 is
179     type camp is table of campionat.id_campionat%TYPE INDEX BY PLS_INTEGER;
180     lista camp;
181     min_buget echipa.buget%type;
182     min_cnt number:=10;
183     max_cnt number:=0;
184     cnt number;
185     id_min_camp campionat.id_campionat%type;
186     id_max_camp campionat.id_campionat%type;
187     id_e echipa.id echipa%type;
188
189     begin
190         select id_campionat
191         bulk collect into lista
192         from campionat;
193
194         for i in 1..lista.count loop
195             select count(*) into cnt
196             from echipa
197             where id_campionat = lista(i);
198
199             if cnt < min_cnt and cnt != 0 then
200                 min_cnt := cnt;
201                 id_min_camp := lista(i);
202             end if;
203
204             if cnt > max_cnt then
205                 max_cnt := cnt;
206                 id_max_camp := lista(i);

```

Script Output x
Task completed in 0.049 seconds

Procedure EX6 compiled

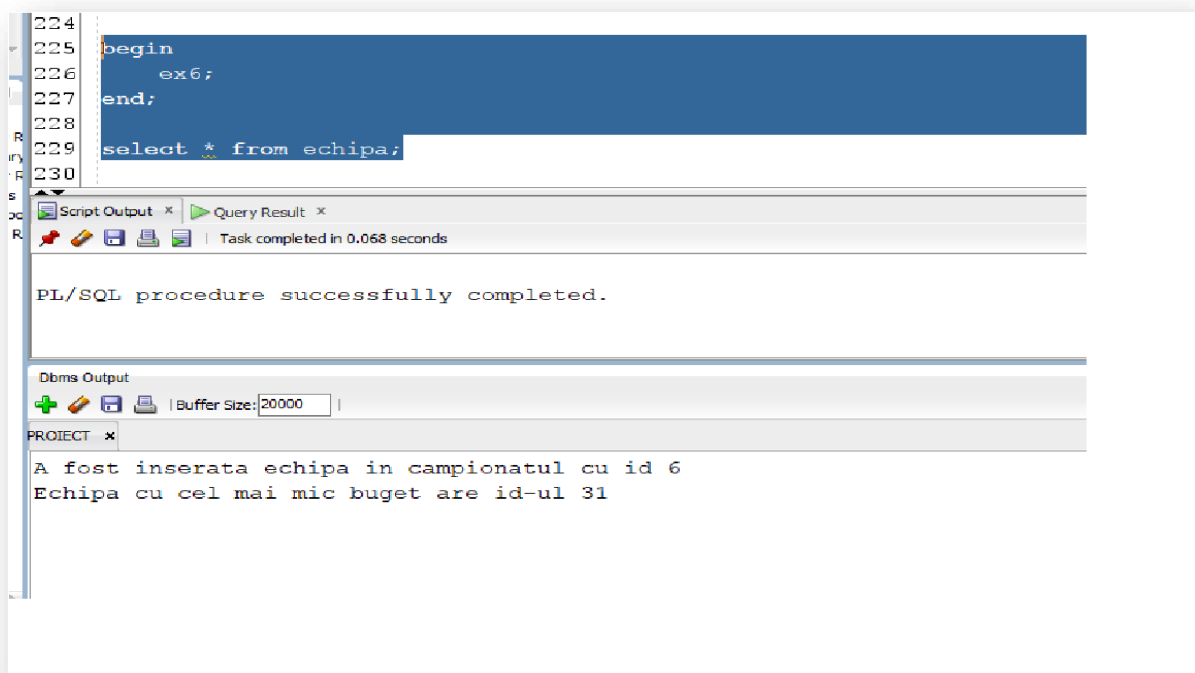
Tabelul ECHIPA inainte de apelarea procedurii:



The screenshot shows a SQL query result window with the query `select * from echipa;` and its results. The results are displayed in a table with 5 columns: ID_ECHIPA, NUME_ECHIPA, BUGET, TITLURI_CASTIGATE, and ID_CAMPIONAT. The table contains 8 rows of data.

ID_ECHIPA	NUME_ECHIPA	BUGET	TITLURI_CASTIGATE	ID_CAMPIONAT
1	30 Scuderia Ferrari	300000	16	1
2	31 Red Bull Racing	150000	4	1
3	32 Mercedes AMG Petronas	190000	7	1
4	40 Prema Racing	50000	1	2
5	60 Techeetah	70000	2	4
6	41 Uni-Virtuosi Racing	40000	0	2
7	33 Team Penske	40000	3	6
8	61 Techeetah2	70000	2	4

Campionatul cu cele mai putine echipe: id_campionat - 6 (are o singura echipa), iar campionatul cu cele mai multe echipe este: id_campionat – 1(are trei echipe). In urma apelarii ex6 va fi adaugata echipa din enunt in campionatul cu id 6, iar echipa cu cel mai mic buget din campionatul 1 este Red Bull Racing, astfel va fi afisat id-ul 31.



7.

Sa se afiseze pentru fiecare fabrica, pretul total al componentelor.

create or replace procedure ex7

is

cursor c is

select id_fabrica, nume_fabrica

from fabrica;

```

cursor d(param number) is
    select pret_prod
    from componente
    where id_fabrica = param;
suma number;
begin
    for i in c loop
        DBMS_OUTPUT.PUT_LINE('-----');
        DBMS_OUTPUT.PUT_LINE ('FABRICA: ' || i.ume_fabrica);
        suma := 0;
        for k in d(i.id_fabrica) loop
            suma := suma+k.pret_prod;
        end loop;
        DBMS_OUTPUT.PUT_LINE ('PRET TOTAL: ' || suma);
        DBMS_OUTPUT.PUT_LINE('-----');
    end loop;

end ex7;

begin
    ex7;
end;
/

```

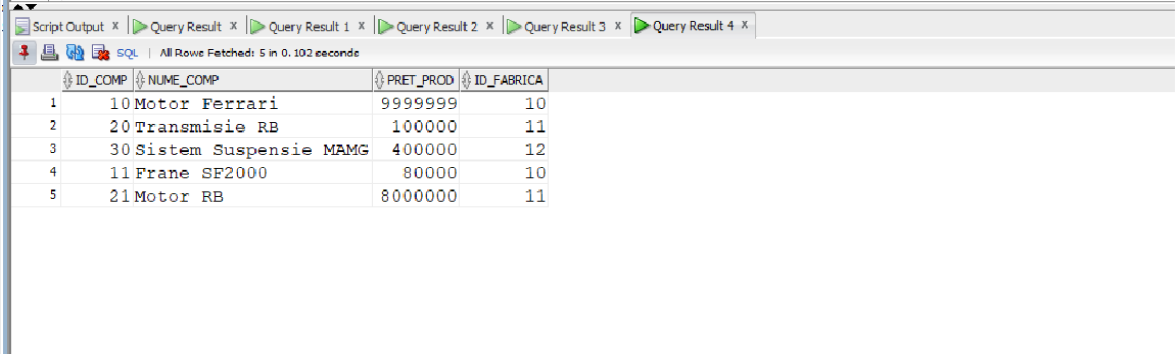
```
174 create or replace procedure ex7
175 is
176     cursor c is
177         select id_fabrica, nume_fabrica
178         from fabrica;
179
180     cursor d(param number) is
181         select pret_prod
182         from componente
183         where id_fabrica = param;
184     suma number;
185 begin
186     for i in c loop
187         DBMS_OUTPUT.PUT_LINE('-----');
188         DBMS_OUTPUT.PUT_LINE ('FABRICA: '|| i.nume_fabrica);
189         suma := 0;
190         for k in d(i.id_fabrica) loop
191             suma := suma+k.pret_prod;
192         end loop;
193         DBMS_OUTPUT.PUT_LINE ('PRET TOTAL: '|| suma);
194         DBMS_OUTPUT.PUT_LINE('-----');
195     end loop;
196
197 end ex7;
198
```

Script Output x | Query Result x

Task completed in 0.067 seconds

Procedure EX7 compiled

```
163
164 -----COMPONENTE
165 insert into componente values(10, 'Motor Ferrari', 9999999, 10);
166 insert into componente values(20, 'Transmisie RB', 100000, 11);
167 insert into componente values(30, 'Sistem Suspensie MAMG', 400000, 12);
168 insert into componente values(11, 'Frane SF2000', 80000, 10);
169 insert into componente values(21, 'Motor RB', 8000000, 11);
170 select * from componente;
171
172
```



ID_COMP	NUME_COMP	PRET_PROD	ID_FABRICA
1	10 Motor Ferrari	9999999	10
2	20 Transmisie RB	100000	11
3	30 Sistem Suspensie MAMG	400000	12
4	11 Frane SF2000	80000	10
5	21 Motor RB	8000000	11

Suma componentelor din fabrica cu id 10(Ferrari Factory) este: 10079999.

Suma componentelor din fabrica cu id 11(Red Bull Racing Factory) este: 8100000.

Suma componentelor din fabrica cu id 12(Mercedes AMG Petronas Factory) este: 400000.

Suma componentelor din fabrica cu id 13(Prema Racing Factory) este: 0.

Suma componentelor din fabrica cu id 14(Techeetah Factory) este: 0.

```

198
199 begin
200     ex7;
201 end;
202
203

```

Dbms Output
 Buffer Size: 20000

PROIECT x

```

-----
FABRICA: Ferrari Factory
PRET TOTAL: 10079999
-----
FABRICA: Red Bull Racing Factory
PRET TOTAL: 8100000
-----
FABRICA: Mercedes AMG Petronas Factory
PRET TOTAL: 4000000
-----
FABRICA: Prema Racing Factory
PRET TOTAL: 0
-----
FABRICA: Techeetah Factory
PRET TOTAL: 0
-----

```

8.

Pentru un pilot dat ca parametru sa se returneze numarul total al victoriilor pilotilor din campionatul in care concureaza acesta.

**create or replace function ex8(v_last_name
pilot.last_name%type) return number**

is

**type echipe is table of echipa.id_echipa%TYPE INDEX BY
PLS_INTEGER;**

lista echipe;

id_e pilot.id_echipa%type;

nume campionat.nume_campionat%type;

total number := 0;

id_camp campionat.id_campionat%type;


```
        ok number;  
begin  
    select id echipa into id_e  
    from pilot  
    where last_name = v_last_name;  
  
    select c.id_campionat into id_camp  
    from echipa e join campionat c on e.id_campionat =  
c.id_campionat  
    where e.id echipa = id_e;  
  
    select id echipa  
    bulk collect into lista  
    from echipa  
    where id_campionat = id_camp;  
  
    for i in (select id echipa, victorii  
              from pilot) loop  
        ok := 0;  
        for j in 1..lista.count loop  
            if i.id echipa = lista(j) then  
                ok := 1;  
            end if;  
        exit when ok = 1;  
    end loop;
```

```
end loop;  
if ok = 1 then  
    total := total + i.victorii;  
end if;  
end loop;
```

```
return total;
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN
```

```
        DBMS_OUTPUT.PUT_LINE ('Nu exista pilot cu acest  
nume');
```

```
        return -1;
```

```
    WHEN TOO_MANY_ROWS THEN
```

```
        DBMS_OUTPUT.PUT_LINE ('Exista mai multi piloti cu  
acest nume');
```

```
        return -2;
```

```
end;
```

```
begin
```

```
    DBMS_OUTPUT.PUT_LINE(ex8('Vettel'));
```

```
    DBMS_OUTPUT.PUT_LINE(ex8('Hernest'));
```

```
    DBMS_OUTPUT.PUT_LINE(ex8('Schumacher'));
```

end;

/

```
174
175 create or replace function ex8(v_last_name pilot.last_name%type) return number
176 is
177     type echipe is table of echipa.id_echipa%TYPE INDEX BY PLS_INTEGER;
178     lista echipe;
179     id_e pilot.id_echipa%type;
180     nume campionat.nume_campionat%type;
181     total number := 0;
182     id_camp campionat.id_campionat%type;
183     ok number;
184 begin
185     select id_echipa into id_e
186     from pilot
187     where last_name = v_last_name;
188
189     select c.id_campionat into id_camp
190     from echipa e join campionat c on e.id_campionat = c.id_campionat
191     where e.id_echipa = id_e;
192
193     select id_echipa
194     bulk collect into lista
195     from echipa
196     where id_campionat = id_camp;
197
198     for i in (select id_echipa, victorii
199              from pilot) loop
200         ok := 0;
```

Script Output x

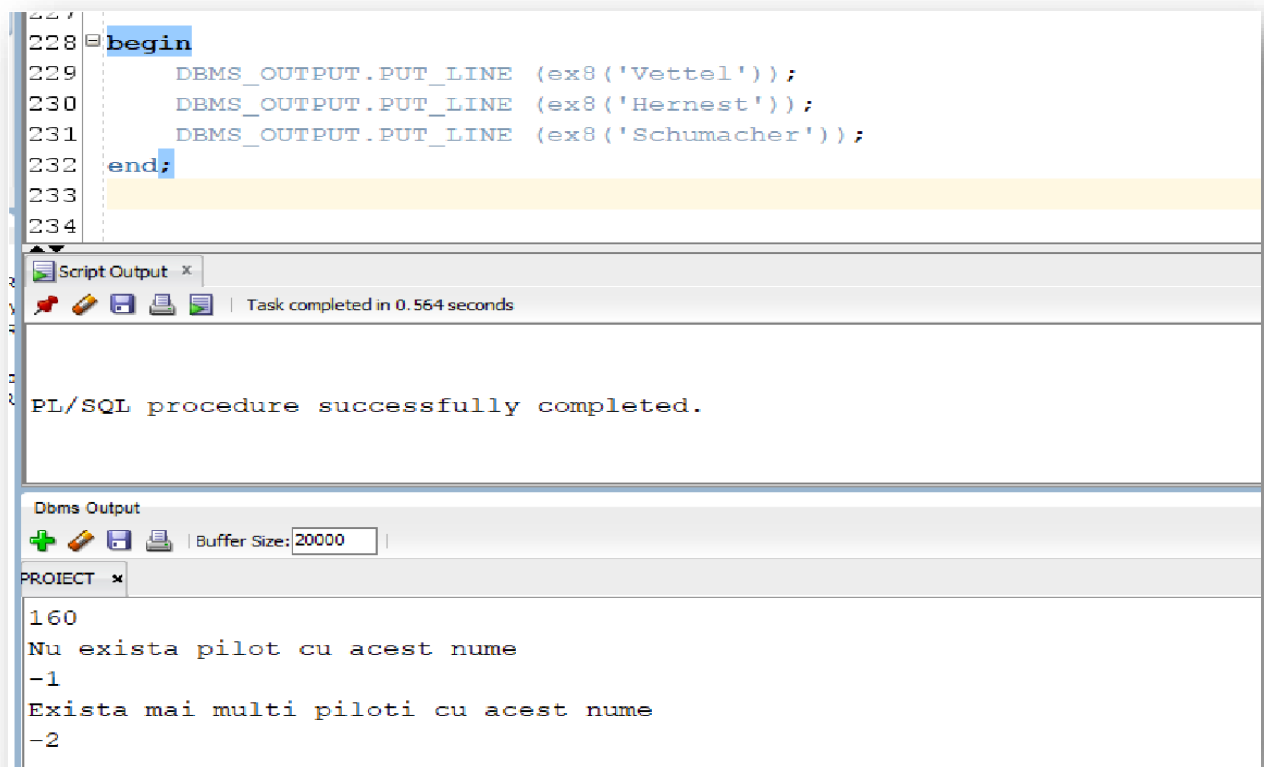
Task completed in 0.442 seconds

Function EX8 compiled

Daca apelam ex8("Vettel"), vom avea toti pilotii care concureaza in campionatul cu id 1: Vettel – 53 de victorii, Leclerc - 2 victorii, Hamilton – 95 de victorii, Verstappen – 10 victorii => In total sunt 160 de victorii.

Prima exceptie: nu exista pilotul cu numele dat ca parametru.

A doua exceptie: exista mai multi piloti cu acelasi nume.(ulterior a fost adaugat un alt pilot cu acelasi nume: insert into pilot values(48, 'Michael', 'Schumacher', 'legend@f1.com', 555555, TO_DATE('22/06/1996', 'dd/mm/yyyy'), 99, 91, 155, 30);)



The screenshot displays the Oracle SQL Developer environment. The top pane shows a PL/SQL script with the following code:

```
228 begin
229     DBMS_OUTPUT.PUT_LINE (ex8('Vettel'));
230     DBMS_OUTPUT.PUT_LINE (ex8('Hernest'));
231     DBMS_OUTPUT.PUT_LINE (ex8('Schumacher'));
232 end;
```

The bottom pane shows the 'Script Output' window with the message: 'PL/SQL procedure successfully completed.' Below this, the 'Dbms Output' window shows the results of the script execution:

```
160
Nu exista pilot cu acest nume
-1
Exista mai multi piloti cu acest nume
-2
```

9.

Pentru circuitul cu cel mai mic numar de spectatori din tara data ca parametru, sa se transfere pilotul cu cel mai mic salariu care a testat la acel circuit, la echipa cu cele mai multe titluri castigate cu un salariu dublu(bugetul noii echipe va fi scazut corespunzator), iar pentru echipa care a pierdut pilotul sa se construiasca o noua fabrica cu jumatate din banii repartizati fostului pilot(bugetul se va schimba corespunzator).

CREATE SEQUENCE new_factory START WITH 50;

**create or replace procedure ex9(v_num_tara
tara.num_tara%type)**

is

id_tr tara.id_tara%type;

**type circuite is table of circuit.id_circuit%type index by
pls_integer;**

id_c circuite;

**type piloti is table of pilot.id_pilot%TYPE INDEX BY
PLS_INTEGER;**

lista piloti;

nr_spec number;

max_spec number := 0;

id_c_max circuit.id_circuit%type;

min1 number:= 999999;

minteam echipa.id_echipa%type;

id_pilot_min pilot.id_pilot%type;

id_echipa_min pilot.id_echipa%type;

id_echipa_max pilot.id_echipa%type;

titles echipa.titluri_castigate%type;

sal pilot.salary%type;

team echipa.id_echipa%type;

exceptie EXCEPTION;

begin

select id_tara into id_tr

```
from tara
where nume_tara = v_nume_tara;

select id_circuit bulk collect into id_c
from circuit
where id_tara = id_tr;
```

```
for i in 1..id_c.count loop
    select nr_spectatori into nr_spec
    from circuit
    where id_circuit = id_c(i);

    if nr_spec > max_spec then
        max_spec := nr_spec;
        id_c_max := id_c(i);
    end if;
end loop;
```

```
select id_pilot
bulk collect into lista
from testeaza_la
where id_circuit = id_c_max;
```

```
for i in 1..lista.count loop
```

```
select salary, id echipa into sal, team
from pilot
where id_pilot = lista(i);
```

```
if sal < min1 then
    min1 := sal;
    minteam := team;
    id_pilot_min := lista(i);
end if;
end loop;
```

```
select id echipa, titluri_castigate into id echipa_max, titles
from echipa
where titluri_castigate in (select max(titluri_castigate)
                             from echipa);
if minteam = id echipa_max then
    raise exceptie;
end if;
```

```
update pilot set id echipa = id echipa_max,
                salary = salary * 2,
                hire_date = SYSDATE,
                contract = 1
where id_pilot = id_pilot_min;
```

```

select salary into sal
from pilot
where id_pilot = id_pilot_min;

update echipa set buget = buget - sal where id echipa =
id echipa_max;

insert into fabrica values(new_factory.nextval, 'New Factory',
'New Street', minteam);

update echipa set buget = buget + sal/4 where id echipa =
minteam;

EXCEPTION

WHEN NO_DATA_FOUND THEN

    DBMS_OUTPUT.PUT_LINE('Nu exista tara cu acest nume');

WHEN exceptie THEN

    DBMS_OUTPUT.PUT_LINE('Echipa la care trebuie transferat
pilotul corespunde cu echipa actuala');

end ex9;

begin

    ex9('Italia');

    ex9('Test');

    ex9('Italia');

```


end;

/

```
176
177 create or replace procedure ex9(v_nume_tara tara.nume_tara%type)
178 is
179     id_tr tara.id_tara%type;
180     type circuite is table of circuit.id_circuit%type index by pls_integer;
181     id_c circuite;
182     type piloti is table of pilot.id_pilot%TYPE INDEX BY PLS_INTEGER;
183     lista piloti;
184     nr_spec number;
185     max_spec number := 0;
186     id_c_max circuit.id_circuit%type;
187     min1 number:= 999999;
188     minteam echipa.id echipa%type;
189     id_pilot_min pilot.id_pilot%type;
190     id echipa_min pilot.id echipa%type;
191     id echipa_max pilot.id echipa%type;
192     titles echipa.titluri_castigate%type;
193     sal pilot.salary%type;
194     team echipa.id echipa%type;
195     exceptie EXCEPTION;
196 begin
197
198     select id_tara into id_tr
199     from tara
200     where nume_tara = v_nume_tara;
201
202     select id_circuit bulk collect into id_c
203     from circuit
204     where id_tara = id_tr;
205
206     for i in 1..id_c.count loop
```

Script Output x
Task completed in 0.086 seconds

Procedure EX9 compiled

```
264
265 begin
266     ex9('Italia');
267     ex9('Test');
268     ex9('Italia');
269 end;
270
271 select * from testeaza_la;
272 select * from pilot;
273 rollback;
274 update pilot set salary = 30000 where id_pilot = 16;
275
```

Script Output x Query Result x

Task completed in 0.071 seconds

Rollback complete.

PL/SQL procedure successfully completed.

Dbms Output

Buffer Size: 20000

PROJECT x

Nu exista tara cu acest nume
Echipa la care trebuie transferat pilotul corespunde cu echipa actuala

In exemplul de mai sus, pilotul cu cel mai mic salariu care a testat in Italia este Verstappen(id 33), cu un salariu de 25000(a fost actualizata valoarea salariului pilotului cu id 16, deoarece se afla in situatia cu cel mai mic salariu si nu s-ar fi produs schimbarea).

Pentru parametru Test se afiseaza mesajul “Nu exista tara”.

Apoi, la reapelarea metodei cu parametrul Italia, pilotul cu id 16 are salariul 30000, devenind pilotul cu cel mai mic salariu dupa actualizarea de mai sus. Astfel, fiind din echipa la care trebuie transferat, se afiseaza mesajul corespunzator.

```

271
272 select * from echipa;
273

```

ID_ECHIPA	NUME_ECHIPA	BUGET	TITLURI_CASTIGATE	ID_CAMPIONAT
30	Scuderia Ferrari	250000	16	1
31	Red Bull Racing	162500	4	1
32	Mercedes AMG Petronas	190000	7	1
40	Prema Racing	50000	1	2
60	Techeetah	70000	2	4
41	Uni-Virtuosi Racing	40000	0	2
33	Team Penske	40000	3	6
61	Techeetah2	70000	2	4
100	EX6 RACING	10000	0	6

In poza de sus se pot observa schimbarile in bugetul echipelor(Ferrari are un buget de 250000 de la 300000, iar Redbull de 162500 de la 150000).

271
272 `select * from fabrica;`
273

Script Output x Query Result x

SQL All Rows Fetched: 6 in 0.003 seconds

ID_FABRICA	NUME_FABRICA	ADRESA	ID_ECHIPA
1	10 Ferrari Factory	Via Abetone Inferiore	30
2	11 Red Bull Racing Factory	Bradbourne Dr MK7 8DJ	31
3	12 Mercedes AMG Petronas Factory	Operations Centre NN13 7BD	32
4	13 Prema Racing Factory	Via Alcide de Gasperi	40
5	14 Techeetah Factory	Jiajin Expy	60
6	58 New Factory	New Street	31

Display Output

Mai sus se poate observa noua fabrica adaugata in urma transferului, in dreptul echipei cu id 31, care a pierdut pilotul.

10.

Triggerul nu permite modificarea tabelului pilot in timpul sezonului regulat(ianuarie-octombrie).

`create or replace trigger ex10`

`before insert or update or delete on pilot`

`begin`

`if to_char(sysdate,'mm') >= 1 and to_char(sysdate,'mm') < 10`
`then`

`if inserting then`

`raise_application_error(-20000, 'Nu se pot insera piloti noi in`
`timpul sezonului');`

end if;

if updating then

**raise_application_error(-20001, 'Nu se pot actualiza
informatii despre piloti in timpul sezonului');**

end if;

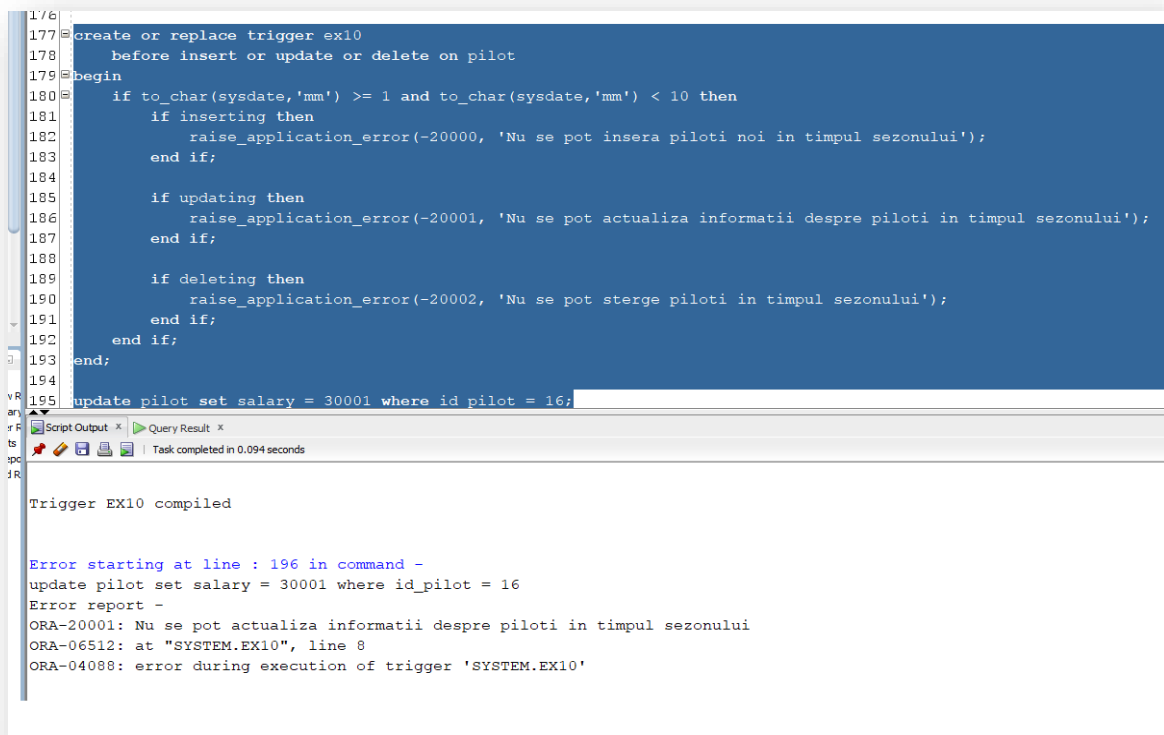
if deleting then

**raise_application_error(-20002, 'Nu se pot sterge piloti in
timpul sezonului');**

end if;

end if;

end;



```
176
177 create or replace trigger ex10
178 before insert or update or delete on pilot
179 begin
180 if to_char(sysdate,'mm') >= 1 and to_char(sysdate,'mm') < 10 then
181 if inserting then
182 raise_application_error(-20000, 'Nu se pot insera piloti noi in timpul sezonului');
183 end if;
184
185 if updating then
186 raise_application_error(-20001, 'Nu se pot actualiza informatii despre piloti in timpul sezonului');
187 end if;
188
189 if deleting then
190 raise_application_error(-20002, 'Nu se pot sterge piloti in timpul sezonului');
191 end if;
192 end if;
193 end;
194
195 update pilot set salary = 30001 where id_pilot = 16;
```

Trigger EX10 compiled

Error starting at line : 196 in command -
update pilot set salary = 30001 where id_pilot = 16
Error report -
ORA-20001: Nu se pot actualiza informatii despre piloti in timpul sezonului
ORA-06512: at "SYSTEM.EX10", line 8
ORA-04088: error during execution of trigger 'SYSTEM.EX10'

Triggerul intoarce o eroare atunci cand este introdus sau actualizat bugetul unei echipe intr-un mod ce nu corespunde limitelor impuse de campionat.

create or replace trigger ex11

before insert or update on echipa

for each row

declare

bug echipa.buget%type;

nc campionat.num_campionat%type;

begin

bug := :NEW.buget;

select num_campionat into nc

from campionat

where id_campionat = :NEW.id_campionat;

if nc = 'Formula 1' then

if bug < 50000 or bug > 300000 then

raise_application_error(-20003, 'Bugetul echipei nu se incadreaza in limitele impuse de Formula 1');

end if;

elsif nc = 'Formula 2' then

if bug < 10000 or bug > 50000 then

raise_application_error(-20004, 'Bugetul echipei nu se incadreaza in limitele impuse de Formula 2');

```

        end if;

    end if;

end;

/

update echipa set buget = 400000 where id_echipa = 30;

```

The screenshot shows the 'Query Builder' window in Oracle SQL Developer. The script contains a trigger named 'ex11' that fires before insert or update on the 'echipa' table. The trigger logic includes a 'declare' section for variables 'bug' and 'nc', followed by a 'begin' block that sets 'bug' to ':NEW.buget;', selects 'nc' from the 'campionat' table, and then uses conditional logic to raise application errors based on the 'nc' value and the 'bug' value. The script ends with an 'update echipa set buget = 400000 where id_echipa = 30;' statement.

The 'Script Output' window shows the following error messages:

```

Error starting at line : 202 in command -
update echipa set buget = 400000 where id_echipa = 30
Error report -
ORA-20003: Bugetul echipei nu se incadreaza in limitele impuse de Formula 1
ORA-06512: at "SYSTEM.EX11", line 13
ORA-04088: error during execution of trigger 'SYSTEM.EX11'

```

12.

Triggerul introduce date in tabelul info_useri dupa ce utilizatorul a folosit o comanda LDD.

create or replace trigger ex12

after create or drop or alter on schema

begin

**insert into info_useri values(sys.login_user, sys.database_name,
sys.sysevent, sys.dictionary_obj_name, sysdate);**

end;

create table test12(testt number);

```
177 create or replace trigger ex12
178     after create or drop or alter on schema
179 begin
180     insert into info_useri values(sys.login_user, sys.database_name,
181     sys.sysevent, sys.dictionary_obj_name, sysdate);
182 end;
183
184 create table test12(testt number);
185 select * from info_useri;
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.044 seconds

	UTILIZATOR	NUME_BD	EVENTIMENT	NUME_OBJECT	DATA_INFO
1	SYSTEM	XE	CREATE	TEST12	08-JAN-21

13.

create or replace package pachet_proiect as

procedure ex6;

procedure ex7;

function ex8(v_last_name pilot.last_name%type) return number;

procedure ex9(v_nume_tara tara.nume_tara%type);

end pachet_proiect;

/

create or replace package body pachet_proiect as

procedure ex6

is

**type camp is table of campionat.id_campionat%TYPE INDEX BY
PLS_INTEGER;**

lista camp;

min_buget echipa.buget%type;

min_cnt number:=10;

max_cnt number:=0;

cnt number;

id_min_camp campionat.id_campionat%type;

id_max_camp campionat.id_campionat%type;

id_e echipa.id_echipa%type;

begin

select id_campionat

bulk collect into lista

from campionat;

for i in 1..lista.count loop

select count(*) into cnt

from echipa

where id_campionat = lista(i);

if cnt < min_cnt and cnt != 0 then

min_cnt := cnt;

id_min_camp := lista(i);

end if;

if cnt > max_cnt then

max_cnt := cnt;

id_max_camp := lista(i);

select min(buget) into min_buget

from echipa

where id_campionat = lista(i);

end if;

end loop;

**insert into echipa values(id_echipe.nextval, 'EX6 RACING',
10000, 0, id_min_camp);**

**dbms_output.put_line('A fost inserata echipa in campionatul
cu id ' || id_min_camp);**

select id_echipa into id_e

from echipa

where id_campionat = id_max_camp and buget = min_buget;

```
        dbms_output.put_line('Echipa cu cel mai mic buget are id-ul '
|| id_e);
```

```
end ex6;
```

```
procedure ex7
```

```
is
```

```
    cursor c is
```

```
        select id_fabrica, nume_fabrica
        from fabrica;
```

```
    cursor d(param number) is
```

```
        select pret_prod
        from componente
        where id_fabrica = param;
```

```
    suma number;
```

```
begin
```

```
    for i in c loop
```

```
        DBMS_OUTPUT.PUT_LINE('-----');
```

```
        DBMS_OUTPUT.PUT_LINE ('FABRICA: ' || i.nume_fabrica);
```

```
        suma := 0;
```

```
        for k in d(i.id_fabrica) loop
```

```
            suma := suma+k.pret_prod;
```

```
        end loop;
```

```
        DBMS_OUTPUT.PUT_LINE ('PRET TOTAL: ' || suma);
```

```

        DBMS_OUTPUT.PUT_LINE('-----');
    end loop;

end ex7;

function ex8(v_last_name pilot.last_name%type) return number
is
    type echipe is table of echipa.id_echipa%TYPE INDEX BY
    PLS_INTEGER;
    lista echipe;
    id_e pilot.id_echipa%type;
    nume campionat.nume_campionat%type;
    total number := 0;
    id_camp campionat.id_campionat%type;
    ok number;
begin
    select id_echipa into id_e
    from pilot
    where last_name = v_last_name;

    select c.id_campionat into id_camp
    from echipa e join campionat c on e.id_campionat =
    c.id_campionat
    where e.id_echipa = id_e;

```

```
select id echipa
bulk collect into lista
from echipa
where id_campionat = id_camp;
```

```
for i in (select id echipa, victorii
          from pilot) loop
    ok := 0;
    for j in 1..lista.count loop
        if i.id echipa = lista(j) then
            ok := 1;
        end if;
        exit when ok = 1;
    end loop;
```

```
    if ok = 1 then
        total := total + i.victorii;
    end if;
end loop;
```

```
return total;
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN
```

```
        DBMS_OUTPUT.PUT_LINE ('Nu exista pilot cu acest nume');
```

```
        return -1;

        WHEN TOO_MANY_ROWS THEN

            DBMS_OUTPUT.PUT_LINE ('Exista mai multi piloti cu acest
nume');

            return -2;

end;
```

```
procedure ex9(v_nume_tara tara.nume_tara%type)
is
    id_tr tara.id_tara%type;

    type circuite is table of circuit.id_circuit%type index by
pls_integer;

    id_c circuite;

    type piloti is table of pilot.id_pilot%TYPE INDEX BY
PLS_INTEGER;

    lista piloti;

    nr_spec number;

    max_spec number := 0;

    id_c_max circuit.id_circuit%type;

    min1 number:= 999999;

    minteam echipa.id echipa%type;

    id_pilot_min pilot.id_pilot%type;

    id echipa_min pilot.id echipa%type;
```

```
id_echipa_max pilot.id_echipa%type;
titles echipa.titluri_castigate%type;
sal pilot.salary%type;
team echipa.id_echipa%type;
exceptie EXCEPTION;
begin

select id_tara into id_tr
from tara
where nume_tara = v_nume_tara;

select id_circuit bulk collect into id_c
from circuit
where id_tara = id_tr;

for i in 1..id_c.count loop
    select nr_spectatori into nr_spec
    from circuit
    where id_circuit = id_c(i);

    if nr_spec > max_spec then
        max_spec := nr_spec;
        id_c_max := id_c(i);
    end if;
end loop;
```

```
select id_pilot
bulk collect into lista
from testeaza_la
where id_circuit = id_c_max;
```

```
for i in 1..lista.count loop
    select salary, id echipa into sal, team
    from pilot
    where id_pilot = lista(i);
```

```
    if sal < min1 then
        min1 := sal;
        minteam := team;
        id_pilot_min := lista(i);
    end if;
end loop;
```

```
select id echipa, titluri_castigate into id echipa_max, titles
from echipa
where titluri_castigate in (select max(titluri_castigate)
                             from echipa);
```

```
if minteam = id echipa_max then
    raise exceptie;
end if;
```



```
update pilot set id echipa = id echipa_max,  
               salary = salary * 2,  
               hire_date = SYSDATE,  
               contract = 1  
where id_pilot = id_pilot_min;
```

```
select salary into sal  
from pilot  
where id_pilot = id_pilot_min;
```

```
update echipa set buget = buget - sal where id echipa =  
id echipa_max;
```

```
insert into fabrica values(new_factory.nextval, 'New Factory',  
'New Street', minteam);
```

```
update echipa set buget = buget + sal/4 where id echipa =  
minteam;
```

EXCEPTION

WHEN NO_DATA_FOUND THEN

DBMS_OUTPUT.PUT_LINE('Nu exista tara cu acest nume');

WHEN exceptie THEN

**DBMS_OUTPUT.PUT_LINE('Echipa la care trebuie transferat
pilotul corespunde cu echipa actuala');**

end ex9;

end pachet_proiect;

/

begin

pachet_proiect.ex6;

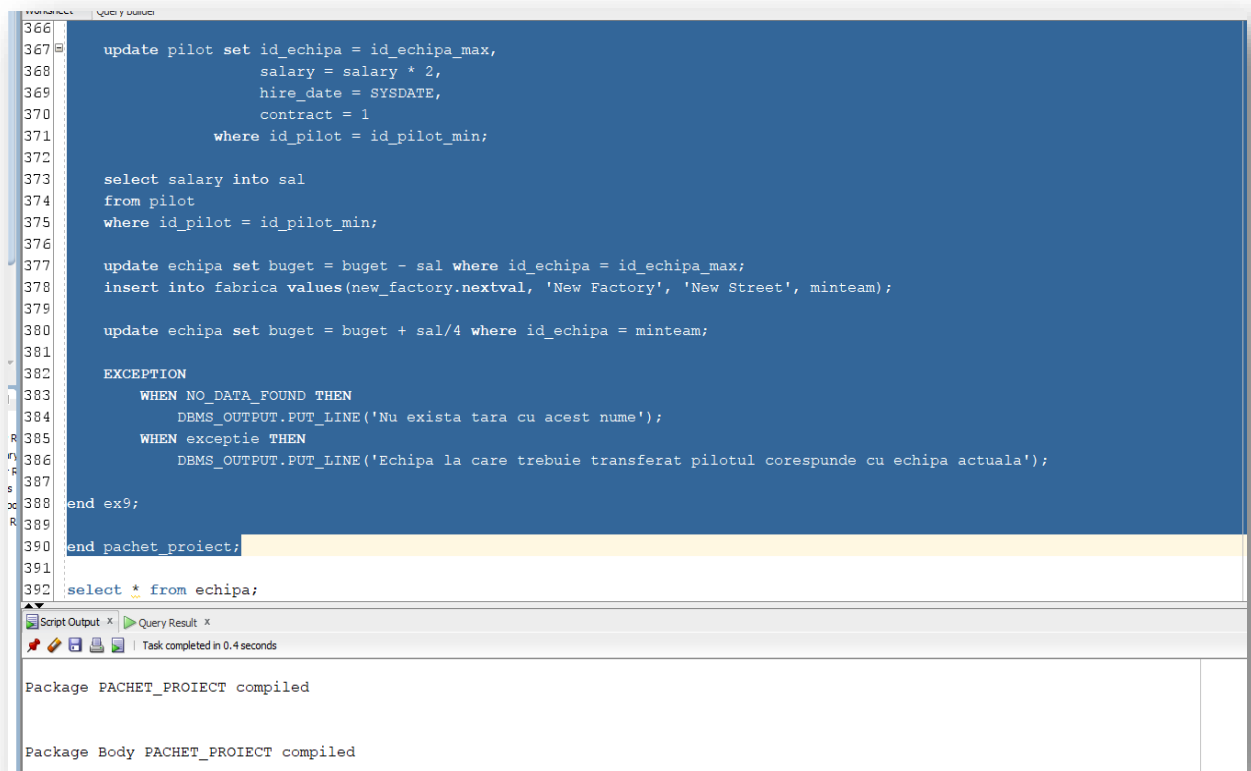
pachet_proiect.ex7;

DBMS_OUTPUT.PUT_LINE(pachet_proiect.ex8('Vettel'));

pachet_proiect.ex9('Italia');

end;

/



The screenshot displays the Oracle SQL Developer environment. The main window shows a PL/SQL script with line numbers 366 through 392. The script includes several SQL and PL/SQL statements: an update on the 'pilot' table, a select statement for salary, an update on the 'echipa' table, an insert into the 'fabrica' table, and an exception block. The script concludes with 'end ex9;', 'end pachet_proiect;', and a final 'select * from echipa;' statement. Below the script editor, the 'Script Output' pane shows the results of the execution, indicating that the package 'PACHET_PROIECT' and its body were successfully compiled. The status bar at the bottom indicates that the task was completed in 0.4 seconds.

```
366 update pilot set id echipa = id echipa_max,
367 salary = salary * 2,
368 hire_date = SYSDATE,
369 contract = 1
370 where id_pilot = id_pilot_min;
371
372
373 select salary into sal
374 from pilot
375 where id_pilot = id_pilot_min;
376
377 update echipa set buget = buget - sal where id echipa = id echipa_max;
378 insert into fabrica values(new_factory.nextval, 'New Factory', 'New Street', minteam);
379
380 update echipa set buget = buget + sal/4 where id echipa = minteam;
381
382 EXCEPTION
383 WHEN NO_DATA_FOUND THEN
384 DBMS_OUTPUT.PUT_LINE('Nu exista tara cu acest nume');
385 WHEN exceptie THEN
386 DBMS_OUTPUT.PUT_LINE('Echipa la care trebuie transferat pilotul corespunde cu echipa actuala');
387
388 end ex9;
389
390 end pachet_proiect;
391
392 select * from echipa;
```

Script Output x Query Result x

Task completed in 0.4 seconds

Package PACHET_PROIECT compiled

Package Body PACHET_PROIECT compiled

WorksheetQuery Builder

392begin

393pachet_proiect.ex6;

394pachet_proiect.ex7;

395DBMS_OUTPUT.PUT_LINE(pachet_proiect.ex8('Vettel'));

396pachet_proiect.ex9('Italia');

397end;

DBMS Output

Buffer Size:20000

PROJECT

A fost inserata echipa in campionatul cu id 2

Echipa cu cel mai mic buget are id-ul 31

FABRICA: Ferrari Factory

PRET TOTAL: 10079999

FABRICA: Red Bull Racing Factory

PRET TOTAL: 8100000

FABRICA: Mercedes AMG Petronas Factory

PRET TOTAL: 400000

FABRICA: Prema Racing Factory

PRET TOTAL: 0

FABRICA: Techeetah Factory

PRET TOTAL: 0

FABRICA: New Factory

PRET TOTAL: 0

251