

I.DESCRIEREA TEMEI ECONOMICE

1.Descrierea temei economice alese:

Proiectul constă în crearea unei baze de date pentru un sistem bancar prin intermediul căruia se pot acumula și concretiza informații legate de clienti, conturile acestora, cardurile, tranzacțiile, creditele și alte informații. Aceasta baza de date creată are ca scop principal centralizarea informațiilor și facilitarea accesului la datele necesare pentru operațiunile bancare cum ar fi acordarea unui credit bancar clientilor, monitorizarea tranzacțiilor asociate cardurilor unor clienti în vederea tranzacțiilor ulterioare, dar și anumite informații precise despre emiterea unui extras de cont sau ratele asociate creditelor acordate de către banca.

Ulterior, pe lângă tabelele create de baza care tin informații despre client și activitatea lor financiară, sunt create pe parcurs alte două tabele asociate informațiilor despre banca și angajatii acesteia, reflectand, totodata, și date despre relația angajat-client.

2.Descrierea bazei de date:

Obiectiv: obiectivul proiectului este acela de a proiecta și implementa o baza de date care să permită administrarea datelor clientilor, gestionarea conturilor bancare și informațiilor legate de acestea, evidența tranzacțiilor financiare. Descrierea tabelelor și a atributelor, precizarea restricțiilor și a tipurilor de legături:Baza de date este construită din tabela principală CLIENTI_PROIECT care conține date despre clienti, având restricții de tipul PRIMARY KEY(id_client), NOT NULL(ume_client, prenume_client), CHECK(coloanele telefon – să aibă exact 10 cifre și cnp – să aibă exact 13 cifre). Pe lângă tabela principală, mai sunt și alte tabele ce tin informații despre conturile și cardurile clientilor: CONTURI_PROIECT, CARDURI_PROIECT, care sunt legate prin cheia externă id_client de tabela CLIENTI_PROIECT.Celelalte tabele detin informații despre activitatea financiară a clientilor:INCASARI_PROIECT, TRANZACTII_PROIECT, TRANSFER_CATRE_ALTE_BANCI_PROIECT, CREDITE_PROIECT, RATE_CREDIT_PROIECT, ce sunt legate cu tabelele de baza prin chei externe fie id_cont, fie id_client, fie id_credit.

*****TABELE ADAUGATE ULTERIOR PRIN COMENZILE LDD*****

BANCA_PROIECT:Tabela referitoare la banchile din România (o parte), având coloanele: id_banca, PRIMARY KEY, denumire, adresa, an_fondare, id_client, FOREIGN KEY și id_angajat FOREIGN KEY.

ANGAJATI_PROIECT:Tabela referitoare la informațiile legate de angajati unei banchi, având coloanele: id_angajat, PRIMARY KEY, ume_angajat, prenume_angajat, varsta, salariul, id_client ca FOREIGN KEY, id_manager și id_departamente ca FOREIGN KEY.

DEPARTAMENTE_PROIECT:Tabela ce tine informații despre departamentele din cadrul banchii, având coloanele: id_departamente, PRIMARY KEY și denumire_departament.

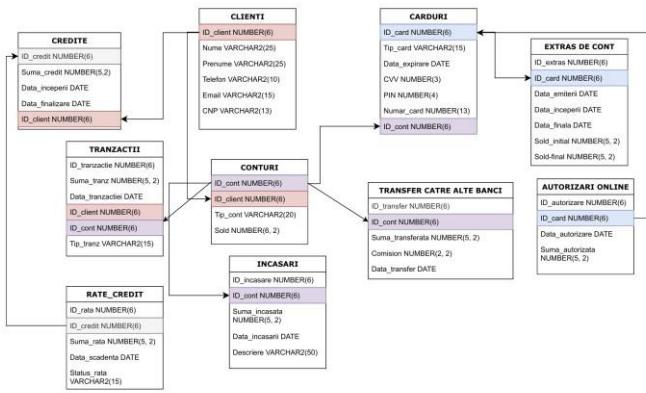
3.Tipurile de legături dintre tabele:

In general, legăturile dintre tabele sunt de forma 1:N (one to many) care poate fi evidențiată prin următoarele tabele: banca_PROIECT și clienti_PROIECT (o bancă poate avea mai mulți clienti), respectiv clienti_PROIECT și carduri_PROIECT (un client poate avea mai multe carduri) sau clienti_PROIECT și conturi_PROIECT (un client poate avea mai multe conturi) etc.

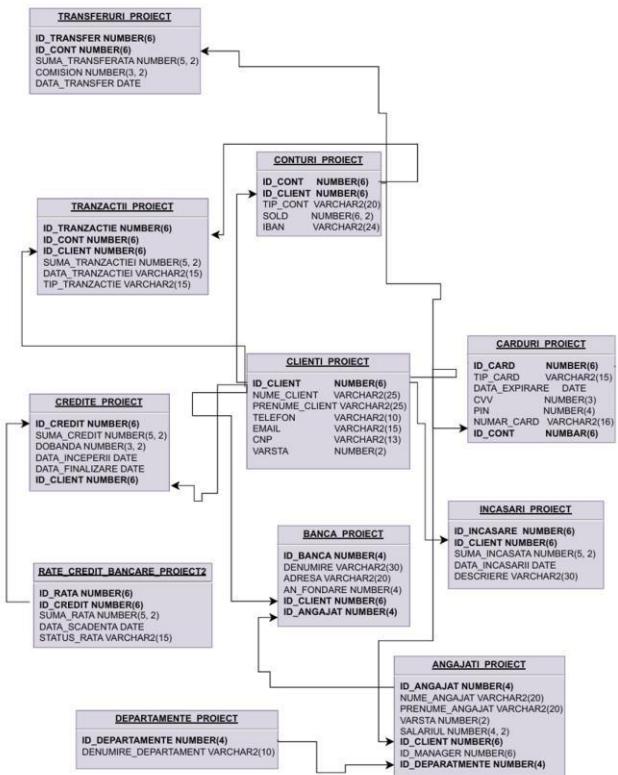
II. SCHEMA CONCEPTUALA PENTRU MODELAREA TEMEI ALESE

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

***Schema conceptuala initiala:



***Schema conceptuala dupa comenziile LDD efectuate:



III. CONSTRUIREA BAZEI DE DATE–tabele, legături între tabele și restricții de integritate. Exemplificarea operațiilor LDD (CREATE, ALTER, DROP) asupra tabelelor (minim 8).

1) Crearea tabelelor:

1.Crearea tabelei client_PROJECT CREATE
TABLE clienti_PROJECT(

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
id_client NUMBER(6) CONSTRAINT pk_id_client10 PRIMARY KEY,  
nume_client  VARCHAR2(25)  NOT  NULL,  prenume_client  
VARCHAR2(25) NOT NULL,  
telefon VARCHAR2(10) CONSTRAINT check_telefon CHECK (LENGTH(telefon)=10), email  
VARCHAR2(15),  
CNP VARCHAR2(13) CONSTRAINT check_cnp CHECK (LENGTH(CNP)=13));
```

2.Crearea tabelei conturi_PROIECT CREATE

```
TABLE conturi_PROIECT(  
id_cont NUMBER(6) CONSTRAINT pk_id_cont10 PRIMARY KEY, id_client  
NUMBER(6),  
CONSTRAINT fk_id_client FOREIGN KEY(id_client) REFERENCES clienti_PROIECT(id_client),  
tip_cont VARCHAR2(20), sold NUMBER(6, 2));
```

3.Crearea tabelei carduri_PROIECT CREATE

```
TABLE carduri_PROIECT(  
id_card NUMBER(6) CONSTRAINT pk_id_card10 PRIMARY KEY, tip_card  
VARCHAR2(15),  
data_expirare DATE,  
CVV  NUMBER(3),  PIN  
NUMBER(4),  numar_card  
VARCHAR2(16),  id_cont  
NUMBER(6),  
CONSTRAINT fk_id_cont FOREIGN KEY(id_cont) REFERENCES conturi_PROIECT(id_cont));
```

4.Crearea tabelei incasari_PROIECT CREATE

```
TABLE incasari_PROIECT(  
id_incasare NUMBER(6) CONSTRAINT pk_id_incasare10 PRIMARY KEY, id_client  
NUMBER(6),  
CONSTRAINT fk_id_client2 FOREIGN KEY(id_client) REFERENCES clienti_PROIECT(id_client),  
suma_incasata NUMBER(5, 2), data_incasarii DATE, descriere VARCHAR2(30));
```

5.Crearea tabelei tranzactii_PROIECT CREATE

```
TABLE tranzactii_PROIECT(  
id_tranzactie NUMBER(6) CONSTRAINT pk_id_tranzactie10 PRIMARY KEY, id_cont  
NUMBER(6),  
CONSTRAINT fk_id_cont2 FOREIGN KEY(id_cont) REFERENCES conturi_PROIECT(id_cont),  
id_client NUMBER(6),  
CONSTRAINT fk_id_client3 FOREIGN KEY(id_client) REFERENCES clienti_PROIECT(id_client),  
suma_tranzactiei NUMBER(5, 2), data_tranzactiei DATE, tip_tranzactie VARCHAR2(15));
```

6.Crearea tabelei transfer_catre_alte_banci_PROIECT CREATE

```
TABLE transfer_catre_alte_banci_PROIECT(  
id_transfer NUMBER(6) CONSTRAINT pk_id_transfer10 PRIMARY KEY, id_cont  
NUMBER(6),  
CONSTRAINT fk_id_cont3 FOREIGN KEY(id_cont) REFERENCES conturi_PROIECT(id_cont),  
suma_transferata NUMBER(5, 2), comision NUMBER(3, 2), data_transfer DATE);
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

7.Crearea tableei credite_PROIECT CREATE

```
TABLE credite_PROIECT(
id_credit NUMBER(6) CONSTRAINT fk_id_credit10 PRIMARY KEY,
suma_credit NUMBER(5, 2), dobanda NUMBER(3, 2),
data_inceperii DATE,
data_finalizare DATE, id_client
NUMBER(6),
CONSTRAINT fk_id_client4 FOREIGN KEY(id_client) REFERENCES clienti_PROIECT(id_client));
```

8.Crearea tableei rate_credit_PROIECT CREATE

```
TABLE rate_credit_PROIECT(
id_rata NUMBER(6) CONSTRAINT pk_id_rata10 PRIMARY KEY, id_credit
NUMBER(6),
CONSTRAINT fk_id_credit FOREIGN KEY(id_credit) REFERENCES credite_PROIECT(id_credit),
suma_rata NUMBER(5, 2), data_scadenta DATE, status_rata VARCHAR2(15));
```

2)Comenzi LDD-create, drop, alter

1)Modificarea coloanei sold prin adaugarea unei restrictii

```
--1)Modificarea coloanei sold din tabela conturi_PROJECT
SELECT USER FROM DUAL;
ALTER TABLE conturi_PROJECT
MODIFY (sold NUMBER(6, 2) DEFAULT 0 CONSTRAINT check_sold CHECK (sold>=0));
```

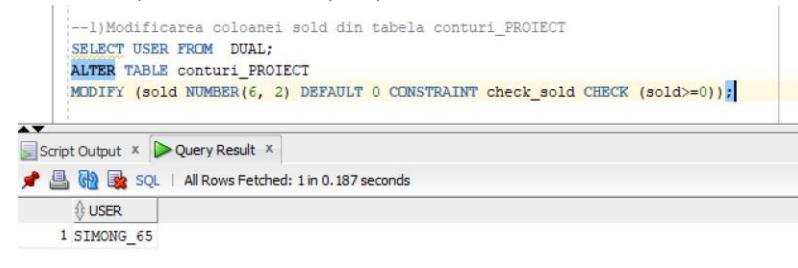


Table CONTURI_PROJECT altered.

2)Redenumirea tableei rate_credit_bancare_PROJECT cu rate_credit_bancare_PROJECT2.

```
--2)Redenumirea tableei rate_credit_bancare_PROJECT cu rate_credit_bancare_PROJECT2
SELECT USER FROM DUAL;
RENAME rate_credit_bancare_PROJECT TO rate_credit_bancare_PROJECT2;
```



Table renamed.

PROIECT – BAZE DE DATE – SIMON GEORGIANA

3) Adaugarea coloanei IBAN in tabela Conturi_PROIECT de tipul VARCHAR2(24) cu restrictia ca acesta sa aiba exact 24 de caractere.

```
SELECT USER FROM DUAL;  
ALTER TABLE conturi_PROJECT  
ADD (iban VARCHAR2(24)) CONSTRAINT ck_ibан CHECK(LENGTH(iban)=24);
```

```
--3)Adaugarea coloanelor IBAN in tabela Conturi_PROJECT de tipul VARCHAR2(24) cu restrictia ca acesta sa aiba exact 24 de caractere.  
SELECT USER FROM DUAL;  
ALTER TABLE conturi_PROJECT  
ADD (iban VARCHAR2(24)) CONSTRAINT ck_ibан CHECK(LENGTH(iban)=24);  
  
Table renamed.  
  
Table CONTURI_PROJECT altered.
```

4) Adaugarea coloanelor varsta si adresa in tabela Clienti_Conturi cu restrictia ca varsta sa fie mai mare sau egala cu 18.

```
SELECT USER FROM DUAL;  
ALTER TABLE clienti_PROJECT  
ADD (varsta NUMBER(2)) CONSTRAINT ck_varsta10 CHECK (varsta>=18), adresa  
VARCHAR2(25));
```

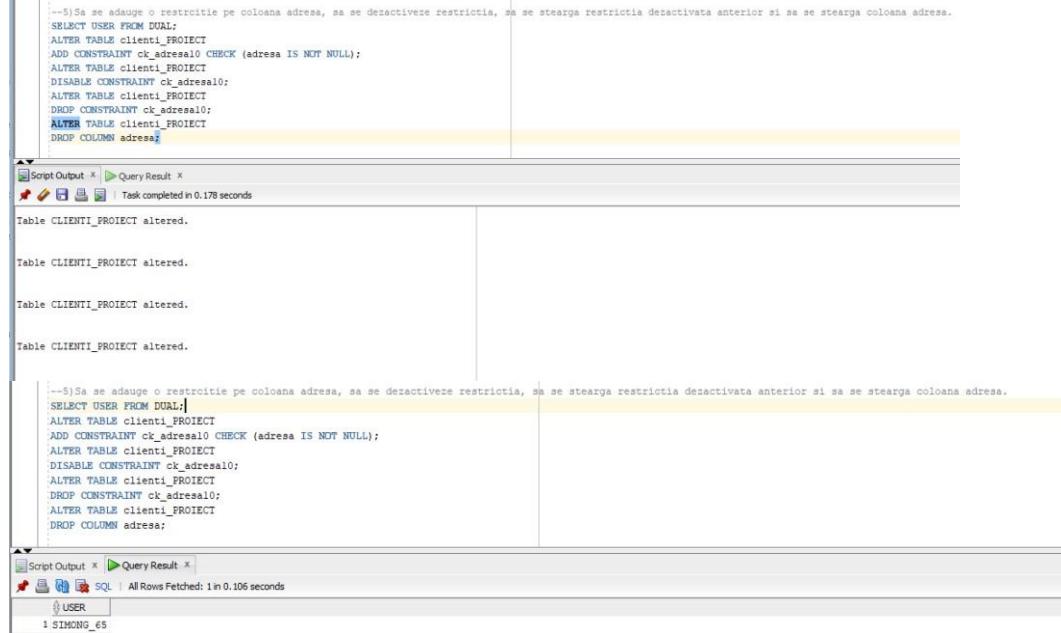
```
--4)Adaugarea coloanelor varsta si adresa in tabela Clienti_PROJECT cu restrictia ca varsta sa fie mai mare sau egala cu 18.  
SELECT USER FROM DUAL;  
ALTER TABLE clienti_PROJECT  
ADD (varsta NUMBER(2)) CONSTRAINT ck_varsta10 CHECK (varsta>=18),  
adresa VARCHAR2(25));  
  
Table renamed.  
  
Table CONTURI_PROJECT altered.
```

5) Sa se adauge o restrictie pe coloana adresa, sa se dezactiveze restrictia, sa se stearga restrictia dezactivata anterior si sa se stearga coloanal adresa.

```
SELECT USER FROM DUAL;  
ALTER TABLE clienti_PROJECT --1  
ADD CONSTRAINT ck_adresa10 CHECK (adresa IS NOT NULL);  
ALTER TABLE clienti_PROJECT --2
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
DISABLE CONSTRAINT ck_adresa10;  
ALTER TABLE clienti_PROJECT --3  
DROP CONSTRAINT ck_adresa10;  
ALTER TABLE clienti_PROJECT --4  
DROP COLUMN adresa;
```



```
--5)Sa se adauge o restricție pe coloana adresa, sa se dezactiveze restrictia, sa se stearga restrictia dezactivata anterior si sa se stearga coloana adresa.  
SELECT USER FROM DUAL;  
ALTER TABLE clienti_PROJECT  
ADD CONSTRAINT ck_adresa10 CHECK (adresa IS NOT NULL);  
ALTER TABLE clienti_PROJECT  
DISABLE CONSTRAINT ck_adresa10;  
ALTER TABLE clienti_PROJECT  
DROP CONSTRAINT ck_adresa10;  
ALTER TABLE clienti_PROJECT  
DROP COLUMN adresa;
```

```
Script Output x | Query Result x  
Task completed in 0.178 seconds
```

```
Table CLIENTI_PROJECT altered.  
Table CLIENTI_PROJECT altered.  
Table CLIENTI_PROJECT altered.  
Table CLIENTI_PROJECT altered.
```

```
--5)Sa se adauge o restricție pe coloana adresa, sa se dezactiveze restrictia, sa se stearga restrictia dezactivata anterior si sa se stearga coloana adresa.  
SELECT USER FROM DUAL;  
ALTER TABLE clienti_PROJECT  
ADD CONSTRAINT ck_adresa10 CHECK (adresa IS NOT NULL);  
ALTER TABLE clienti_PROJECT  
DISABLE CONSTRAINT ck_adresa10;  
ALTER TABLE clienti_PROJECT  
DROP CONSTRAINT ck_adresa10;  
ALTER TABLE clienti_PROJECT  
DROP COLUMN adresa;
```

```
Script Output x | Query Result x  
SQL : All Rows Fetched: 1 in 0.106 seconds
```

USER
SIMONG_65

6)Sa se creeze tabela fosti_clienti_PROJECT pe baza tablei client_PROJECT, iar apoi sa se adauge o coloana in tabela fosti_clienti cu o restrictie de integritate si sa se stearga tabela fosti_clienti_PROJECT fara posibilitatea de recuperare.

```
SELECT USER FROM DUAL;  
CREATE TABLE fosti_clienti_PROJECT  
AS  
SELECT *FROM clienti_PROJECT;  
ALTER TABLE fosti_clienti_PROJECT  
ADD (salariul NUMBER(4, 2) CONSTRAINT ck_salariul10 CHECK (salariul IS NOT NULL));  
DROP TABLE fosti_clienti_PROJECT PURGE;
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
--6)Sa se creeze tabela fosti_clienti_PROJECT pe baza tablei client_PROJECT, iar apoi sa se adauge o coloana in tabela fosti_clienti
--cu o restrictie de integritate si a sa se stearga tabela fosti_clienti_PROJECT fara posibilitatea de recuperare.
SELECT USER FROM DUAL;
CREATE TABLE fosti_clienti_PROJECT
AS
SELECT *FROM clienti_PROJECT;
ALTER TABLE fosti_clienti_PROJECT
ADD (salariu NUMBER(4, 2) CONSTRAINT ck_salariu10 CHECK (salariu IS NOT NULL));
DROP TABLE fosti_clienti_PROJECT PURGE;
```

Script Output X | Query Result X
Task completed in 0.193 seconds

Table CLIENTI_PROJECT altered.

Table FOSTI_CLIENTI_PROJECT created.

Table FOSTI_CLIENTI_PROJECT altered.

Table FOSTI_CLIENTI_PROJECT dropped.

```
--6)Sa se creeze tabela fosti_clienti_PROJECT pe baza tablei client_PROJECT, iar apoi sa se adauge o coloana in tabela fosti_clienti
--cu o restrictie de integritate si a sa se stearga tabela fosti_clienti_PROJECT fara posibilitatea de recuperare.
SELECT USER FROM DUAL;
CREATE TABLE fosti_clienti_PROJECT
AS
SELECT *FROM clienti_PROJECT;
ALTER TABLE fosti_clienti_PROJECT
ADD (salariu NUMBER(4, 2) CONSTRAINT ck_salariu10 CHECK (salariu IS NOT NULL));
DROP TABLE fosti_clienti_PROJECT PURGE;
```

Script Output X | Query Result X
All Rows Fetched: 1 in 0.425 seconds

USER
1 SIMONG_65

7)Sa se creeze tablele Banca_PROJECT si Angajati_PROJECT.

SELECT USER FROM DUAL; CREATE

TABLE banca_PROJECT(

```
id_banca NUMBER(4) CONSTRAINT pk_id_banca10 PRIMARY KEY,
denumire VARCHAR2(30) NOT NULL,   adresa VARCHAR2(20),
an_fondare NUMBER(4)); CREATE TABLE angajati_PROJECT(
id_angajat NUMBER(4) CONSTRAINT pk_id_angajat10 PRIMARY KEY,
nume_angajat VARCHAR2(20), prenume_angajat VARCHAR2(20),
varsta NUMBER(2) CONSTRAINT ck_varsta_10 CHECK( varsta BETWEEN 18 AND 65), salariu
NUMBER(4, 2) CONSTRAINT ck_salariu10 CHECK(salariu>0));
```

```
--7)Sa se creeze tablele Banca_PROJECT si Angajati_PROJECT.
CREATE TABLE banca_PROJECT(
id_banca NUMBER(4) CONSTRAINT pk_id_banca10 PRIMARY KEY,
denumire VARCHAR2(30) NOT NULL,
adresa VARCHAR2(20),
an_fondare NUMBER(4));

CREATE TABLE angajati_PROJECT(
id_angajat NUMBER(4) CONSTRAINT pk_id_angajat10 PRIMARY KEY,
nume_angajat VARCHAR2(20),
prenume_angajat VARCHAR2(20),
varsta NUMBER(2) CONSTRAINT ck_varsta_10 CHECK( varsta BETWEEN 18 AND 65),
salariu NUMBER(4, 2) CONSTRAINT ck_salariu10 CHECK(salariu>0));
```

```
Script Output X | Query Result X  
Task completed in 1.462 seconds
ORA-00907: missing right parenthesis
```

Table BANCA_PROJECT created.

```
--7)Sa se creeze tablele Banca_PROJECT si Angajati_PROJECT.
SELECT USER FROM DUAL;
```

```
CREATE TABLE banca_PROJECT(
id_banca NUMBER(4) CONSTRAINT pk_id_banca10 PRIMARY KEY,
denumire VARCHAR2(30) NOT NULL,
adresa VARCHAR2(20),
an_fondare NUMBER(4));

CREATE TABLE angajati_PROJECT(
id_angajat NUMBER(4) CONSTRAINT pk_id_angajat10 PRIMARY KEY,
nume_angajat VARCHAR2(20),
prenume_angajat VARCHAR2(20),
varsta NUMBER(2) CONSTRAINT ck_varsta_10 CHECK( varsta BETWEEN 18 AND 65),
salariu NUMBER(4, 2) CONSTRAINT ck_salariu10 CHECK(salariu>0));
```

```
--7)sa se creeze tablele BANCA_PROJECT si ANGAJATI_PROJECT.
CREATE TABLE banca_PROJECT(
id_banca NUMBER(4) CONSTRAINT pk_id_banca10 PRIMARY KEY,
denumire VARCHAR2(30) NOT NULL,
adresa VARCHAR2(20),
an_fondare NUMBER(4));

CREATE TABLE angajati_PROJECT(
id_angajat NUMBER(4) CONSTRAINT pk_id_angajat10 PRIMARY KEY,
nume_angajat VARCHAR2(20),
prenume_angajat VARCHAR2(20),
varsta NUMBER(2) CONSTRAINT ck_varsta_10 CHECK( varsta BETWEEN 18 AND 65),
salariu NUMBER(4, 2) CONSTRAINT ck_salariu10 CHECK(salariu>0));
```

Script Output X | Query Result X
Task completed in 1.462 seconds

[https://docs.oracle.com/error-help/db/ora-00906/00906.00000 - "missing left parenthesis"](https://docs.oracle.com/error-help/db/ora-00906/00906.00000 -)

*Cause: A required left parenthesis has been omitted. Certain commands, such as CREATE TABLE, CREATE CLUSTER, and INSERT, also require around subqueries in WHERE clauses and in UPDATE table SET column = (SELECT...) statements.

*Action: Correct the syntax, inserting a left parenthesis where required, and retry the statement.

Table ANGAJATI_PROJECT created.

Script Output X | Query Result X
All Rows Fetched: 1 in 0.065 seconds

USER
1 SIMONG_65

PROIECT – BAZE DE DATE – SIMON GEORGIANA

8) Adaugarea unor cheie externe in tabela banca_PROIECT si o cheie externa in tabela angajati_PROIECT.

SELECT USER FROM DUAL;

ALTER TABLE banca_PROIECT

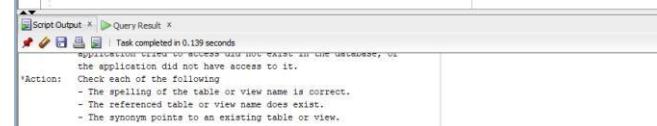
ADD (id_client NUMBER (6), CONSTRAINT fk_id_clients FOREIGN KEY(id_client) REFERENCES clienti_PROIECT(id_client),

id_angajat NUMBER(4), CONSTRAINT fk_id_angajat FOREIGN KEY (id_angajat) REFERENCES angajati_PROIECT(id_angajat));

ALTER TABLE angajati_PROIECT

ADD (id_client NUMBER (6), CONSTRAINT fk_id_client6 FOREIGN KEY(id_client) REFERENCES clienti_PROIECT(id_client), id_manager NUMBER(6));

```
--8)Adaugarea unor cheie externe in tabela banca_PROJECT si tabela angajati_PROJECT
SELECT USER FROM DUAL;
ALTER TABLE banca_PROJECT
ADD (id_client NUMBER (6), CONSTRAINT fk_id_clients FOREIGN KEY(id_client) REFERENCES clienti_PROJECT(id_client),
id_angajat NUMBER(4), CONSTRAINT fk_id_angajat FOREIGN KEY (id_angajat) REFERENCES angajati_PROJECT(id_angajat));
ALTER TABLE angajati_PROJECT
ADD (id_client NUMBER (6), CONSTRAINT fk_id_client6 FOREIGN KEY(id_client) REFERENCES clienti_PROJECT(id_client),
id_manager NUMBER(6));
```



Script Output | Query Result | Task completed in 0.139 seconds
ORA-00942: table or view does not exist in the username, or the application did not have access to it.
Action:
- Check each of the following
- The spelling of the table or view name is correct.
- The referenced table or view name does exist.
- The synonym points to an existing table or view.

Table BANCA_PROJECT altered.

Table ANGAJATI_PROJECT altered.

```
--8)Adaugarea unor cheie externe in tabela banca_PROJECT si tabela angajati_PROJECT
SELECT USER FROM DUAL;
ALTER TABLE banca_PROJECT
ADD (id_client NUMBER (6), CONSTRAINT fk_id_clients FOREIGN KEY(id_client) REFERENCES clienti_PROJECT(id_client),
id_angajat NUMBER(4), CONSTRAINT fk_id_angajat FOREIGN KEY (id_angajat) REFERENCES angajati_PROJECT(id_angajat));
ALTER TABLE angajati_PROJECT
ADD (id_client NUMBER (6), CONSTRAINT fk_id_client6 FOREIGN KEY(id_client) REFERENCES clienti_PROJECT(id_client),
id_manager NUMBER(6));
```



Script Output | Query Result | All Rows Fetched: 1 in 0.095 seconds

9) Sa se adauge coloana id_departament in tabela Angajati_Banca ca find o cheie externa a tablei departamente_PROIECT (trebuie creata). select user from dual;

CREATE TABLE departamente_PROIECT(

id_departamente NUMBER (4) CONSTRAINT pk_departamente_id10 PRIMARY KEY,
denumire_departament VARCHAR2(10));

ALTER TABLE angajati_PROIECT

ADD (id_departamente NUMBER(4), CONSTRAINT fk_id_departament FOREIGN KEY
(id_departamente) REFERENCES departamente_PROIECT(id_departamente));

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
--10)Sa se adauge coloana id_departament in tabela Angajati_Banca ca fiind o cheie externa a tabelui departamente_PROJECT (trebuie creats).
select user from dual;
CREATE TABLE departamente_PROJECT(
id_departamente NUMBER(4) CONSTRAINT pk_departamente_idIO PRIMARY KEY,
denumire_departament VARCHAR2(10));
ALTER TABLE angajati_PROJECT
ADD (id_departamente NUMBER(4), CONSTRAINT fk_id_departament FOREIGN KEY (id_departamente)
REFERENCES departamente_PROJECT(id_departamente));
```

Table BANCA_PROJECT altered.

Table ANGAJATI_PROJECT altered.

Table DEPARTAMENTE_PROJECT created.

Table ANGAJATI_PROJECT altered.

```
--11)Sa se adauge coloana id_departament in tabela Angajati_Banca ca fiind o cheie externa a tabel
select user from dual;
CREATE TABLE departamente_PROJECT(
id_departamente NUMBER(4) CONSTRAINT pk_departamente_idIO PRIMARY KEY,
denumire_departament VARCHAR2(10));
ALTER TABLE angajati_PROJECT
ADD (id_departamente NUMBER(4), CONSTRAINT fk_id_departament FOREIGN KEY (id_departamente)
REFERENCES departamente_PROJECT(id_departamente));
```

Table ANGAJATI_PROJECT altered.

10)Sa se modifice restrictia de integritate pe coloanal suma_incasata din tabela incasari_PROJECT in care sa se verifice daca suma este in intervalul 1 - 1.000.000.

SELECT USER FROM DUAL;

ALTER TABLE incasari_PROJECT

ADD CONSTRAINT ck_suma_incasata10 CHECK(suma_incasata BETWEEN 1 AND 1000000);

```
--11)Sa se modifice restrictia de integritate pe coloanal suma_incasata
--din tabela incasari_PROJECT in care sa se verifice daca suma este
--in intervalul 1 - 1.000.000.
SELECT USER FROM DUAL;
ALTER TABLE incasari_PROJECT
ADD CONSTRAINT ck_suma_incasata10 CHECK(suma_incasata BETWEEN 1 AND 1000000);
```

Table DEPARTAMENTE_PROJECT created.

Table ANGAJATI_PROJECT altered.

Table AUTORIZARI_ONLINE_PROJECT altered.

Table INCASARI_PROJECT altered.

```
--11)Sa se modifice restrictia de integritate pe coloanal suma_incasata
--in tabela incasari_PROJECT in care sa se verifice daca suma este
--in intervalul 1 - 1.000.000.
SELECT USER FROM DUAL;
ALTER TABLE incasari_PROJECT
ADD CONSTRAINT ck_suma_incasata10 CHECK(suma_incasata BETWEEN 1 AND 1000000);
```

Table INCASARI_PROJECT altered.

11)Sa se adauge o restrictie de integritate pe coloanal data_finalizare din tabela credite_bancare_PROJECT2 astfel incat sa se verifica ca data de finalizare sa fie mai mare decat data de incepere a creditului bancar.

SELECT USER FROM DUAL;

ALTER TABLE credite_PROJECT

ADD CONSTRAINT ck_data_finalizare10 CHECK(data_finalizare>data_incepereii);

```
--12)Sa se adauge o restrictie de integritate pe coloana data_finalizare din tabela credite_PROJECT
--astfel incat sa se verifice ca data de finalizare sa fie mai mare decat data de incepere a creditului bancar.
SELECT USER FROM DUAL;
ALTER TABLE credite_PROJECT
ADD CONSTRAINT ck_data_finalizare10 CHECK(data_finalizare>data_incepereii);
```

Table ANGAJATI_PROJECT altered.

Table AUTORIZARI_ONLINE_PROJECT altered.

Table INCASARI_PROJECT altered.

Table CREDITE_PROJECT altered.

```
--12)Sa se adauge o restrictie de integritate pe
--astfel incat sa se verifice ca data de finalizare sa fie mai mare decat data de incepere a creditului bancar.
SELECT USER FROM DUAL;
ALTER TABLE credite_PROJECT
ADD CONSTRAINT ck_data_finalizare10 CHECK(data_finalizare>data_incepereii);
```

Table CREDITE_PROJECT altered.

12)Sa se adauge o restrictie de integritate pe coloanal status_credit din tabela rate_credit_PROJECT astfel incat statusul creditului sa nu fie NULL.

PROIECT – BAZE DE DATE – SIMON GEORGIANA

SELECT USER FROM DUAL;

ALTER TABLE rate_credit_bancare_PROIECT2

ADD CONSTRAINT ck_rata10 CHECK(status_rata IS NOT NULL);

--13) Sa se adauge o restrictie de integritate pe coloana status_credit din
--tabela rate_credit_bancare_PROIECT2 astfel incat statutul creditului sa nu fie NULL.
SELECT USER FROM DUAL;
ALTER TABLE rate_credit_bancare_PROIECT2
ADD CONSTRAINT ck_rata10 CHECK(status_rata IS NOT NULL);

Script Output | Query Result | Task completed in 0.181 seconds

Table AUTORIZARI_ONLINE_PROJECT altered.

Table INCASARI_PROJECT altered.

Table CREDITE_PROJECT altered.

Table RATE_CREDIT_BANCARE_PROJECT2 altered.

--13) Sa se adauge o restrictie de integritate pe coloan
--tabela rate_credit_bancare_PROJECT2 astfel incat stat
SELECT USER FROM DUAL;
ALTER TABLE rate_credit_bancare_PROJECT2
ADD CONSTRAINT ck_rata10 CHECK(status_rata IS NOT NULL)

Script Output | Query Result | All Rows Fetched: 1 in 0.087 seconds

USER
1 SIMONG_65

13)Sa se adauge o restrictie de integritate pe coloana comision din tabela transfer_catre_alte_banci_PROJECT astfel incat sa nu fie mai mare de 50%.

SELECT USER FROM DUAL;

ALTER TABLE transfer_catre_alte_banci_PROJECT

ADD CONSTRAINT ck_comision2 CHECK(comision<=60);

--14) Sa se adauge o restrictie de integritate pe coloana comision
--din tabela transfer_catre_alte_banci_PROJECT astfel incat sa nu fie mai mare de 60%.
SELECT USER FROM DUAL;
ALTER TABLE transfer_catre_alte_banci_PROJECT
ADD CONSTRAINT ck_comision2 CHECK(comision<=60);

Script Output | Query Result | Task completed in 0.146 seconds

Table TRANSFER_CATRE_ALTE_BANCI_PROJECT altered.

--14) Sa se adauge o restrictie de int
--din tabela transfer_catre_alte_banci_PROJECT astfel incat sa nu fie mai mare de 60%.
SELECT USER FROM DUAL;
ALTER TABLE transfer_catre_alte_banci_PROJECT
ADD CONSTRAINT ck_comision2 CHECK(comision<=60)

Script Output | Query Result | All Rows Fetched: 1 in 0.079 seconds

USER
1 SIMONG_65

14) Sa se redenumeasca tabela transfer_catre_alte_banci_PROJECT in transferuri_PROJECT.

SELECT USER FROM DUAL;

RENAME transfer_catre_alte_banci_PROJECT TO transferuri_PROJECT;

--15) Sa se redenumeasca tabela transfer_catre_alte_banci_PROJECT in transferuri_PROJECT.
SELECT USER FROM DUAL;
RENAME transfer_catre_alte_banci_PROJECT TO transferuri_PROJECT;

Script Output | Query Result | Task completed in 0.479 seconds

Table TRANSFER_CATRE_ALTE_BANCI_PROJECT altered.

Table renamed.

--15) Sa se redenumeasca tabela transfer_catre_alte_banci_PROJECT in transferuri_PROJECT.
SELECT USER FROM DUAL;
RENAME transfer_catre_alte_banci_PROJECT TO transferuri_PROJECT

Script Output | Query Result | All Rows Fetched: 1 in 0.158 seconds

USER
1 SIMONG_65

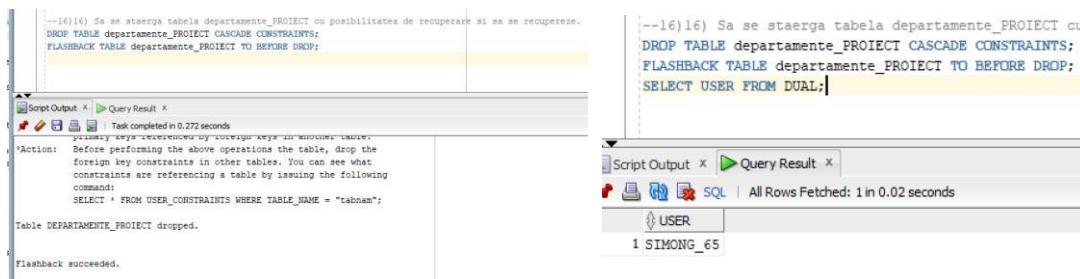
15) Sa se stearga tabela departamente_PROJECT cu posibilitatea de recuperare si sa se recupereze.

DROP TABLE departamente_PROJECT CASCADE CONSTRAINTS;

FLASHBACK TABLE departamente_PROJECT TO BEFORE DROP;

SELECT USER FROM DUAL;

PROIECT – BAZE DE DATE – SIMON GEORGIANA



```
--16) Se se sterga tabela departamente_PROJECT cu posibilitates de recuperare si sa se recuperese.  
DROP TABLE departamente_PROJECT CASCADE CONSTRAINTS;  
FLASHBACK TABLE departamente_PROJECT TO BEFORE DROP;  
  
--16) Se se sterga tabela departamente_PROJECT cu  
DROP TABLE departamente_PROJECT CASCADE CONSTRAINTS;  
FLASHBACK TABLE departamente_PROJECT TO BEFORE DROP;  
SELECT USER FROM DUAL;
```

Script Output | Query Result | Task completed in 0.272 seconds | Action: Primary key referenced by foreign keys in another table. Before performing the above operations the table, drop the foreign key constraints in other tables. You can see what constraints are referencing a table by issuing the following command:
SELECT * FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'tabnam';
Table DEPARTAMENTE_PROJECT dropped.
Flashback succeeded.

Script Output | Query Result | All Rows Fetched: 1 in 0.02 seconds | USER | 1 SIMONG_65

IV. INTRODUCEREA INREGISTRARILOR IN TABELE (minim 10 pentru fiecare)

1) Tabela clienti_PROIECT

```
SELECT *FROM clienti_PROIECT;  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1000, 'Simon', 'Georgiana', '0737420833', 'simonG@gmail', '6050325030000', 20);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1001, 'Ilinca', 'Lucian', '0723459190', 'ilincaL@gmail', '5020704030001', 23);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1002, 'Marinca', 'Ana', '0737420835', 'marincaA@gmail', '6001218030002', 25);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES  
(1003, 'Dumitru', 'Stefan', '0728420833', 'dumitruS@gmail', '5930201030003', 32);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1004, 'Marin', 'Ioana', '0717790837', 'marinI@yahoo', '6880517030004', 37);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1005, 'Radu', 'Mihai', '0738920831', 'raduM@yahoo', '5901014030005', 35);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1006, 'Stoica', 'Elena', '0728550987', 'stoicaE@yahoo', '6010925030006', 24);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1007, 'Vasiliu', 'Daniel', '0719805678', 'vasiliuD@gmail', '5840818030007', 41);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1008, 'Popa', 'Adriana', '0709512387', 'popaA@gmail', '6880725030008', 37);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES  
(1009, 'Preda', 'Andrei', '0737755109', 'predaA@yahoo', '5771103030009', 48);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1010, 'Dejanu', 'Nicolae', '0728820833', 'dejanuN@yahoo', '6730312030010', 52);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1011, 'Tudor', 'Alexandra', '0737420987', 'tudorA@gmail', '5041218030011', 21);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1012, 'Gheorghe', 'Malina', '0733020845', 'gheorgheM@gmail', '6640412030012', 61);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1013, 'Cojocaru', 'Florin', '0712567309', 'cojocaruF@yahoo', '5800118030013', 45);  
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)  
VALUES (1014, 'Dobre', 'Alexys', '0732421147', 'dobreA@gmail', '6931017030014', 32);
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1015, 'Ilie', 'Pavel', '0776420848', 'ilieP@yahoo', '5790512030015', 46);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1016, 'Enache', 'Cristina', '0733421849', 'enacheC@gmail', '6960811030016', 29); INSERT
INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta) VALUES
(1017, 'Nica', 'Ionel', '0707410850', 'nicaI@yahoo', '5620318030017',63 );
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1018, 'Simion', 'Lorena', '0732410850', 'simionL@gmail', '6910217030018', 34);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1019, 'Filip', 'Vlad', '0727420652', 'filipV@gmail', '5840912030019', 41);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1020, 'Mitu', 'Bianca', '0737420853', 'mituB@gmail', '6580411030020', 67);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1021, 'Mavrodin', 'Tiberiu', '0712440803', 'mavrodinT@gmail', '5860210030017', 39);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1022, 'Chiosa', 'Ciprian', '0721067893', 'chiosaL@yahoo', '5821026030080', 43);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1023, 'Popescu', 'Marcel', '0734123456', 'popescuM@gmail', '5891204030012', 36);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1024, 'Ionescu', 'Tania', '0745123456', 'ionescuT@yahoo', '6810314030098', 44);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1025, 'Marin', 'Ionut', '0756123456', 'marinI@hotmail', '5011204030011', 24);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1026, 'Dumitru', 'Camelia', '0767123456', 'dumitruC@gmail', '6061226030012', 19);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1027, 'Stana', 'Raul', '0778123456', 'stanaR@hotmail', '5850322030014', 40);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1028, 'Vasile', 'Gabriel', '0789123456', 'vasileG@hotmail', '5841104030065', 41);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1029, 'Neagu', 'Liliana', '0790123456', 'neaguL@hotmail', '6750320030014', 50);
INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta)
VALUES (1030, 'Ghita', 'Mihai', '0701123456', 'ghitaM@hotmail', '5050815030018', 20);
SELECT USER FROM DUAL;
SELECT *FROM CLIENTI_PROIECT;
COMMIT;
SELECT *FROM CLIENTI_PROIECT;
```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```

CREATE TABLE clienti_PROIECT (
    id_client NUMBER(10, 2),
    nume_client VARCHAR2(50),
    prenume_client VARCHAR2(50),
    telefon VARCHAR2(15),
    email VARCHAR2(100),
    cnp NUMBER(13, 0),
    varsta NUMBER(3, 0)
);

INSERT INTO clienti_PROIECT (id_client, nume_client, prenume_client, telefon, email, cnp, varsta) VALUES
(1024, 'Ionescu', 'Tudor', '073740833', 'tudor@yahoo', '65050325030001', 24),
(1025, 'Marin', 'Ionel', '0738123456', 'marin@hotmail', '50112040300011', 24),
(1026, 'Dumitru', 'Mihai', '073740835', 'mihai@yahoo', '65050325030002', 25),
(1027, 'Stana', 'Raul', '073740836', 'stana@hotmail', '50112040300014', 40),
(1028, 'Vasile', 'Gabriel', '073740837', 'vasile@yahoo', '65050325030002', 25),
(1029, 'Neagu', 'Liliana', '0790123456', 'neagu@hotmail', '6750320030014', 50);

SELECT * FROM clienti_PROIECT;

```

Script Output | Query Result | Task completed in 0.855 seconds
http://docs.oracle.com/error-help/db/crs-12899/
1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.

Script Output | Query Result | All Rows Fetched: 1 in 0.016 seconds
USER | SIMONG_65

```

SELECT USER FROM DUAL;
SELECT *FROM CLIENTI_PROIECT;
COMMIT;
SELECT *FROM CLIENTI_PROIECT;

```

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	TELEFON	EMAIL	CNP	VARSTA
1	1000 Simon	Georgiana	073740833	simon@gmail	65050325030000	20
2	1001 Ilie	Lucian	0723459190	ilie@yahoo	5020704030001	23
3	1002 Marin	Ana	0737420835	marina@yahoo	60012103030002	25
4	1003 Dumitru	Stefan	0728420833	dumitru@gmail	5530201030003	32
5	1004 Marin	Ioana	0717790837	marin@yahoo	6880517030004	37
6	1005 Radu	Mihai	0735920831	radu@yahoo	5501014030005	35
7	1006 Stoica	Elena	0725550987	stoica@yahoo	6010925030006	24
8	1007 Vasiliu	Daniel	0719805678	vasiliu@gmail	5840818030007	41
9	1008 Popa	Adriana	0709512387	popa@yahoo	6880725030008	37
10	1009 Preda	Andrei	0737755109	preda@yahoo	5771103030009	48
11	1010 Dejanu	Nicolae	0728820833	dejanu@yahoo	6730312030010	52
12	1011 Tudor	Alexandra	0737420897	tudor@yahoo	50412103030011	21
13	1012 Gheorghe	Malina	0733020845	gheorgheh@gmail	6440412030012	61
14	1013 Cojocaru	Florin	071567309	cojocaruf@yahoo	58001103030013	45
15	1014 Dobre	Alexys	0732421147	dobreA@gmail	6931017030014	32
16	1015 Ilie	Pavel	0776420848	ilieP@yahoo	5790512030015	46
17	1016 Enache	Cristina	0733421849	enacheC@gmail	6560811030016	29
18	1017 Nica	Ionel	0707410850	nicaI@yahoo	5620318030017	63
19	1018 Simion	Lorena	0732410850	simionL@gmail	6910217030018	34
20	1019 Filip	Vlad	0727420652	filipV@gmail	5840912030019	41
21	1020 Mitu	Bianca	0737420853	mituB@gmail	6580411030020	67

Obligatoriu, într-o tabelă trebuie să existe o înregistrare (rând) cu numele studentului. Se va prezenta o captură de ecran după interogarea care demonstrează acest lucru. În caz contrar, proiectul va fi notat cu 1p:

Demonstratie:

SELECT *FROM CLIENTI_PROIECT WHERE nume_client='Simon';

```

SELECT *FROM CLIENTI_PROIECT WHERE
nume_client='Simon';

```

Script Output | Query Result | All Rows Fetched: 1 in 0.061 seconds
ID_CLIENT | NUME_CLIENT | PRENUME_CLIENT | TELEFON | EMAIL | CNP | VARSTA
1 | 1000 Simon | Georgiana | 073740833 | simon@gmail | 65050325030000 | 20

2) Tabela conturi_PROIECT

DESCRIBE conturi_PROIECT;

ALTER TABLE conturi_PROIECT

MODIFY sold NUMBER(10, 2); --am modificat tipul, pt a putea avea un sold mai mare.

INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9000, 1000, 'Investitiile', 99999.99, 'RORAIF000000001000X9000');

INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9001, 1001, 'Economii', 85000.75, 'ROBBCR000000001001X9001');

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9002, 1002, 'Curent', 30000.00, 'ROBBRD0000000001002X9002');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9003, 1003, 'Investitii', 75000.00, 'ROBREV0000000001003X9003');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9004, 1004, 'Economii', 62000.25, 'ROBT00000000001004X9004');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9005, 1005, 'Curent', 8000.00, 'RORAIF000000001005X9005');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9006, 1006, 'Investitii', 92000.75, 'ROBBCR0000000001006X9006');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9007, 1007, 'Economii', 45000.00, 'ROBBRD0000000001007X9007');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9008, 1008, 'Curent', 25000.50, 'ROREV00000000001008X9008');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9009, 1009, 'Investitii', 10000.00, 'ROBT00000000001009X9009');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9010, 1010, 'Economii', 72000.75, 'RORAIF000000001010X9010');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9011, 1011, 'Curent', 15000.00, 'ROBBCR0000000001011X9011');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9012, 1012, 'Investitii', 45000.50, 'ROBBRD0000000001012X9012');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9013, 1013, 'Economii', 12000.75, 'ROREV00000000001013X9013');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9014, 1014, 'Curent', 5000.00, 'ROBT00000000001014X9014');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9015, 1015, 'Investitii', 29000.00, 'RORAIF0000000001015X9015');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9016, 1016, 'Economii', 14000.25, 'ROBBCR0000000001016X9016');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9017, 1017, 'Curent', 7500.50, 'ROBBRD0000000001017X9017');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9018, 1018, 'Investitii', 31000.75, 'ROREV00000000001018X9018');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9019, 1019, 'Economii', 17000.00, 'ROBT00000000001019X9019');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9020, 1020, 'Curent', 9500.00, 'RORAIF000000001020X9020');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9021, 1021, 'Curent', 7500.00, 'ROUNICR000000001021X9021');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9022, 1022, 'Economii', 1500.00, 'ROINGBK000000001022X9022');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES  
(9023, 1023, 'Investitii', 9500.00, 'ROALPHA000000001023X9023');  
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
(9024, 1024, 'Curent', 31000.00, 'ROTPBK000000001024X9024');
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9025, 1025, 'Economii', 32500.00, 'ROCECBK000000001025X9025');
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9026, 1026, 'Curent', 41000.00, 'ROUNICR000000001026X9026');
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9027, 1027, 'Curent', 95000.00, 'ROINGBK000000001027X9027');
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9028, 1028, 'Investitii', 78000.00, 'ROALPHA000000001028X9028');
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9029, 1029, 'Investitii', 99000.00, 'ROCECBK000000001029X9029');
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9030, 1030, 'Economii', 11000.00, 'ROCECBK000000001030X9030');
SELECT USER FROM DUAL;
SELECT *FROM conturi_PROIECT;
COMMIT;
```

The screenshot shows the Oracle SQL Developer interface with two main panes: 'Script Output' and 'Query Result'.

Script Output:

```
INSERT INTO conturi_proiect (id_cont, id_client, tip_cont, sold, iban) VALUES
(9030, 1030, 'Economii', 11000.00, 'ROCECBK000000001030X9030');
1 row inserted.

Commit complete.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```

Query Result - All Rows Fetched: 21 in 0.055 seconds:

ID_CONT	ID_CLIENT	TIPO_CONT	SOLD	IBAN
1	9000	1001 Investitii	99999.99	RORAIFF000000001000X9000
2	9001	1001 Economii	85000.75	ROBBCR000000001001X9001
3	9002	1002 Curent	30000	ROBBRD000000001002X9002
4	9003	1003 Investitii	75000	ROREV000000001003X9003
5	9004	1004 Economii	62000.25	ROBT00000000001004X9004
6	9005	1005 Curent	8000	RORAIFF000000001005X9005
7	9006	1006 Investitii	92000.75	ROBBCR000000001006X9006
8	9007	1007 Economii	45000	ROBBRD000000001007X9007
9	9008	1008 Curent	25000.5	ROREV000000001008X9008
10	9009	1009 Investitii	10000	ROBT00000000001009X9009
11	9010	1010 Economii	72000.75	RORAIFF000000001010X9010
12	9011	1011 Curent	15000	ROBBCR000000001011X9011
13	9012	1012 Investitii	45000.5	ROBBRD000000001012X9012
14	9013	1013 Economii	12000.75	ROREV000000001013X9013
15	9014	1014 Curent	5000	ROBT00000000001014X9014
16	9015	1015 Investitii	29000	RORAIFF000000001015X9015
17	9016	1016 Economii	14000.25	ROBBCR000000001016X9016
18	9017	1017 Curent	7500.5	ROBBRD000000001017X9017
19	9018	1018 Investitii	31000.75	ROREV000000001018X9018
20	9019	1019 Economii	17000	ROCECBK000000001019X9019
21	9020	1020 Curent	9500	RORAIFF000000001020X9020
22	9021	1021 Curent	7500	ROUNICR000000001021X9021
23	9022	1022 Economii	1500	ROINGBK000000001022X9022
24	9023	1023 Investitii	9500	ROALPHA000000001023X9023
25	9024	1024 Curent	31000	ROOTPBK000000001024X9024
26	9025	1025 Economii	32500	ROCECBK000000001025X9025
27	9026	1026 Curent	41000	ROUNICR000000001026X9026
28	9027	1027 Curent	95000	ROINGBK000000001027X9027
29	9028	1028 Investitii	78000	ROALPHA000000001028X9028
30	9029	1029 Investitii	99000	ROCECBK000000001029X9029
31	9030	1030 Economii	11000	ROCECBK000000001030X9030

3)TABELA carduri_PROIECT

```
DESCRIBE carduri_PROIECT;
```

```
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1100, 'Debit', TO_DATE('2027-12-31', 'YYYY-MM-DD'), 123, 9876, '1234567812345678',
9000);
```

```
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1101, 'Credit', TO_DATE('2026-11-30', 'YYYY-MM-DD'), 234, 1234, '2345678923456789',
9001);
```

```
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1102, 'Debit', TO_DATE('2025-05-15', 'YYYY-MM-DD'), 345, 5678, '3456789034567890',
9002);
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1103, 'Credit', TO_DATE('2028-09-30', 'YYYY-MM-DD'), 456, 4321, '4567890145678901', 9003);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1104, 'Debit', TO_DATE('2027-01-01', 'YYYY-MM-DD'), 567, 8765, '5678901256789012', 9004);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1105, 'Credit', TO_DATE('2029-03-21', 'YYYY-MM-DD'), 678, 6543, '6789012367890123', 9005);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1106, 'Debit', TO_DATE('2026-06-30', 'YYYY-MM-DD'), 789, 1122, '7890123478901234', 9006);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1107, 'Credit', TO_DATE('2027-02-28', 'YYYY-MM-DD'), 890, 2211, '8901234589012345', 9007);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1108, 'Debit', TO_DATE('2025-12-31', 'YYYY-MM-DD'), 901, 3344, '9012345690123456', 9008);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1109, 'Credit', TO_DATE('2026-07-15', 'YYYY-MM-DD'), 234, 4433, '2345678923456789', 9009);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1110, 'Debit', TO_DATE('2025-11-30', 'YYYY-MM-DD'), 123, 5566, '1234567812345678', 9010);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1111, 'Credit', TO_DATE('2027-08-20', 'YYYY-MM-DD'), 345, 6677, '3456789034567890', 9011);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1112, 'Debit', TO_DATE('2028-01-01', 'YYYY-MM-DD'), 456, 7788, '4567890145678901', 9012);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1113, 'Credit', TO_DATE('2025-05-10', 'YYYY-MM-DD'), 567, 8899, '5678901256789012', 9013);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1114, 'Debit', TO_DATE('2026-10-31', 'YYYY-MM-DD'), 678, 9900, '6789012367890123', 9014);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1115, 'Credit', TO_DATE('2027-02-20', 'YYYY-MM-DD'), 789, 1122, '7890123478901234', 9015);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1116, 'Debit', TO_DATE('2025-09-30', 'YYYY-MM-DD'), 890, 2233, '8901234589012345', 9016);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1117, 'Credit', TO_DATE('2026-04-25', 'YYYY-MM-DD'), 234, 3344, '2345678923456789', 9017);
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1118, 'Debit', TO_DATE('2027-03-15', 'YYYY-MM-DD'), 345, 4455, '3456789034567890', 9018);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1119, 'Credit', TO_DATE('2025-08-10', 'YYYY-MM-DD'), 456, 5566, '4567890145678901', 9019);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1120, 'Debit', TO_DATE('2028-05-30', 'YYYY-MM-DD'), 567, 6677, '5678901256789012', 9020);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1121, 'Debit', TO_DATE('2029-06-30', 'YYYY-MM-DD'), 331, 9331, '6678901256239011', 9021);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1122, 'Debit', TO_DATE('2027-04-20', 'YYYY-MM-DD'), 617, 6997, '1672380356789012', 9022);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1123, 'Credit', TO_DATE('2028-01-06', 'YYYY-MM-DD'), 207, 1003, '3692540256789012', 9023);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1124, 'Debit', TO_DATE('2028-02-06', 'YYYY-MM-DD'), 305, 1402, '3012540256895432', 9024);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1125, 'Credit', TO_DATE('2029-03-06', 'YYYY-MM-DD'), 407, 5231, '2292540256843921', 9025);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1126, 'Debit', TO_DATE('2030-04-06', 'YYYY-MM-DD'), 589, 2145, '3692540256758124', 9026);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1127, 'Credit', TO_DATE('2098-05-06', 'YYYY-MM-DD'), 614, 7539, '1982540256714928', 9027);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1128, 'Debit', TO_DATE('2030-06-06', 'YYYY-MM-DD'), 782, 6321, '2292540256794836', 9028);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1129, 'Credit', TO_DATE('2027-07-06', 'YYYY-MM-DD'), 893, 4157, '4552540256903475', 9029);
INSERT INTO CARDURI_PROIECT (id_card, tip_card, data_expirare, cvv, pin, numar_card, id_cont)
VALUES (1130, 'Debit', TO_DATE('2026-08-06', 'YYYY-MM-DD'), 234, 8765, '9222540256885392', 9030);
SELECT USER FROM DUAL;
SELECT *FROM carduri_PROIECT;
COMMIT;
```


PROIECT – BAZE DE DATE – SIMON GEORGIANA

(20, 'Consiliul de administratie');

```
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(21, 'Consultanta financiara');
```

```
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(22, 'Call center');
```

```
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(23, 'Operatiuni leasing');
```

```
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES (24,
'Resurse umane');
```

```

INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(20, 'Consiliul de administratie');
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(21, 'Consultanta financiara');
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(22, 'Call center');
INSERT INTO departamente_PROIECT (id_departamente, denumire_departament) VALUES
(23, 'Operatiuni leasing');

-- Verify user exists in DUAL
SELECT USER FROM DUAL;

```

Script Output X | All Rows Fetched: 1 in 0.037 seconds

USER
SIMONG_65


```

(24, 'Resurse umane');
SELECT USER FROM DUAL;
SELECT *FROM departamente_PROIECT;

```

Script Output X | All Rows Fetched: 16 in 0.046 seconds

ID_DEPARTAMENTE	DENUMIRE_DEPARTAMENT
1	10 Relatii cu clientii persoane fizice
2	11 Credite/Imprumuturi
3	12 IT si suport tehnic
4	13 Marketing
5	14 Schimb valutar
6	15 Evaluatori imobiliari
7	16 Relatii cu clientii persoana juridice
8	17 Control intern
9	18 Servicii de achiziere si documentatie
10	19 Planificare strategica si dezvoltare
11	20 Consiliul de administratie
12	21 Consultante financiare
13	22 Call center
14	23 Operatiuni leasing
15	24 Resurse umane
16	25 Departament juridic

5) Tabela banca_PROIECT

Aici am modificat tabela banca_PROIECT, deoarece mi-am dat seama ca nu am ales corect coloanele si am adaugat tabela banca_clienti_PROIECT pentru a face legatura corecta intre banca_PROIECT si client_PROIECT si pentru a evidenta legatura one to many. Mai multe persoane sunt clienti la aceeasi banca. Am pastrat in tabela banca_PROIECT date despre banci, la general.

```
ALTER TABLE banca_PROIECT
```

```
DROP COLUMN id_client;
```

```
ALTER TABLE banca_PROIECT
```

```
DROP COLUMN id_angajat;
```

```
COMMIT;
```

```
DESCRIBE banca_PROIECT;
```

```
Table BANCA_PROIECT altered.
```

```
Commit complete.
```

Name	Null?	Type
ID_BANCA	NOT NULL	NUMBER(4)
DENUMIRE	NOT NULL	VARCHAR2(30)
ADRESA		VARCHAR2(20)
AN_FONDARE		NUMBER(4)

PROIECT – BAZE DE DATE – SIMON GEORGIANA

Crearea unei tabele intermediare:

```
CREATE TABLE banca_clienti_PROIECT(
    id_banca NUMBER(4),
    CONSTRAINT fk_id_banca10 FOREIGN KEY (id_banca) REFERENCES banca_PROIECT(id_banca),
    id_client NUMBER(6),
    CONSTRAINT fk_id_client60 FOREIGN KEY (id_client) REFERENCES clienti_PROIECT(id_client));
```

```
AN_FONDARE          NUMBER(4)
Table BANCA_CLIENTI_PROIECT created.
```

Tot in acest moment, am realizat ca ar fi bine sa modific si tabela angajati_PROIECT:

```
DESCRIBE angajati_PROIECT;
ALTER TABLE angajati_PROIECT
DROP COLUMN id_client;
ALTER TABLE angajati_PROIECT
ADD id_banca NUMBER(4);
ALTER TABLE angajati_PROIECT
ADD CONSTRAINT fk_banca FOREIGN KEY (id_banca) REFERENCES
banca_PROIECT (ID_BANCA);
```

```
Table ANGAJATI_PROJECT altered.
Table ANGAJATI_PROJECT altered.
Table ANGAJATI_PROJECT altered.
```

```
Worksheet - Query Builder
ALTER TABLE angajati_PROJECT
ADD id_banca NUMBER(4);
ALTER TABLE angajati_PROJECT
ADD CONSTRAINT fk_banca FOREIGN KEY (id_banca)
REFERENCES banca_PROJECT (ID_BANCA);
COMMIT;
DESCRIBE angajati_PROJECT;

ALTER TABLE banca_PROJECT
DROP COLUMN id_client;
ALTER TABLE banca_PROJECT
DROP COLUMN id_angajat;
COMMIT;
DESCRIBE banca_PROJECT;

CREATE TABLE banca_clienti_PROJECT(
    id_banca NUMBER(4),
    CONSTRAINT fk_id_banca10 FOREIGN KEY (id_banca) REFERENCES banca_PROJECT(id_banca),
    id_client NUMBER(6),
    CONSTRAINT fk_id_client60 FOREIGN KEY (id_client) REFERENCES clienti_PROJECT(id_client));
SELECT USER FROM DUAL;
```

```
Script Output * | Query Result *
SQL | All Rows Fetched: 1 in 0.019 seconds
USER
1 SECOND 45

INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3000, 'Raiffeisen Bank', 'Bucuresti', 2002);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3001, 'Revolut', 'Bucuresti', 2015);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3002, 'BCR', 'Bucuresti', 1990);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3003, 'BRD', 'Bucuresti', 1923);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3004, 'Banca Transilvania', 'Cluj-Napoca', 1993);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3005, 'OTP Bank', 'Bucuresti', 1949);
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3006, 'CEC Bank', 'Bucuresti', 1864);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3007, 'Alpha Bank', 'Bucuresti', 1994);
INSERT INTO banca_PROIECT (id_banca, denumire, adresa, an_fondare)
VALUES (3008, 'Unicredit', 'Bucuresti', 2005);
SELECT *FROM banca_PROIECT;
SELECT USER FROM DUAL;
COMMIT;

```

The screenshot shows the execution of an SQL script in SSMS. The script inserts four rows into the 'banca_PROIECT' table and then selects all rows from it. Finally, it prints the current user and commits the transaction.

```

-- Script Output
INSERT INTO banca_PROJECT (id_banca, denumire, adresa, an_fondare)
VALUES (3006, 'CEC Bank', 'Bucuresti', 1864);
INSERT INTO banca_PROJECT (id_banca, denumire, adresa, an_fondare)
VALUES (3007, 'Alpha Bank', 'Bucuresti', 1994);
INSERT INTO banca_PROJECT (id_banca, denumire, adresa, an_fondare)
VALUES (3008, 'Unicredit', 'Bucuresti', 2005);

-- Query Result
SELECT *FROM banca_PROJECT;

-- Script Output
1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

-- Query Result
ID_BANCA | DENUMIRE | ADRESA | AN_FONDARE
1         | 3000 Raiffeisen Bank | Bucuresti | 2002
2         | 3001 Revolut | Bucuresti | 2015
3         | 3002 BCR | Bucuresti | 1990
4         | 3003 BRD | Bucuresti | 1923
5         | 3004 Banca Transilvania Cluj-Napoca | Bucuresti | 1993
6         | 3005 OTP Bank | Bucuresti | 1949
7         | 3006 CEC Bank | Bucuresti | 1864
8         | 3007 Alpha Bank | Bucuresti | 1994
9         | 3008 Unicredit | Bucuresti | 2005

-- Script Output
SELECT *FROM banca_PROJECT;
SELECT USER FROM DUAL;

-- Query Result
USER
SIMONG_65

```

6) Tabel banca_clienti_PROIECT

```

INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1000, 3000); -- Raiffeisen
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1001, 3002); -- BCR
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1002, 3003); -- BRD
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1003, 3001); -- Revolut
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1004, 3004); -- BT
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1005, 3000); -- Raiffeisen
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1006, 3002); -- BCR
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1007, 3003); -- BRD
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1008, 3001); -- Revolut
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1009, 3004); -- BT
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1010, 3000); -- Raiffeisen
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1011, 3002); -- BCR
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1012, 3003); -- BRD
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1013, 3001); -- Revolut
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1014, 3004); -- BT
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1015, 3000); -- Raiffeisen
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1016, 3002); -- BCR
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1017, 3003); -- BRD
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1018, 3001); -- Revolut
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1019, 3004); -- BT
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1020, 3000); -- Raiffeisen

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1021, 3008); -- UniCredit
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1022, 3009); -- ING
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1023, 3007); -- Alpha
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1024, 3005); -- OTP
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1025, 3006); -- CEC
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1026, 3008); -- UniCredit
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1027, 3009); -- ING
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1028, 3007); -- Alpha
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1029, 3006); -- CEC
INSERT INTO banca_clienti_PROIECT (id_client, id_banca) VALUES (1030, 3006); -- CEC
SELECT *FROM banca_clienti_PROIECT;
SELECT USER FROM DUAL;
COMMIT;

```

The screenshot shows the execution of a SQL script in SSMS. The script inserts 10 rows into the 'banca_clienti_PROIECT' table and then performs a SELECT * operation. It also includes a COMMIT statement and a SELECT USER FROM DUAL; query.

Script Output:

```

1 row inserted.

```

Query Result:

ID_BANCA	ID_CLIENT	
4	3001	1003
5	3004	1004
6	3000	1005
7	3002	1006
8	3003	1007
9	3001	1008
10	3004	1009
11	3000	1010
12	3002	1011
13	3003	1012
14	3001	1013
15	3004	1014
16	3000	1015
17	3002	1016
18	3003	1017
19	3001	1018
20	3004	1019

Results of SELECT * FROM banca_clienti_PROIECT;

Results of SELECT USER FROM DUAL;

USER
SIMONG_65

7) Tabela angajati_PROIECT

```

DESCRIBE angajati_PROIECT;
ALTER TABLE angajati_PROIECT

```

```

MODIFY salariul NUMBER(6, 2); --am constatat ca nu am ales bine precizia si scala.

```

1) PT BANCA RAIFFEISEN:

```

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2000, 'Ciobanu', 'Marian', 42, 8500.00, NULL, 20,
3000);

```

```

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2001, 'Ene', 'Mihaela', 33, 4500.00, 2000, 15, 3000);

```

```

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2002, 'Gavrila', 'Madalina', 37, 5000.00, 2000, 16,
3000);

```

```

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2003, 'Lungu', 'Mihai', 23, 2350.00, 2000, 22, 3000);

```

```

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2004, 'Oprea', 'Marius', 30, 3000.00, 2000, 23, 3000);

```

```

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2005, 'Popescu', 'Andrei', 28, 2000.00, 2000, 24, 3000);

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

id_manager, id_departamente, id_banca) VALUES (2005, 'Popa', 'Adriana', 29, 4000.00, 2001, 15, 3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2006, 'Vasilescu', 'George', 34, 4300.00, 2005, 15,
3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2007, 'Marinescu', 'Ioana', 28, 4200.00, 2002, 16,
3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2008, 'Grigorescu', 'Dan', 31, 4700.00, 2007, 16,
3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2009, 'Ionescu', 'Radu', 26, 3100.00, 2003, 22, 3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2010, 'Georgescu', 'Ana', 24, 2900.00, 2003, 22,
3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2011, 'Stan', 'Mircea', 35, 4500.00, 2004, 23, 3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2012, 'Gavrilă', 'Mihaila', 33, 4500, 2001, 23, 3000);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2013, 'Petrescu', 'Cătălin', 27, 3700.00, 2011, 23, 3000);

```

```

SELECT USER FROM DUAL;
SELECT *FROM angajati_PROIECT;
COMMIT;

```

The screenshot shows three tabs in the SSMS interface:

- Script Output**: Shows the execution of an INSERT query for user 'SIMONG_65'.
- Query Result**: Shows the result of the 'SELECT USER FROM DUAL;' query, returning 'SIMONG_65'.
- Script Output**: Shows the execution of a SELECT * query for the 'angajati_PROJECT' table, returning 14 rows of data.

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARIUl	ID_MANAGER	ID_DEPARTAMENTE	ID_BANCA
1	2000 Ciobanu	Marian	42	8500	(null)	20	3000
2	2001 Enă	Mihaela	33	4500	2000	15	3000
3	2002 Gavrila	Madalina	37	5000	2000	16	3000
4	2003 Lungu	Mihai	23	2350	2000	22	3000
5	2004 Oprea	Marius	30	3000	2000	23	3000
6	2005 Popa	Adriana	29	4000	2001	15	3000
7	2006 Vasilescu	George	34	4300	2005	15	3000
8	2007 Marinescu	Ioana	28	4200	2002	16	3000
9	2008 Grigorescu	Dan	31	4700	2007	16	3000
10	2009 Ionescu	Radu	26	3100	2003	22	3000
11	2010 Georgescu	Ana	24	2900	2003	22	3000
12	2011 Stan	Mircea	35	4500	2004	23	3000
13	2012 Dumitrescu	Laura	32	4100	2011	23	3000
14	2013 Petrescu	Cătălin	27	3700	2011	23	3000

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

2)PT BANCA REVOLUT:

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2014, 'Toma', 'Gabriela', 46, 7600.00, NULL, 20,
3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2015, 'Balan', 'Roberta', 39, 6500.00, 2014, 12, 3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2016, 'Cristescu', 'Ioana', 26, 4850.00, 2014, 14,
3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2017, 'Stanciu', 'Dumitru', 30, 5800.00, 2014, 13,
3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2018, 'Popescu', 'Mihai', 34, 6200.00, 2015, 12,
3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2019, 'Dinu', 'Alina', 28, 5200.00, 2018, 12, 3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2020, 'Vlad', 'Radu', 31, 6000.00, 2017, 13, 3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2021, 'Ion', 'Andreea', 27, 4750.00, 2020, 13, 3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2022, 'Enache', 'Vlad', 40, 5400.00, 2016, 14, 3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2023, 'Morar', 'Iulia', 35, 5000.00, 2022, 14, 3001);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,
id_manager, id_departamente, id_banca) VALUES (2024, 'Rusu', 'George', 32, 4800.00, 2022, 14, 3001);
SELECT USER FROM DUAL;
SELECT *FROM angajati_PROIECT;
COMMIT;
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

The screenshot shows several windows from SQL Server Management Studio:

- Query Result:** Shows two INSERT statements into the 'angajati_PROJECT' table:


```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, (2023, 'Mihai', 'Iulia', 35, 6200.00, 2023, 14, 3001);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, (2024, 'Raluca', 'George', 32, 4800.00, 2022, 14, 3002);
```
- Script Output:** Shows a SELECT statement from DUAL:


```
SELECT USER FROM DUAL;
```

 Result: SIMONG_65
- Table Data:** Shows the contents of the 'angajati_PROJECT' table:

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARIU	ID_MANAGER	ID_DEPARTAMENTE	ID_BANCA
1	Mihai	Iulia	35	6200.00	2023	14	3001
2	Raluca	George	32	4800.00	2024	14	3002
3	Andrei	Vasile	40	7000.00	2025	19	3003
4	Andreea	Barbu	36	5500.00	2026	12	3004
5	Daniel	Mihai	34	4200.00	2027	13	3005
6	Elena	Stoica	38	5200.00	2028	12	3006
7	Eugen	Tudor	39	6000.00	2029	13	3007
8	Felicia	Andreea	37	4750.00	2030	14	3008
9	Gabriel	Andrea	35	5000.00	2031	14	3009
10	George	Marin	33	4500.00	2032	13	3010
11	Horia	Alina	30	5800.00	2033	13	3011
12	Iulia	Radu	32	4100.00	2034	12	3012
13	Ion	Andrea	31	4750.00	2035	13	3013
14	Ionut	Andrei	38	5500.00	2036	12	3014
15	Iulian	Andrea	39	6400.00	2037	13	3015
16	Iulian	Andrea	37	5200.00	2038	12	3016
17	Iulian	Andrea	36	5000.00	2039	13	3017
18	Iulian	Andrea	35	4900.00	2040	14	3018
19	Iulian	Andrea	34	4800.00	2041	13	3019
20	Iulian	Andrea	33	4750.00	2042	12	3020
21	Iulian	Andrea	32	4700.00	2043	13	3021
22	Iulian	Andrea	31	4650.00	2044	12	3022
23	Iulian	Andrea	30	4600.00	2045	13	3023
24	Iulian	Andrea	29	4550.00	2046	12	3024
25	Iulian	Andrea	28	4500.00	2047	13	3025
26	Iulian	Andrea	27	4450.00	2048	12	3026
27	Iulian	Andrea	26	4400.00	2049	13	3027
28	Iulian	Andrea	25	4350.00	2050	12	3028
29	Iulian	Andrea	24	4300.00	2051	13	3029
30	Iulian	Andrea	23	4250.00	2052	12	3030
31	Iulian	Andrea	22	4200.00	2053	13	3031
32	Iulian	Andrea	21	4150.00	2054	12	3032
33	Iulian	Andrea	20	4100.00	2055	13	3033
34	Iulian	Andrea	19	4050.00	2056	12	3034
35	Iulian	Andrea	18	4000.00	2057	13	3035
36	Iulian	Andrea	17	3950.00	2058	12	3036
37	Iulian	Andrea	16	3900.00	2059	13	3037
38	Iulian	Andrea	15	3850.00	2060	12	3038
39	Iulian	Andrea	14	3800.00	2061	13	3039
40	Iulian	Andrea	13	3750.00	2062	12	3040
41	Iulian	Andrea	12	3700.00	2063	13	3041
42	Iulian	Andrea	11	3650.00	2064	12	3042
43	Iulian	Andrea	10	3600.00	2065	13	3043
44	Iulian	Andrea	9	3550.00	2066	12	3044
45	Iulian	Andrea	8	3500.00	2067	13	3045
46	Iulian	Andrea	7	3450.00	2068	12	3046
47	Iulian	Andrea	6	3400.00	2069	13	3047
48	Iulian	Andrea	5	3350.00	2070	12	3048
49	Iulian	Andrea	4	3300.00	2071	13	3049
50	Iulian	Andrea	3	3250.00	2072	12	3050
51	Iulian	Andrea	2	3200.00	2073	13	3051
52	Iulian	Andrea	1	3150.00	2074	12	3052
53	Iulian	Andrea	0	3100.00	2075	13	3053

3)BANCA BCR

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2025, 'Sandu', 'Alexandru', 45, 7800.00, NULL, 20, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2026, 'Matei', 'Iulia', 38, 6400.00, 2025, 10, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2027, 'Voinea', 'Cristian', 29, 5100.00, 2026, 10, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2028, 'Marin', 'Roxana', 34, 6200.00, 2025, 11, 3002);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2029, 'Dragomir', 'Gabriel', 30, 4900.00, 2028, 11, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2030, 'Barbu', 'Vasile', 40, 7000.00, 2025, 19, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2031, 'Stoica', 'Elena', 36, 5500.00, 2030, 19, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2032, 'Tudor', 'Andrei', 28, 4800.00, 2030, 19, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2033, 'Gheorghe', 'Maria', 32, 4600.00, 2031, 19, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2034, 'Pavel', 'Daniel', 27, 4500.00, 2032, 19, 3002);
```

```
SELECT USER FROM DUAL;
```

```
SELECT *FROM angajati_PROJECT;
```

```
COMMIT;
```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

The screenshot shows three separate query windows in Oracle SQL Developer:

- Query 1 (Left):** Inserts four rows into the `angajati_PROJECT` table.
- Query 2 (Top Right):** Inserts two rows into the `angajati_PROJECT` table.
- Query 3 (Bottom Right):** Selects the user from the `DUAL` table, returning the value `SIMONG_65`.
- Query 4 (Bottom Left):** Selects all columns from the `angajati_PROJECT` table, resulting in 35 rows of data.

4) BANCA BRD

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2032, 'Tudor', 'Andrei', 28, 4800.00, 2030, 19, 3002);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2033, 'Gheorghe', 'Maria', 32, 4600.00, 2031, 18, 3003);
```

```
SELECT USER FROM DUAL;
```

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARIU	ID_MANAGER	ID_DEPARTAMENTE	ID_BANCA
22	2021Ion	Andreea	27	4750	2020	13	3001
23	2022Enache	Vlad	40	5400	2016	14	3001
24	2023Morar	Iulia	35	5000	2022	14	3001
25	2024Rusu	George	32	4800	2022	14	3001
26	2025Sandu	Alexandru	45	7800	(null)	20	3002
27	2026Matei	Iulia	38	6400	2025	10	3002
28	2027Voinea	Cristian	29	5100	2026	10	3002
29	2028Marin	Roxana	34	6200	2025	11	3002
30	2029Dragomir	Gabriel	30	4900	2028	11	3002
31	2030Barbu	Vasile	40	7000	2025	19	3002
32	2031Stoica	Elena	36	5500	2030	19	3002
33	2032Tudor	Andrei	28	4800	2030	19	3002
34	2033Gheorghe	Maria	32	4600	2031	19	3002
35	2034Pavel	Daniel	27	4500	2032	19	3002

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2035, 'Ilie', 'Alexandra', 47, 8000.00, NULL, 20, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2036, 'Moldovan', 'Andrei', 39, 6700.00, 2035, 13, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2037, 'Ciucă', 'Ioana', 29, 5400.00, 2036, 13, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2038, 'Anghel', 'Mihai', 35, 6200.00, 2035, 17, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2039, 'Dumitru', 'Gabriela', 28, 5100.00, 2038, 17, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2040, 'Petrescu', 'Radu', 33, 5900.00, 2035, 18, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2041, 'Iacob', 'Simona', 37, 5600.00, 2040, 18, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2042, 'Rusu', 'Florin', 30, 4800.00, 2041, 18, 3003);
```

```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2043, 'Neagu', 'Elena', 27, 4700.00, 2041, 18, 3003);
```

```
SELECT USER FROM DUAL;
```

```
SELECT *FROM angajati_PROJECT;
```

```
COMMIT;
```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

The screenshot shows several tabs in the Oracle SQL Developer interface:

- Script Output X**: Contains the following SQL code:


```
(2041, 'Iacob', 'Diana', 37, 5600.00, 2040, 18, 3003);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca)
VALUES (2042, 'Rusu', 'Florin', 30, 4800.00, 2041, 15, 3003);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca)
VALUES (2043, 'Neagu', 'Elena', 27, 4700.00, 2041, 18, 3003);
```
- Query Result X**: Shows the result of the last query: "1 row inserted."
- Script Output X**: Contains the following SQL code:


```
(2041, 'Iacob', 'Simona', 37, 5600.00, 2040, 18,
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca)
VALUES (2042, 'Rusu', 'Florin', 30, 4800.00, 2041, 15,
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca)
VALUES (2043, 'Neagu', 'Elena', 27, 4700.00, 2041, 18,
```
- Query Result X**: Shows the result of the last query: "1 row inserted."
- Script Output X**: Contains the following SQL code:


```
SELECT USER FROM DUAL;
```
- Query Result X**: Shows the result: "SIMONG_65"
- Script Output X**: Contains the following SQL code:


```
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca)
VALUES (2044, 'Neagu', 'Tudor', 30, 4800.00, 2040, 18, 3003);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca)
VALUES (2045, 'Neagu', 'Elena', 27, 4700.00, 2041, 18, 3003);

SELECT * FROM DUAL;
```
- Query Result X**: Shows the result: "All Rows Fetched: 1 in 0.015 seconds"

5) BANCA BT

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2044, 'Lazar', 'Ion', 50, 8500.00, NULL, 20, 3004);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2045, 'Manole', 'Daniela', 42, 7100.00, 2044, 21, 3004);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2046, 'Radu', 'Victor', 35, 6200.00, 2045, 21, 3004);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2047, 'Voicu', 'Alexandru', 29, 5100.00, 2046, 21, 3004);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2048, 'Popa', 'Andreea', 38, 7200.00, 2044, 24, 3004);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2049, 'Teodorescu', 'Gabriel', 31, 5500.00, 2048, 24, 3004);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2050, 'Vasile', 'Mihai', 28, 4900.00, 2049, 24, 3004);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2051, 'Enescu', 'Florin', 33, 5200.00, 2049, 24, 3004);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2052, 'Ciobanu', 'Roxana', 40, 7500.00, 2044, 25, 3004);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2053, 'Ionescu', 'Alin', 36, 6000.00, 2052, 25, 3004);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departament, id_banca) VALUES (2054, 'Barbu', 'Elena', 32, 5200.00, 2052, 25, 3004);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul,

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
id_manager, id_departamente, id_banca) VALUES (2055, 'Gheorghiu', 'George', 29, 4800.00, 2053, 25, 3004);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2056, 'Constantin', 'Irina', 27, 4700.00, 2054, 25, 3004);
```

```
SELECT USER FROM DUAL;
```

```
SELECT *FROM angajati_PROIECT;
```

```
COMMIT;
```

The screenshot shows the Oracle SQL Developer interface with three tabs: Script Output, Query Result, and SQL.

- Script Output:** Shows the execution of an INSERT script with 6 rows inserted. The script includes:


```
(2054, 'Babu', 'Elena', 30, 5200.00, 2052, 25, 3004);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca)
VALUES (2055, 'Gheorghiu', 'George', 29, 4800.00, 2053, 25, 3004);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca)
VALUES (2056, 'Constantin', 'Irina', 27, 4700.00, 2054, 25, 3004);
```
- Query Result:** Shows the result of the COMMIT command, which returns 1 row inserted.
- SQL:** Shows the SELECT USER FROM DUAL command, which returns 'SIMONG_65'.

6) BANCA OTP

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2057, 'Marin', 'Ioan', 49, 8600.00, NULL, 20, 3005);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2058, 'Dobre', 'Claudia', 41, 7300.00, 2057, 23, 3005);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2059, 'Petre', 'Adrian', 36, 6100.00, 2058, 23, 3005);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2060, 'Stoica', 'Maria', 32, 5800.00, 2059, 23, 3005);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2061, 'Neacsu', 'Ion', 44, 7500.00, 2057, 11, 3005);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2062, 'Rusu', 'Mirela', 30, 5400.00, 2061, 11, 3005);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2063, 'Popescu', 'Andrei', 29, 4900.00, 2062, 11, 3005);
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2064, 'Radulescu', 'Cristina', 39, 6900.00, 2057, 17, 3005);

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2065, 'Ciobotaru', 'Marius', 35, 6200.00, 2064, 17, 3005);

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2066, 'Voinea', 'Elena', 31, 5600.00, 2064, 17, 3005);
 INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2067, 'Iordache', 'Vlad', 27, 4700.00, 2065, 17, 3005);
 INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2068, 'Mazilu', 'Gabriela', 28, 4800.00, 2066, 17, 3005);

SELECT USER FROM DUAL;

SELECT *FROM angajati_PROIECT;

COMMIT;

The screenshot shows the Oracle SQL Developer interface with several tabs open:

- Script Output X | Query Result X**: Shows the execution of four INSERT statements for the angajati_PROIECT table, each returning a success message ("1 row inserted.") and a task completion time.
- Query Result X**: Shows the result of the COMMIT command.
- Script Output X | Query Result X**: Shows the execution of the SELECT USER FROM DUAL command, returning "SIMONG_65".
- Query Result X**: Shows the result of the SELECT *FROM angajati_PROIECT command, displaying a table with 69 rows of employee data.

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARIU	ID_MANAGER	ID_DEPARTAMENTE	ID_BANCA
2051	Ionescu	Fictiv	33	5200	2049	24	3004
53	2052 Ciobanu	Roxana	40	7500	2044	25	3004
54	2053 Ionescu	Alin	36	6000	2052	25	3004
55	2054 Barbu	Elena	32	5200	2052	25	3004
56	2055 Gheorghiu	George	29	4800	2053	25	3004
57	2056 Constantin	Irina	27	4700	2054	25	3004
58	2057 Marin	Ioan	49	8600	(null)	20	3005
59	2058 Dobre	Claudia	41	7300	2057	23	3005
60	2059 Petre	Adrian	36	6100	2058	23	3005
61	2060 Stoica	Maria	32	5800	2059	23	3005
62	2061 Neacsu	Ion	44	7500	2057	11	3005
63	2062 Rusu	Mirela	30	5400	2061	11	3005
64	2063 Popescu	Andrei	29	4900	2062	11	3005
65	2064 Radulescu	Cristina	39	6900	2057	17	3005
66	2065 Ciobotaru	Marius	35	6200	2064	17	3005
67	2066 Voinea	Elena	31	5600	2064	17	3005
68	2067 Iordache	Vlad	27	4700	2065	17	3005
69	2068 Mazilu	Gabriela	28	4800	2066	17	3005

7)BANCA CEC

INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2069, 'Tudor', 'Ioana', 48, 8700.00, NULL, 20, 3006);
 INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2070, 'Matei', 'Andreea', 40, 7400.00, 2069, 12, 3006);

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2071, 'Dragomir', 'Mihai', 37, 6300.00, 2070, 12, 3006);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2072, 'Olaru', 'Cristina', 33, 5700.00, 2071, 12, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2073, 'Filip', 'George', 41, 7500.00, 2069, 25, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2074, 'Iacob', 'Daniela', 36, 6200.00, 2073, 25, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2075, 'Sandu', 'Roxana', 32, 5600.00, 2074, 25, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2076, 'Mocanu', 'Victor', 45, 7800.00, 2069, 11, 3006);
```

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2077, 'Dinu', 'Elena', 38, 6700.00, 2076, 11, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2078, 'Grigore', 'Alin', 34, 5900.00, 2077, 11, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2079, 'Barbu', 'Ioana', 43, 7300.00, 2069, 21, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2080, 'Pavel', 'Radu', 39, 6600.00, 2079, 21, 3006);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2081, 'Roman', 'Gabriela', 29, 5300.00, 2080, 21, 3006);
```

SELECT USER FROM DUAL;

SELECT *FROM angajati_PROIECT;

COMMIT;

The screenshot shows three tabs in the SQL Developer interface:

- Script Output**: Displays the execution of an insert script with 82 rows inserted.
- Query Result**: Displays the result of the SELECT USER FROM DUAL query, showing 'SIMONG_65'.
- SQL**: Displays the final COMMIT statement.

The bottom part of the screenshot shows the results of the SELECT * FROM angajati_PROJECT query, listing 82 rows of employee data.

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARUL	ID_MANAGER	ID_DEPARTAMENTE	ID_BANCA
73	2072Olaru	Cristina	33	5700	2071	12	3006
74	2073Filip	George	41	7800	2069	25	3006
75	2074Iacob	Daniela	36	6200	2073	25	3006
76	2075Sandu	Roxana	32	5600	2074	25	3006
77	2076Mocanu	Victor	45	7800	2069	11	3006
78	2077Dinu	Elena	38	6700	2076	11	3006
79	2078Grigore	Alin	34	5900	2077	11	3006
80	2079Barbu	Ioana	43	7300	2069	21	3006
81	2080Pavel	Radu	39	6600	2079	21	3006
82	2081Roman	Gabriela	29	5300	2080	21	3006

8)BANCA ALPHA

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2082, 'Gheorghiu', 'Adrian', 50, 8900.00, NULL, 20, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2083, 'Vlad', 'Simona', 42, 7600.00, 2082, 18, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2084, 'Stan', 'Marius', 38, 6500.00, 2083, 18, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2085, 'Ionescu', 'Andreea', 34, 5700.00, 2084, 18, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2086, 'Nistor', 'Alina', 44, 7400.00, 2082, 15, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2087, 'Voicu', 'Daniel', 40, 6900.00, 2086, 15, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2088, 'Marcu', 'Ioana', 32, 5800.00, 2087, 15, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2089, 'Andrei', 'George', 39, 7100.00, 2082, 16, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2090, 'Pana', 'Elena', 36, 6200.00, 2089, 16, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2091, 'Serban', 'Ana', 31, 5600.00, 2090, 16, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2092, 'Popa', 'Radu', 47, 7800.00, 2082, 23, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2093, 'Lazar', 'Claudia', 40, 6900.00, 2092, 23, 3007);
INSERT INTO angajati_PROIECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2094, 'Preda', 'Cristian', 29, 5300.00, 2093, 23, 3007);
SELECT USER FROM DUAL;
SELECT *FROM angajati_PROIECT;
COMMIT;
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2091, 'Seban', 'Ana', 35, 5600.00, 2090, 16, 3007);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2092, 'Popa', 'Radu', 47, 7800.00, 2091, 21, 3007);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2093, 'Laser', 'Claudia', 40, 6900.00, 2092, 23, 3007);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2094, 'Preda', 'Cristian', 29, 5300.00, 2093, 23, 3007);

Script Output A | Query Result X
SQL | All Rows Fetched: 1 in 0.016 seconds
USER
1 SIMONG_65

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca)
VALUES (2095, 'Laser', 'Claudia', 40, 6900.00, 2092, 23, 3007);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca)
VALUES (2096, 'Preda', 'Cristian', 29, 5300.00, 2093, 23, 3007);
SELECT USER FROM DUAL;
SELECT *FROM angajati_PROJECT;
COMMIT;

Script Output X | Query Result X
SQL | All Rows Fetched: 95 in 0.086 seconds
ID_ANGAJAT | NUME_ANGAJAT | PRENUME_ANGAJAT | VARSTA | SALARIUL | ID_MANAGER | ID_DEPARTAMENTE | ID_BANCA
85 | 2084 Stan | Marius | 38 | 6500 | 2083 | 18 | 3007
86 | 2095 Andreescu | Andreea | 34 | 5700 | 2084 | 18 | 3007
87 | 2086 Ionescu | Alina | 44 | 7400 | 2082 | 15 | 3007
88 | 2087 Voicu | Daniel | 40 | 6900 | 2086 | 15 | 3007
89 | 2088 Marcu | Ioana | 32 | 5800 | 2087 | 15 | 3007
90 | 2089 Andrei | George | 39 | 7100 | 2082 | 16 | 3007
91 | 2090 Pana | Elena | 36 | 6200 | 2089 | 16 | 3007
92 | 2091 Serban | Ana | 31 | 5600 | 2090 | 16 | 3007
93 | 2092 Popa | Radu | 47 | 7800 | 2082 | 23 | 3007
94 | 2093 Laser | Claudia | 40 | 6900 | 2092 | 23 | 3007
95 | 2094 Preda | Cristian | 29 | 5300 | 2093 | 23 | 3007

```

9)BANCA UNICREDIT

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2095, 'Miron', 'Valeriu', 52, 9100.00, NULL, 20, 3008);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2096, 'Dobre', 'Alina', 44, 7800.00, 2095, 10, 3008);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2097, 'Vasile', 'Dan', 38, 6700.00, 2096, 10, 3008);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2098, 'Baciu', 'Irina', 34, 5700.00, 2097, 10, 3008);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2099, 'Neacsu', 'Ovidiu', 46, 7600.00, 2095, 21, 3008);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2100, 'Cojocaru', 'Raluca', 39, 6500.00, 2099, 21, 3008);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2101, 'Stoenescu', 'Mihai', 33, 5500.00, 2100, 21, 3008);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2102, 'Lupescu', 'Cristina', 48, 7900.00, 2095, 24, 3008);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2103, 'Rosu', 'Ionel', 41, 6900.00, 2102, 24, 3008);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariul, id_manager, id_departamente, id_banca) VALUES (2104, 'Marin', 'Claudia', 36, 5800.00, 2103, 24, 3008);
 SELECT USER FROM DUAL;

SELECT *FROM angajati_PROJECT;

COMMIT;

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2105, 'Popescu', 'Florin', 55, 9200.00, NULL, 20, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2106, 'Iacob', 'Daniela', 47, 7800.00, 2105, 13, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2107, 'Grigorescu', 'Andrei', 41, 6700.00, 2106, 13, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2108, 'Petrescu', 'Roxana', 37, 5600.00, 2107, 13, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2109, 'Stoica', 'Adrian', 45, 7500.00, 2105, 22, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2110, 'Avram', 'Mihaela', 40, 6400.00, 2109, 22, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2111, 'Moraru', 'Cristian', 33, 5400.00, 2110, 22, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2112, 'Furtuna', 'Ion', 50, 7700.00, 2105, 25, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2113, 'Enache', 'Alina', 43, 6700.00, 2112, 25, 3009);
INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2114, 'Olaru', 'George', 38, 5700.00, 2113, 25, 3009);

SELECT USER FROM DUAL;
SELECT *FROM angajati_PROJECT;
COMMIT;

```

Script Output | Query Result | Task completed in 0.041 seconds
1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.

Script Output | Query Result | All Rows Fetched: 1 in 0.034 seconds
USER
1 SIMONG_65

10) BANCA ING

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2105, 'Popescu', 'Florin', 55, 9200.00, NULL, 20, 3009);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2106, 'Iacob', 'Daniela', 47, 7800.00, 2105, 13, 3009);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2107, 'Grigorescu', 'Andrei', 41, 6700.00, 2106, 13, 3009);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2108, 'Petrescu', 'Roxana', 37, 5600.00, 2107, 13, 3009);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2109, 'Stoica', 'Adrian', 45, 7500.00, 2105, 22, 3009);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2110, 'Avram', 'Mihaela', 40, 6400.00, 2109, 22, 3009);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2111, 'Moraru', 'Cristian', 33, 5400.00, 2110, 22, 3009);

INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2112, 'Furtuna', 'Ion', 50, 7700.00, 2105, 25, 3009);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2113, 'Enache', 'Alina', 43, 6700.00, 2112, 25, 3009);
 INSERT INTO angajati_PROJECT (id_angajat, nume_angajat, prenume_angajat, varsta, salariu, id_manager, id_departamente, id_banca) VALUES (2114, 'Olaru', 'George', 38, 5700.00, 2113, 25, 3009);
 SELECT USER FROM DUAL;
 SELECT *FROM angajati_PROJECT;
 COMMIT;

PROIECT – BAZE DE DATE – SIMON GEORGIANA

The screenshot shows four tabs in the Oracle SQL Developer interface:

- Script Output**: Shows two INSERT statements for the 'angajati_PROIECT' table.
- Query Result**: Shows the results of the first INSERT statement (1 row inserted).
- Script Output**: Shows the second INSERT statement for the 'angajati_PROIECT' table.
- Query Result**: Shows the results of the second INSERT statement (1 row inserted).
- Script Output**: Shows the final part of the script including a SELECT from DUAL, a SELECT * from 'angajati_PROIECT', and a COMMIT.
- Query Result**: Shows the results of the final SELECT query, displaying a single row: USER SIMONG_65.
- Script Output**: Shows the entire script again for reference.
- Query Result**: Shows the final output of the entire script, listing 15 rows of employee data.

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARIU	ID_MANAGER	ID_DEPARTAMENT	ID_BANCA
106	2105Popescu	Florin	55	9200	(null)	20	3009
107	2106Iacob	Daniela	47	7800	2105	13	3009
108	2107Dragorescu	Andrei	41	6700	2106	13	3009
109	2108Petreasu	Roxana	37	5600	2107	13	3009
110	2109Stolca	Adrian	45	7500	2105	22	3009
111	2110Avram	Mihaila	40	6400	2109	22	3009
112	2111Moraru	Cristian	33	5400	2110	22	3009
113	2112Purume	Ion	50	7700	2105	25	3009
114	2113Enache	Alina	43	6700	2112	25	3009
115	2114Olaru	George	38	5700	2113	25	3009

8) Tabela incasari_PROIECT

DESCRIBE incasari_PROIECT;

SELECT id_client FROM clienti_PROIECT;

ALTER TABLE incasari_PROIECT

MODIFY (descriere VARCHAR2(60), suma_incasata NUMBER(6, 2));

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500001, 1000, 1500.00, TO_DATE('2024-12-01', 'YYYY-MM-DD'), 'plata datorie luna decembrie');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500002, 1000, 1700.00, TO_DATE('2024-12-03', 'YYYY-MM-DD'), 'achitare imprumut personal');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500003, 1001, 1750.00, TO_DATE('2024-12-05', 'YYYY-MM-DD'), 'plata factura utilitatii');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500004, 1001, 1900.00, TO_DATE('2024-12-07', 'YYYY-MM-DD'), 'transfer pentru cumparaturi');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500005, 1002, 2000.00, TO_DATE('2024-12-09', 'YYYY-MM-DD'), 'achitare colet');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500006, 1002, 2200.00, TO_DATE('2024-12-11', 'YYYY-MM-DD'), 'plata factura telefonica');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500007, 1003, 2300.00, TO_DATE('2024-12-13', 'YYYY-MM-DD'), 'achitare credit auto');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500008, 1003, 2500.00, TO_DATE('2024-12-15', 'YYYY-MM-DD'), 'plata avans chirie');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500009, 1004, 1800.00, TO_DATE('2024-12-17', 'YYYY-MM-DD'), 'plata pentru chirie');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500010, 1004, 2000.00, TO_DATE('2024-12-19', 'YYYY-MM-DD'), 'achitare imprumut');

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500011, 1005, 1600.00, TO_DATE('2024-12-21', 'YYYY-MM-DD'), 'plata meditatie');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500012, 1005, 1700.00, TO_DATE('2024-12-23', 'YYYY-MM-DD'), 'achitare factura apa');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500013, 1006, 2100.00, TO_DATE('2024-12-25', 'YYYY-MM-DD'), 'plata factura curent');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500014, 1006, 2300.00, TO_DATE('2024-12-27', 'YYYY-MM-DD'), 'achitare imprumut');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500015, 1007, 2400.00, TO_DATE('2024-12-29', 'YYYY-MM-DD'), 'plata pentru
cumparaturi');

INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500016, 1007, 2500.00, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'plata imprumut');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500017, 1008, 2300.00, TO_DATE('2025-01-02', 'YYYY-MM-DD'), 'achitare factura gaze');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500018, 1008, 2500.00, TO_DATE('2025-01-04', 'YYYY-MM-DD'), 'plata factura inretinere');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500019, 1009, 2000.00, TO_DATE('2025-01-06', 'YYYY-MM-DD'), 'plata chirie');
INSERT INTO incasari_PROIECT (id_incasare, id_client, suma_incasata, data_incasarii, descriere)
VALUES (500020, 1009, 2200.00, TO_DATE('2025-01-08', 'YYYY-MM-DD'), 'achitare chirie');

SELECT USER FROM DUAL;
SELECT *FROM incasari_PROIECT;
COMMIT;

```

ID_INCASARE	ID_CLIENT	SUMA_INCASATA	DATA_INCASARII	DESCRIERE
1	500001	1000	1500 01-DEC-24	plata datoriei luna decembrie
2	500002	1000	1700 03-DEC-24	achitare imprumut personal
3	500003	1001	1750 05-DEC-24	plata factura utilitatii
4	500004	1001	1900 07-DEC-24	transfer pentru cumparaturi
5	500005	1002	2000 09-DEC-24	achitare colet
6	500006	1002	2200 11-DEC-24	plata factura telefonia
7	500007	1003	2300 13-DEC-24	achitare credit auto
8	500008	1003	2500 15-DEC-24	plata avans chirie
9	500009	1004	1800 17-DEC-24	plata pentru chirie

9) Tabela transferuri_PROIECT

DESCRIBE transferuri_PROIECT;

SELECT id_cont FROM conturi_PROIECT;

ALTER TABLE transferuri_PROIECT

MODIFY (suma_transferata NUMBER(6, 2), comision NUMBER(4, 2));

INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)

VALUES (5000, 9000, 1500.00, 15.00, TO_DATE('2024-12-01', 'YYYY-MM-DD'));

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5001, 9000, 2000.00, 20.00, TO_DATE('2024-12-05', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5002, 9001, 2500.00, 25.00, TO_DATE('2024-12-08', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5003, 9001, 3000.00, 30.00, TO_DATE('2024-12-10', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5004, 9002, 3500.00, 35.00, TO_DATE('2024-12-12', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5005, 9002, 4000.00, 40.00, TO_DATE('2024-12-14', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5006, 9003, 4500.00, 45.00, TO_DATE('2024-12-16', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5007, 9003, 5000.00, 50.00, TO_DATE('2024-12-18', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5008, 9004, 5500.00, 45.00, TO_DATE('2024-12-20', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5009, 9004, 6000.00, 50.00, TO_DATE('2024-12-22', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5010, 9005, 6500.00, 35.00, TO_DATE('2024-12-24', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5011, 9005, 7000.00, 30.00, TO_DATE('2024-12-26', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5012, 9006, 7500.00, 25.00, TO_DATE('2024-12-28', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5013, 9006, 8000.00, 40.00, TO_DATE('2024-12-30', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5014, 9007, 8500.00, 45.00, TO_DATE('2025-01-01', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5015, 9007, 9000.00, 10.00, TO_DATE('2025-01-03', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5016, 9008, 9500.00, 35.00, TO_DATE('2025-01-05', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5017, 9008, 1000.00, 50.00, TO_DATE('2025-01-07', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5018, 9009, 1050.00, 5.00, TO_DATE('2025-01-09', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5019, 9009, 1100.00, 10.00, TO_DATE('2025-01-11', 'YYYY-MM-DD'));
INSERT INTO transferuri_PROIECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5020, 9010, 1150.00, 15.00, TO_DATE('2025-01-13', 'YYYY-MM-DD'));
SELECT USER FROM DUAL;
SELECT *FROM transferuri_PROIECT;
COMMIT;
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

INSERT INTO transferuri_PROJECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5020, 9010, 1150.00, 15.00, TO_DATE('2025-01-13', 'YYYY-MM-DD'));

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

VALUES (5019, 9009, 1100.00, 10.00, TO_DATE('2025-01-11', 'YYYY-MM-DD'));

INSERT INTO transferuri_PROJECT (id_transfer, id_cont, suma_transferata, comision, data_transfer)
VALUES (5020, 9010, 1150.00, 15.00, TO_DATE('2025-01-13', 'YYYY-MM-DD'));

SELECT USER FROM DUAL;

1 SIMONG_65

SELECT *FROM transferuri_PROJECT;

```

ID_TRANSFER	ID_CONT	SUMA_TRANSFERATA	COMISION	DATA_TRANSFER
1	5000	9000	1500	15.01-DEC-24
2	5001	9000	2000	20.05-DEC-24
3	5002	9001	2500	25.08-DEC-24
4	5003	9001	3000	30.10-DEC-24
5	5004	9002	3500	35.12-DEC-24
6	5005	9002	4000	40.14-DEC-24
7	5006	9003	4500	45.16-DEC-24
8	5007	9003	5000	50.18-DEC-24
9	5008	9004	5500	45.20-DEC-24
10	5009	9004	6000	50.22-DEC-24
11	5010	9005	6500	35.24-DEC-24
12	5011	9005	7000	30.26-DEC-24
13	5012	9006	7500	25.28-DEC-24
14	5013	9006	8000	40.30-DEC-24

10) Tabela tranzactii_PROIECT

DESCRIBE tranzactii_PROIECT;

ALTER TABLE tranzactii_PROIECT

MODIFY suma_tranzactiei NUMBER(6, 2);

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600000, 9000, 1000, 1200.00, TO_DATE('2024-12-01', 'YYYY-MM-DD'), 'Depunere');

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600001, 9001, 1001, 1500.00, TO_DATE('2024-12-03', 'YYYY-MM-DD'), 'Retragere');

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600002, 9001, 1001, 2000.00, TO_DATE('2024-12-05', 'YYYY-MM-DD'), 'Depunere');

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600003, 9001, 1001, 2500.00, TO_DATE('2024-12-07', 'YYYY-MM-DD'), 'Retragere');

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600004, 9002, 1002, 3000.00, TO_DATE('2024-12-09', 'YYYY-MM-DD'), 'Depunere');

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600005, 9002, 1002, 3500.00, TO_DATE('2024-12-11', 'YYYY-MM-DD'), 'Retragere');

INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600006, 9003, 1003, 4000.00, TO_DATE('2024-12-13', 'YYYY-MM-DD'), 'Depunere');

PROIECT – BAZE DE DATE – ŞIMON GEORGIANA

```
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600007, 9003, 1003, 4500.00, TO_DATE('2024-12-15', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600008, 9003, 1003, 5000.00, TO_DATE('2024-12-17', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600009, 9003, 1003, 5500.00, TO_DATE('2024-12-19', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600010, 9004, 1004, 6000.00, TO_DATE('2024-12-21', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600011, 9005, 1005, 6500.00, TO_DATE('2024-12-23', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600012, 9005, 1005, 7000.00, TO_DATE('2024-12-25', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600013, 9005, 1005, 7500.00, TO_DATE('2024-12-27', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600014, 9005, 1005, 8000.00, TO_DATE('2024-12-29', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600015, 9007, 1015, 8500.00, TO_DATE('2025-01-01', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600016, 9008, 1016, 9000.00, TO_DATE('2025-01-03', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600017, 9008, 1016, 9500.00, TO_DATE('2025-01-05', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600018, 9009, 1018, 1000.00, TO_DATE('2025-01-07', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO tranzactii_PROIECT (id_tranzactie, id_cont, id_client, suma_tranzactiei, data_tranzactiei, tip_tranzactie) VALUES (600019, 9009, 1018, 1050.00, TO_DATE('2025-01-09', 'YYYY-MM-DD'), 'Retragere');
SELECT USER FROM DUAL;
SELECT *FROM tranzactii_PROIECT;
COMMIT;
```

```

INSERT INTO transactii_PROJECT (id_transactie, id_cont, id_client, suma_transactie)
VALUES(9019, 9018, 1018, 1000.00, TO_DATE('2023-01-09', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO transactii_PROJECT (id_transactie, id_cont, id_client, suma_transactie)
VALUES(9016, 9016, 1016, 9000.00, TO_DATE('2023-01-03', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO transactii_PROJECT (id_transactie, id_cont, id_client, suma_transactie)
VALUES(9001, 9012, 1012, 9500.00, TO_DATE('2023-01-01', 'YYYY-MM-DD'), 'Depunere');
INSERT INTO transactii_PROJECT (id_transactie, id_cont, id_client, suma_transactie)
VALUES(9018, 9018, 1018, 1000.00, TO_DATE('2023-01-09', 'YYYY-MM-DD'), 'Retragere');
INSERT INTO transactii_PROJECT (id_transactie, id_cont, id_client, suma_transactie)
VALUES(9019, 9018, 1018, 1050.00, TO_DATE('2023-01-09', 'YYYY-MM-DD'), 'Retragere');

SELECT USER FROM DUAL;
  
```

ID_TRANACTIE	ID_CONT	ID_CLIENT	SUMA_TRANACTIE	DATA_TRANACTIE	TIP_TRANACTIE
1	600000	9008	1000	12/01/2023-24	Depunere
2	600001	9001	1001	15/01/2023-24	Retragere
3	600002	9001	1001	20/01/2023-24	Depunere
4	600003	9001	1001	25/01/2023-24	Retragere
5	600004	9002	1002	30/01/2023-24	Depunere
6	600005	9002	1002	35/01/2023-24	Retragere
7	600006	9003	1003	40/01/2023-24	Depunere
8	600007	9003	1003	45/01/2023-24	Retragere
9	600008	9003	1003	50/01/2023-24	Depunere
10	600009	9003	1003	55/01/2023-24	Retragere
11	600010	9004	1004	60/01/2023-24	Depunere
12	600011	9005	1005	65/01/2023-24	Retragere
13	600012	9005	1005	70/01/2023-24	Depunere
14	600013	9005	1005	75/01/2023-24	Retragere
15	600014	9018	1005	80/01/2023-24	Depunere
16	600015	9018	1015	85/01/2023-25	Retragere
17	600016	9016	1016	90/01/2023-25	Depunere

11) Tabela credite_PROIECT

DESCRIBE credite_PROIECT;

ALTER TABLE credite_PROIECT

MODIFY suma_credit NUMBER(7, 2);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700001, 5000.00, 4.50, TO_DATE('2024-01-15', 'YYYY-MM-DD'), TO_DATE('2029-01-15', 'YYYY-MM-DD'), 1000);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700002, 10000.00, 5.00, TO_DATE('2024-02-01', 'YYYY-MM-DD'), TO_DATE('2030-02-01', 'YYYY-MM-DD'), 1000);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700003, 7000.00, 4.80, TO_DATE('2024-03-10', 'YYYY-MM-DD'), TO_DATE('2029-03-10', 'YYYY-MM-DD'), 1001);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700004, 15000.00, 6.00, TO_DATE('2024-04-20', 'YYYY-MM-DD'), TO_DATE('2030-04-20', 'YYYY-MM-DD'), 1001);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700005, 8000.00, 5.50, TO_DATE('2024-05-15', 'YYYY-MM-DD'), TO_DATE('2029-05-15', 'YYYY-MM-DD'), 1002);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700006, 20000.00, 4.20, TO_DATE('2024-06-01', 'YYYY-MM-DD'), TO_DATE('2030-06-01', 'YYYY-MM-DD'), 1002);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700007, 12000.00, 5.30, TO_DATE('2024-07-10', 'YYYY-MM-DD'), TO_DATE('2029-07-10', 'YYYY-MM-DD'), 1003);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700008, 6000.00, 4.60, TO_DATE('2024-08-05', 'YYYY-MM-DD'), TO_DATE('2029-08-05', 'YYYY-MM-DD'), 1003);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700009, 11000.00, 5.10, TO_DATE('2024-09-20', 'YYYY-MM-DD'), TO_DATE('2029-09-20', 'YYYY-MM-DD'), 1004);

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_inceperii, data_finalizare, id_client) VALUES (700010, 13000.00, 4.75, TO_DATE('2024-10-15', 'YYYY-MM-DD'), TO_DATE('2029-10-15', 'YYYY-MM-DD'), 1004);

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700011, 9000.00, 5.60, TO_DATE('2024-11-01', 'YYYY-MM-DD'),
TO_DATE('2029-11-01', 'YYYY-MM-DD'), 1005);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700012, 16000.00, 5.20, TO_DATE('2024-12-01', 'YYYY-MM-DD'),
TO_DATE('2030-12-01', 'YYYY-MM-DD'), 1005);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700013, 9500.00, 4.90, TO_DATE('2025-01-10', 'YYYY-MM-DD'),
TO_DATE('2030-01-10', 'YYYY-MM-DD'), 1006);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700014, 11000.00, 5.30, TO_DATE('2025-02-05', 'YYYY-MM-DD'),
TO_DATE('2030-02-05', 'YYYY-MM-DD'), 1006);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700015, 7000.00, 4.50, TO_DATE('2025-03-20', 'YYYY-MM-DD'),
TO_DATE('2029-03-20', 'YYYY-MM-DD'), 1007);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700016, 18000.00, 6.20, TO_DATE('2025-04-10', 'YYYY-MM-DD'),
TO_DATE('2030-04-10', 'YYYY-MM-DD'), 1007);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700017, 14000.00, 5.40, TO_DATE('2025-05-01', 'YYYY-MM-DD'),
TO_DATE('2030-05-01', 'YYYY-MM-DD'), 1008);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700018, 20000.00, 4.90, TO_DATE('2025-06-15', 'YYYY-MM-DD'),
TO_DATE('2030-06-15', 'YYYY-MM-DD'), 1008);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700019, 11000.00, 5.00, TO_DATE('2025-07-25', 'YYYY-MM-DD'),
TO_DATE('2030-07-25', 'YYYY-MM-DD'), 1009);
INSERT INTO credite_PROIECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare,
id_client) VALUES (700020, 13000.00, 4.80, TO_DATE('2025-08-12', 'YYYY-MM-DD'),
TO_DATE('2030-08-12', 'YYYY-MM-DD'), 1009);
SELECT USER FROM DUAL;
SELECT *FROM credite_PROIECT;
COMMIT;

```

The screenshot shows the Oracle SQL Developer interface with two panes. The left pane displays the script being run:

```

INSERT INTO credite_PROJECT (id_credit, suma_credit, dobanda, data_incepereii, data_finalizare, id_client)
VALUES (700020, 13000.00, 4.80, TO_DATE('2025-08-12', 'YYYY-MM-DD'), TO_DATE('2030-08-12', 'YYYY-MM-DD'), 1009);

SELECT USER FROM DUAL;

```

The right pane shows the results of the query:

ID_CREDIT	SUMA_CREDIT	DOBANDA	DATA_INCEPERII	DATA_FINALIZARE	ID_CLIENT
1	700001	5000	4.5 15-JAN-24	15-JAN-29	1000
2	700002	10000	5.0 1-FEB-24	01-FEB-30	1000
3	700003	10000	4.8 10-MAR-24	10-MAR-29	1001
4	700004	15000	6.2 20-APR-24	20-APR-30	1001
5	700005	8000	5.5 15-MAY-24	15-MAY-29	1002
6	700006	20000	4.2 01-JUN-24	01-JUN-30	1002
7	700007	12000	5.3 10-JUL-24	10-JUL-29	1003
8	700008	6000	4.6 05-AUG-24	05-AUG-29	1003
9	700009	11000	5.1 20-SEP-24	20-SEP-29	1004
10	700010	13000	4.7 15-OCT-24	15-OCT-29	1004
11	700011	9000	5.6 01-NOV-24	01-NOV-29	1005
12	700012	16000	5.2 01-DEC-24	01-DEC-30	1005
13	700013	9500	4.9 10-JAN-25	10-JAN-30	1006
14	700014	11000	5.3 05-FEB-25	05-FEB-30	1006

12) Tabela rate_credit_bancare_PROIECT2
DESCRIBE rate_credit_bancare_PROIECT2;
ALTER TABLE rate_credit_bancare_PROIECT2

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
MODIFY suma_rata NUMBER(7, 2);
INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800001, 700001, 950.00, TO_DATE('2024-02-15', 'YYYY-MM-DD'), 'Neachitat'); INSERT
INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata) VALUES (800002, 700001, 950.00, TO_DATE('2024-03-15', 'YYYY-MM-DD'), 'Neachitat');
INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata) VALUES (800003, 700002, 1500.00, TO_DATE('2024-03-01', 'YYYY-MM-DD'), 'Achitat');
INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata) VALUES (800004, 700002, 1500.00, TO_DATE('2024-04-01', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800005, 700003, 1050.00, TO_DATE('2024-04-10', 'YYYY-MM-DD'), 'Achitat'); INSERT
INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800006, 700003, 1050.00, TO_DATE('2024-05-10', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800007, 700004, 2500.00, TO_DATE('2024-05-20', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800008, 700004, 2500.00, TO_DATE('2024-06-20', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800009, 700005, 1500.00, TO_DATE('2024-06-10', 'YYYY-MM-DD'), 'Achitat'); INSERT
INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800010, 700005, 1500.00, TO_DATE('2024-07-10', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800011, 700006, 1600.00, TO_DATE('2024-07-15', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800012, 700006, 1600.00, TO_DATE('2024-08-15', 'YYYY-MM-DD'), 'Achitat'); INSERT
INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800013, 700007, 1400.00, TO_DATE('2024-08-25', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata) VALUES (800014, 700007, 1400.00, TO_DATE('2024-09-25', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800015, 700008, 1600.00, TO_DATE('2024-09-01', 'YYYY-MM-DD'), 'Achitat'); INSERT
INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata) VALUES (800016, 700008, 1600.00, TO_DATE('2024-10-01', 'YYYY-MM-DD'),
'Neachitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata) VALUES (800017, 700009, 1800.00, TO_DATE('2024-10-15', 'YYYY-MM-DD'), 'Achitat');

INSERT INTO rate_credit_bancare_PROIECT2 (id_rata, id_credit, suma_rata, data_scadenta,
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

status_rata) VALUES (800018, 700009, 1800.00, TO_DATE('2024-11-15', 'YYYY-MM-DD'),
'Nechitat');

INSERT INTO rate_credit_bancare_PROJECT2 (id_rata, id_credit, suma_rata, data_scadenta, status_rata)
VALUES (800019, 700010, 1900.00, TO_DATE('2024-11-05', 'YYYY-MM-DD'), 'Achitat'); INSERT
INTO rate_credit_bancare_PROJECT2 (id_rata, id_credit, suma_rata, data_scadenta,
status_rata)VALUES (800020, 700010, 1900.00, TO_DATE('2024-12-05', 'YYYY-MM-DD'),
'Nechitat');

SELECT USER FROM DUAL;
SELECT *FROM rate_credit_bancare_PROJECT2;
COMMIT;

```

ID_RATA	ID_CREDIT	SUMA_RATA	DATA_SCADENTA	STATUS_RATA
1	800001	950.15	FEB-24	Nechitat
2	800002	700001	950.15-MAR-24	Nechitat
3	800003	700002	1500.01-MAR-24	Achitat
4	800004	700002	1500.01-APR-24	Nechitat
5	800005	700003	1050.10-APR-24	Achitat
6	800006	700003	1050.10-MAY-24	Nechitat
7	800007	700004	2500.20-MAY-24	Nechitat
8	800008	700004	2500.20-JUN-24	Nechitat
9	800009	700005	1500.10-JUN-24	Achitat
10	800010	700005	1500.10-JUL-24	Nechitat
11	800011	700006	1600.15-JUL-24	Nechitat
12	800012	700006	1600.15-AUG-24	Achitat
13	800013	700007	1400.25-AUG-24	Nechitat
14	800014	700007	1400.25-SEP-24	Nechitat

V. ACTUALIZARI IN TABELE. INTEROGARI. COMENZI COMPLEXE

1) Sa se actualizeze soldul contului pentru toti clientii care au carduri de tip „Credit” si ale caror card expira in anul 2027. Noua valoare a soldului va fi reprezentata de o majorare a soldului cu 10%.

```

SELECT sold FROM conturi_PROJECT WHERE id_cont IN (SELECT id_cont FROM
carduri_PROJECT WHERE tip_card='Credit' AND EXTRACT(YEAR FROM data_expirare)=2027);
UPDATE conturi_PROJECT
SET sold=sold+0.1*sold
WHERE id_cont IN (SELECT id_cont FROM carduri_PROJECT WHERE tip_card='Credit'
AND EXTRACT(YEAR FROM data_expirare)=2027);

```

(4 randuri)

2) Sa se recalculeze salariul pentru angajatii care lucreaza in departamentul „Marketing” si au varsta sub 30 de ani, salariul va fi majorat cu 10%. Daca salariul depaseste 5000 RON, majorarea va fi de doar 15%.

```

SELECT *FROM angajati_PROJECT WHERE id_departamente IN (SELECT id_departamente FROM
departamente_PROJECT WHERE denumire_departament='Marketing') AND varsta<30;
UPDATE angajati_PROJECT
SET salariul= CASE

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

WHEN salariul>5000 THEN salariul+0.15*salariul
WHEN salariul<5000 THEN salariul+0.1*salariul
ELSE salariul
END
WHERE id_departamente IN (SELECT id_departamente
FROM departamente_PROIECT WHERE upper(denumire_departament)='MARKETING') AND
varsta<30;
SELECT *FROM angajati_PROIECT WHERE
id_departamente IN (SELECT id_departamente FROM departamente_PROIECT WHERE
denumire_departament='Marketing') AND varsta<30;
SELECT USER FROM DUAL;

```

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA	SALARIUL	ID_MANAGER	ID_DEPARTAMENTE	ID_BANCA
1	2021 Ion	Andreea	27	5225	2020	13	3001
2	2037 Ciucă	Ioana	29	6210	2036	13	3003

3) Sa se actualizeze suma_transferata din tabela transferuri_PROIECT, astfel incat daca suma transfertaa este mai mare de 5000, din luna curenta! i se aplica un comisio suplimentar in valoarea procentuala de 10%, suma finala se va calcula ca diferența intre suma initiala si comisionul alocat - valoarea comisionului din coloana comision. (este adaugata ca fiind o suma in RON, nu procentual)

UPDATE transferuri_PROIECT

```

SET suma_transferata=suma_transferata-suma_transferata*0.1-comision
WHERE EXTRACT(MONTH FROM data_transfer)=EXTRACT(MONTH FROM SYSDATE)
AND suma_transferata>5000;
SELECT suma_transferata FROM transferuri_PROIECT WHERE EXTRACT(MONTH FROM data_transfer)=EXTRACT(MONTH FROM SYSDATE) AND suma_transferata>5000;

```

SUMA_TRANSFERATA
7605
8090
8515

(3

randuri)

PROIECT – BAZE DE DATE – SIMON GEORGIANA

- 4) Sa se calculeze suma totala a tranzactiilor pentru fiecare tip de cont. Sa se afiseze doar tipurile de cont cu tranzactii care depasesc suma de 30000 RON.

```
DESCRIBE tranzactii_PROIECT;
SELECT c.tip_cont, SUM(t.suma_tranzactie) AS total
FROM conturi_PROIECT c, tranzactii_PROIECT t
WHERE c.id_cont=t.id_cont
GROUP BY c.tip_cont
HAVING SUM(t.suma_tranzactie)>30000;
```

```
SELECT USER FROM DUAL;
```

The screenshot shows the SQL editor with the query:

```
SELECT c.tip_cont, SUM(t.suma_tranzactie)
FROM conturi_PROIECT c, tranzactii_PROIECT t
WHERE c.id_cont=t.id_cont
GROUP BY c.tip_cont
HAVING SUM(t.suma_tranzactie)>30000;
```

The results are displayed in the 'Query Result' tab:

TIP_CONT	TOTAL
1 Economii	31500
2 Curent	37500

Below the results, the 'Script Output' tab shows the command:

```
SELECT USER FROM DUAL;
```

The output of the 'Script Output' tab is shown in the 'Query Result' tab:

USER
1 SIMONG_65

- 5) Sa se identifice clientii pentru care suma totala a tranzactiilor este mai mica decat suma totala a incasarilor.

```
SELECT c.id_client, c.nume_client, c.prenume_client,
(SELECT NVL(SUM(t.suma_tranzactie), 0)
FROM tranzactii_PROIECT t
WHERE t.id_client = c.id_client) AS suma_tranzactii,
(SELECT NVL(SUM(i.suma_incasata), 0)
FROM incasari_PROIECT i
WHERE i.id_client = c.id_client) AS suma_incasari
FROM clienti_PROIECT c
WHERE(SELECT NVL(SUM(t.suma_tranzactie), 0)
FROM tranzactii_PROIECT t WHERE t.id_client = c.id_client) < (SELECT
NVL(SUM(i.suma_incasata), 0) FROM incasari_PROIECT i
WHERE i.id_client = c.id_client);
SELECT USER FROM DUAL;
```

The screenshot shows the SQL editor with the query:

```
SELECT c.id_client, c.nume_client, c.prenume_client,
(SELECT NVL(SUM(t.suma_tranzactie), 0)
FROM tranzactii_PROIECT t
WHERE t.id_client = c.id_client) AS suma_tranzactii,
(SELECT NVL(SUM(i.suma_incasata), 0)
FROM incasari_PROJECT i
WHERE i.id_client = c.id_client) AS suma_incasari
WHERE t.id_client = c.id_client) <
(SELECT NVL(SUM(i.suma_incasata), 0)
FROM incasari_PROJECT i
WHERE i.id_client = c.id_client);
SELECT USER FROM DUAL;
```

The results are displayed in the 'Query Result' tab:

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	SUMA_TRANZACTII	SUMA_INCASARI
1	1000 Simon	Georgiana	1200	3200
2	1001 Ilinoiu	Lucian	1500	3650
3	1002 Marinca	Ana	2000	4200
4	1003 Dumitru	Stefan	2500	4800
5	1004 Marin	Ioana	3000	3800
6	1006 Stoica	Elena	4000	4400
7	1007 Vasiliu	Daniel	4500	4900

Below the results, the 'Script Output' tab shows the command:

```
SELECT USER FROM DUAL;
```

The output of the 'Script Output' tab is shown in the 'Query Result' tab:

USER
1 SIMONG_65

- 6) Sa se afiseze angajati care fac parte din departamentul “IT si suport tehnic” in ordine ierarhica, sub forma unei organigrame

```
SELECT LEVEL, LPAD(' ', (LEVEL - 1) * 4, '') || a.nume_angajat || ' ' || a.prenume_angajat AS
```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```
"Organigrama", a.id_manager
FROM angajati_PROIECT a, departamente_PROIECT d
WHERE a.id_departamente = d.id_departamente
AND UPPER(d.denumire_departament) = 'IT SI SUPORT TEHNIC'
START WITH a.id_manager IS NULL
CONNECT BY PRIOR a.id_angajat = a.id_manager
ORDER BY LEVEL;
SELECT USER FROM DUAL;
```

```
a.id_manager
FROM angajati_PROIECT a, departamente_PROIECT d
WHERE a.id_departamente = d.id_departamente
AND UPPER(d.denumire_departament) = 'IT SI SUPORT TEHNIC'
START WITH a.id_manager IS NULL
CONNECT BY PRIOR a.id_angajat = a.id_manager
ORDER BY LEVEL;

SELECT USER FROM DUAL;
```

LEVEL	Organigrama	ID Manager
1	2	Balan Roberta
2	2	Matei Andreea
3	3	Dragomir Mihai
4	3	Popescu Mihai
5	4	Olaru Cristina
6	4	Dinu Alina

- 7) Sa se afiseze toti clientii care nu au achitat ratele creditelor bancare, precum si suma totala a creditului aferent.

```
SELECT c.id_client, c.nume_client, c.prenume_client, COUNT(r.id_rata) AS numar_rate_neplatite,
SUM(r.suma_rata) AS suma_totala_neplatite
FROM clienti_PROIECT c
JOIN credite_PROIECT cr ON c.id_client = cr.id_client
JOIN rate_credit_bancare_PROIECT2 r ON cr.id_credit = r.id_credit
WHERE r.status_rata = 'Nechitat'
GROUP BY c.id_client, c.nume_client, c.prenume_client;
SELECT USER FROM DUAL;
```

```
SUM(r.suma_rata) AS suma_totala_neplatite
FROM clienti_PROJECT c
JOIN credite_PROJECT cr ON c.id_client = cr.id_client
JOIN rate_credit_bancare_PROJECT2 r ON cr.id_credit = r.id_credit
WHERE r.status_rata = 'Nechitat'
GROUP BY c.id_client, c.nume_client, c.prenume_client;
```

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	NUMAR_RATE_NEPLATITE	SUMA_TOTALA_NEPLATITE
1	1000 Simon	Georgiana	3	3400
2	1001 Ilinca	Lucian	3	6050
3	1002 Marinca	Ana	2	3100
4	1003 Dumitru	Stefan	3	4400
5	1004 Marin	Ioana	2	3700

- 8) Sa se calculeze rata medie lunara pentru fiecare client, pe baza sumelor de rate.

```
SELECT c.id_client, c.nume_client, c.prenume_client, AVG(r.suma_rata) AS rata_medie_lunara
FROM clienti_proiect c
JOIN credite_proiect cr ON c.id_client = cr.id_client
JOIN rate_credit_bancare_PROIECT2 r ON cr.id_credit = r.id_credit
WHERE r.status_rata != 'Achitata'
GROUP BY c.id_client, c.nume_client, c.prenume_client;
SELECT USER FROM DUAL;
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

|JOIN rate_credit_bancare_PROIECT2 cr ON cr.id_credit = r.id_cr
WHERE r.status_rata != 'Achitata'
GROUP BY c.id_client, c.nume_client, c.prenume_client;
SELECT USER FROM DUAL;

| WHERE r.status_rata != 'Ac'
GROUP BY c.id_client, c.nume_client, c.prenume_client;
SELECT USER FROM DUAL;

```

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	RATA_MEDIE_LUNARA	
1	1000	Simon	Georgiana	1225
2	1001	Ilinca	Lucian	1775
3	1002	Marinca	Ana	1550
4	1003	Dumitru	Stefan	1500
5	1004	Marin	Ioana	1850

9) Sa se afiseze clientii care au contractat credite, impreuna cu suma totala a creditelor lor. In functie de suma totala a creditelor, se vor clasifica astfel: "Imprumut mare" pentru sume mai mari de 30.000 RON, "Imprumut mediu" pentru sume intre 20.000 AND 30.000 RON si "Imprumut mic" pentru sume mai mici de 20.000 RON. Rezultatele trebuie ordonate descrescator in functie de suma totala a creditelor. `SELECT c.id_client, c.nume_client, c.prenume_client, SUM(cr.suma_credit) AS suma_totala_credit,`

CASE

```

WHEN SUM(cr.suma_credit) > 30000 THEN 'Imprumut mare'
WHEN SUM(cr.suma_credit) BETWEEN 20000 AND 30000 THEN 'Imprumut mediu'
ELSE 'Imprumut mic'
END AS tip_credit
FROM clienti_proiect c
JOIN credite_proiect cr ON c.id_client = cr.id_client
GROUP BY c.id_client, c.nume_client, c.prenume_client
ORDER BY suma_totala_credit DESC;
SELECT USER FROM DUAL;

```

```

| JOIN credite_proiect cr ON c.id_client = cr.id_client
| GROUP BY c.id_client, c.nume_client, c.prenume_client
| ORDER BY suma_totala_credit DESC;
| SELECT USER FROM DUAL;

| GROUP BY c.id_client, c.nume_client
| ORDER BY suma_totala_credit DESC;
| SELECT USER FROM DUAL;

```

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	SUMA_TOTALA_CREDIT	TIP_CREDIT
1	1008	Popa	Adriana	34000 Imprumut mare
2	1002	Marinca	Ana	28000 Imprumut mediu
3	1007	Vasiliu	Daniel	25000 Imprumut mediu
4	1005	Radu	Mihai	25000 Imprumut mediu
5	1009	Freda	Andrei	24000 Imprumut mediu
6	1004	Marin	Ioana	24000 Imprumut mediu
7	1001	Ilinca	Lucian	22000 Imprumut mediu
8	1006	Stoica	Elena	20500 Imprumut mediu
9	1003	Dumitru	Stefan	18000 Imprumut mic
10	1000	Simon	Georgiana	15000 Imprumut mic

10)Sa se afiseze toti angajatiii, impreuna cu denumirea bancii la care sunt angajati si salariul lor. Sa se selecteze doar angajati care au cel mai mare salariu din fiecare banca. Rezultatele trebuie sa fie ordonate descrescator in functie de salariu.

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire AS banca, a.salariul
FROM angajati_proiect a
JOIN banca_proiect b ON a.id_banca = b.id_banca WHERE a.salariul = (SELECT MAX(salariul) FROM
angajati_proiect WHERE id_banca = a.id_banca)
ORDER BY a.salariul DESC;
SELECT USER FROM DUAL;

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

SELECT * FROM angajati WHERE id_banca = a.id_banca
ORDER BY a.salariul DESC;
SELECT USER FROM DUAL;

```

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	BANCA	SALARIUL
1	2105 Popescu	Florin	ING Bank	9200
2	2095 Miron	Valeriu	Unicredit	9100
3	2082 Gheorghiu	Adrian	Alpha Bank	8900
4	2069 Tudor	Ioan	CEC Bank	8700
5	2057 Marin	Ioan	OTP Bank	8600
6	2000 Clobanu	Marian	Raiffeisen Bank	8500
7	2044 Lazar	Ion	Banca Transilvania	8500
8	2035 Ilie	Alexandra	BRD	8000
9	2025 Sandu	Alexandru	BCR	7800
10	2014 Toma	Gabriela	Revolut	7600

11) Sa se afiseze toti clientii care au efectuat transferuri pentru conturile lor bancare, impreuna cu ultima data a transferului. Sa se selecteze doar clientii care nu au efectuat niciun transfer sau au efectuat un transfer inainte de 8 ianuarie 2025.

```

SELECT c.id_client, c.nume_client, c.prenume_client, MAX(t.data_transfer) AS ultima_data_transfer
FROM conturi_proiect co
JOIN transferuri_proiect t ON co.id_cont = t.id_cont
JOIN clienti_PROIECT c ON c.id_client=co.id_client
GROUP BY c.id_client, c.nume_client, c.prenume_client
HAVING MAX(t.data_transfer) IS NULL OR MAX(t.data_transfer) < TO_DATE('08-JAN-25',
'DDMON-YY');
SELECT USER FROM DUAL;

```

```

SELECT * FROM conturi_proiect WHERE EXTRACT(DAY FROM t.data_transfer) IS NULL OR MAX(t.data_transfer) < TO_DATE('08-JAN-25', 'DDMON-YY');
SELECT USER FROM DUAL;

```

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	ULTIMA_DATA_TRANSFER
1	1000 Simon	Georgiana	05-DEC-24
2	1001 Ilinca	Lucian	10-DEC-24
3	1002 Marinca	Ana	14-DEC-24
4	1003 Dumitru	Stefan	18-DEC-24
5	1004 Marin	Ioana	22-DEC-24
6	1005 Radu	Mihai	26-DEC-24
7	1006 Stoica	Elena	30-DEC-24
8	1007 Vasiliu	Daniel	03-JAN-25
9	1008 Popa	Adriana	07-JAN-25

12) Sa se actualizeze soldul conturilor din tabela conturi_proiect in functie de numarul de tranzactii efectuate in prima si a doua jumata a lunii. Daca un cont are mai mult de 3 tranzactii in prima jumata a lunii, soldul va fi redus cu 2%. Daca are mai mult de 3 tranzactii in a doua jumata a lunii, soldul va fi redus cu 3%.

```

SELECT * FROM tranzactii_PROIECT;
UPDATE conturi_proiect c
SET c.sold = c.sold *
CASE
WHEN c.id_cont IN
(SELECT t.id_cont FROM tranzactii_proiect t
WHERE EXTRACT(DAY FROM t.data_tranzactiei) <= 15
GROUP BY t.id_cont
HAVING COUNT(*) >= 3) THEN 0.98
WHEN c.id_cont IN
(SELECT t.id_cont FROM tranzactii_proiect t
WHERE EXTRACT(DAY FROM t.data_tranzactiei) > 15
GROUP BY t.id_cont

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
HAVING COUNT(*) >= 3) THEN 0.97
ELSE 1
END;
SELECT id_client, sold FROM conturi_PROIECT
WHERE id_client IN (1001, 1005);
COMMIT;
SELECT USER FROM DUAL;
```

The screenshot shows the Oracle SQL Developer interface with three tabs:

- Script Output:** Contains the PL/SQL code provided above.
- Query Result:** Shows the output of the SELECT statement:

ID_CLIENT	SOLD
1001	83300.74
1005	7760

 A note below says "31 rows updated."
- Script Output:** Shows the output of the final SELECT USER FROM DUAL statement:

USER
SIMONG_65

- 13) Sa se actualizeze dobanda pentru creditele clientilor care au efectuat tranzactii in luna anterioara in valoare totala mai mare de 7000 RON. Dobanda se va reduce cu 5%. Sa se afiseze si ID-urile clientilor care indeplinesc aceasta conditie.

```
UPDATE credite_proiect c
SET c.dobanda = c.dobanda * 0.95 WHERE c.id_client IN
(SELECT t.id_client FROM tranzactii_proiect t
WHERE EXTRACT(MONTH FROM t.data_tranzactiei) = EXTRACT(MONTH FROM
ADD_MONTHS(SYSDATE, -1))
AND EXTRACT(YEAR FROM t.data_tranzactiei) = EXTRACT(YEAR FROM
ADD_MONTHS(SYSDATE, -1))
GROUP BY t.id_client
HAVING SUM(t.suma_tranzactiei) > 7000);
SELECT t.id_client
FROM tranzactii_proiect t
WHERE EXTRACT(MONTH FROM t.data_tranzactiei) = EXTRACT(MONTH FROM
ADD_MONTHS(SYSDATE, -1))
AND EXTRACT(YEAR FROM t.data_tranzactiei) = EXTRACT(YEAR FROM
ADD_MONTHS(SYSDATE, -1))
GROUP BY t.id_client
HAVING SUM(t.suma_tranzactiei) > 7000;
SELECT USER FROM DUAL;
COMMIT;
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

The screenshot shows three separate SQL queries run in Oracle SQL Developer:

- Query 1:** A complex query involving multiple joins and subqueries to find clients who have made transfers in 2024 with a total value over 7000 RON. It uses EXTRACT functions to filter by year and month.
- Query 2:** A simple query to select all users from the DUAL table.
- Query 3:** A query to select all rows from the transferuri_PROIECT table where the ID_CLIENT values are 1003 or 1005.

Below the queries, the results are displayed in tables:

ID_CLIENT	1	1003
	2	1005

7	700007	12000	5.04	10-JUL-24	10-JUL-29	1003
8	700008	6000	4.37	05-AUG-24	05-AUG-29	1003
11	700011	9000	5.32	01-NOV-24	01-NOV-29	1005
12	700012	16000	4.94	01-DEC-24	01-DEC-30	1005

USER	1	SIMONG_65
------	---	-----------

ID_CLIENT	ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	SUMA_TOTALA_TRANSFERATA
1	1006	Stoica	Elena	15500
2	1005	Radu	Mihai	13500
3	1004	Marin	Ioana	11500
4	1003	Dumitru	Stefan	9500

15) Sa se actualizeze suma creditului pentru clientii care au realizat transferuri bancare mai mari de 5000 RON. Suma creditului va fi marita cu 1000 RON pentru acei clienti care au efectuat transferuri de aceasta valoare. Sa se afiseze, de asemenea, si conturile si clientii care au realizat transferuri mai mari de 5000 RON.

```

UPDATE credite_PROIECT c
SET c.suma_credit = c.suma_credit + 1000
WHERE c.id_client IN (SELECT co.id_client FROM transferuri_PROIECT t
JOIN conturi_PROIECT co ON t.id_cont = co.id_cont
WHERE t.suma_transferata > 5000);
SELECT t.id_cont, co.id_client
FROM transferuri_PROIECT t
JOIN conturi_PROIECT co ON t.id_cont = co.id_cont
WHERE t.suma_transferata > 5000;
SELECT t.id_cont, co.id_client
FROM transferuri_PROIECT t
JOIN conturi_PROIECT co ON t.id_cont = co.id_cont

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

WHERE t.suma_transferata > 5000;
 SELECT USER FROM DUAL;

```

SELECT t.id_cont, co.id_client
FROM transferuri_PROJECT t
JOIN conturi_PROJECT co ON t.id_cont = co.id_cont
WHERE t.suma_transferata > 5000;
    
```

ID_CONT	ID_CLIENT
1	1004
2	1004
3	1005
4	1005
5	1006
6	1006
7	1007
8	1007
9	1008

```

SELECT USER FROM DUAL;
    
```

USER
SIMONG_65

16) Sa se actualizeze salariul angajatilor care sunt manageri ai angajatilor cu varsta sub 30 de ani si cu salarii mai mici de 4000, astfel incat salariul acestora va fi marit cu 10%. Sa se afiseze id-urile managerilor care indeplinesc aceste conditii.

UPDATE angajati_PROJECT a

SET a.salariul = a.salariul * 1.10

WHERE NVL(a.id_angajat, 0) IN (

SELECT DISTINCT NVL(a2.id_manager, 0)

FROM angajati_PROJECT a1

JOIN angajati_PROJECT a2 ON a1.id_manager = a2.id_angajat

WHERE a1.varsta < 30 AND a1.salariul < 4000);

SELECT DISTINCT NVL(a2.id_manager, 0)

FROM angajati_PROJECT a1

JOIN angajati_PROJECT a2 ON a1.id_manager = a2.id_angajat

WHERE a1.varsta < 30 AND a1.salariul < 4000;

SELECT USER FROM DUAL;

```

UPDATE angajati_PROJECT a
SET a.salariul = a.salariul * 1.10
WHERE NVL(a.id_angajat, 0) IN (
    SELECT DISTINCT NVL(a2.id_manager, 0)
    FROM angajati_PROJECT a1
    JOIN angajati_PROJECT a2 ON a1.id_manager = a2.id_angajat
    WHERE a1.varsta < 30 AND a1.salariul < 4000);
    
```

```

    
```

Task completed in 0.044 seconds

2 rows updated.

NVL(A2.ID_MANAGER, 0)
0
2000
2004

```

UPDATE angajati_PROJECT a
SET a.salariul = a.salariul * 1.10
WHERE NVL(a.id_angajat, 0) IN (
    SELECT DISTINCT NVL(a2.id_manager, 0)
    FROM angajati_PROJECT a1
    JOIN angajati_PROJECT a2 ON a1.id_manager = a2.id_angajat
    WHERE a1.varsta < 30 AND a1.salariul < 4000);
    
```

```

    
```

SELECT DISTINCT NVL(a2.id_manager, 0)

FROM angajati_PROJECT a1

JOIN angajati_PROJECT a2 ON a1.id_manager = a2.id_angajat

WHERE a1.varsta < 30 AND a1.salariul < 4000;

SELECT USER FROM DUAL;

1 SIMONG_65

17) Sa se creeze o tabela virtuala pentru clientii care au plătit toate ratele.(statusul ratei='Achitat')

CREATE OR REPLACE VIEW v_clienti_rate_achitate AS

SELECT c.id_client, c.nume_client, c.prenume_client FROM clienti_project c

JOIN credite_proiect cr ON c.id_client=cr.id_client

JOIN rate_credit_bancare_proiect r ON cr.id_credit = r.id_credit

GROUP BY c.id_client, c.nume_client, c.prenume_client

HAVING COUNT(CASE WHEN r.status_rata = 'Achitat' THEN 1 END)>=1;

SELECT *FROM v_clienti_rate_achitate;

SELECT USER FROM DUAL;

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

SELECT c.id_client,
       c.num_client,
       c.prenume_client
  FROM clienti_project c
 JOIN credite_proiect cr ON c.id_client=cr.id_client
 JOIN rate_credit_bancare_project2 r ON cr.id_credit = r.id_credit
 GROUP BY c.id_client, c.num_client, c.prenume_client
 HAVING COUNT(CASE WHEN r.status_rata = 'Achitat' THEN 1 END)>=1;
 SELECT *FROM v_clienti_rate_achitate;

-- CREATE OR REPLACE VIEW v_clienti_rate_achitate
AS
SELECT c.id_client,
       c.num_client,
       c.prenume_client
  FROM clienti_project c
 JOIN credite_proiect cr ON c.id_client=cr.id_client
 JOIN rate_credit_bancare_project2 r ON cr.id_credit = r.id_credit
 GROUP BY c.id_client, c.num_client, c.prenume_client
 HAVING COUNT(CASE WHEN r.status_rata = 'Achitat' THEN 1 END)>=1;
SELECT *FROM v_clienti_rate_achitate;

SELECT * FROM v_clienti_rate_achitate;
SELECT USER FROM DUAL;
  
```

View V_CLIENTI_RATE_ACHITATE created.

18) Sa se actualizeze dobanda pentru creditele care au rate neachitate si cu data scadenta in perioada februarie - august. Dobanda acestora va fi marita cu 50%.

```

UPDATE credite_PROJECT c
SET c.dobanda = c.dobanda * 1.5
WHERE c.id_credit IN (SELECT r.id_credit FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Nearchitat' AND r.data_scadenta < SYSDATE
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 2 AND 8);
SELECT r.id_credit
FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Nearchitat'
AND r.data_scadenta < SYSDATE
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 2 AND 8;
SELECT USER FROM DUAL;
  
```

(7 randuri)

```

UPDATE credite_PROJECT c
SET c.dobanda = c.dobanda * 1.5
WHERE c.id_credit IN (
SELECT r.id_credit
FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Nearchitat'
AND r.data_scadenta < SYSDATE
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 2 AND 8);

2 rows updated.

7 rows updated.

-- SELECT r.id_credit
-- FROM rate_credit_bancare_PROJECT2 r
-- WHERE r.status_rata = 'Nearchitat'
-- AND r.data_scadenta < SYSDATE
-- AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 2 AND 8;

-- SELECT USER FROM DUAL;
  
```

19) Sa se actualizeze dobanda la 7 RON (in coloana, am ales sa pastrez o valoare absoluta, nu procent) pentru creditele care au toate ratele achitate si au data scadenta in perioada martie - iulie.

```

UPDATE credite_PROJECT c
SET c.dobanda = 7
WHERE c.id_credit IN ( SELECT r.id_credit FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Achitat'
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 3 AND 7);
SELECT r.id_credit
FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Achitat'
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 3 AND 7;
SELECT USER FROM DUAL;
  
```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

UPDATE credite_PROJECT c
SET c.dobanda = 7
WHERE c.id_credit IN (
SELECT r.id_credit
FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Achitata'
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 3 AND 7);

SELECT r.id_credit
FROM rate_credit_bancare_PROJECT2 r
WHERE r.status_rata = 'Achitata'
AND EXTRACT(MONTH FROM r.data_scadenta) BETWEEN 3 AND 7;

SELECT USER FROM DUAL;

```

Script Output | Query Result | All Rows Fetched: 3 in 0.017 seconds

ID_CREDIT
1 700002
2 700003
3 700005

Script Output | Query Result | All Rows Fetched: 1 in 0.013 seconds

USER
1 SIMONG_65

```

CONNECT BY PRIOR a.id_angajat = a.id_manager
START WITH a.id_manager IS NULL
ORDER BY NIVEL_IERARHIC, a.salariul DESC;
SELECT USER FROM DUAL;

```

Script Output | Query Result | All Rows Fetched: 1 in 0.013 seconds

USER
1 SIMONG_65

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

21) Sa se afiseze managerii cu varsta peste 30 de ani care au cel putin 3 subordonati cu salarii mai mici de 7000, sau fara subordonati. Se va afisa nivelul ierarhic, numele managerului, numarul de subordonati, salariul total al subordonatilor si salariul managerului. Managerii sunt ordonati in functie de nivelul ierarhic si de salariul total al subordonatilor, in ordine descrescatoare.

```
SELECT LEVEL AS NIVEL_IERARHIC, a.nume_angajat || ' ' || a.prenume_angajat AS MANAGER,
COUNT(DISTINCT sub.id_angajat) AS NUMAR_SUBORDONATI,
SUM(NVL(sub.salariul, 0)) AS SALARIU_TOTAL_SUBORDONATI, a.salariul AS
SALARIU_MANAGER
FROM angajati_PROIECT a
LEFT JOIN angajati_PROIECT sub ON sub.id_manager = a.id_angajat
WHERE a.varsta > 30 -- doar managerii cu varsta peste 30 de ani
AND (sub.salariul IS NULL OR sub.salariul < 7000)
CONNECT BY PRIOR a.id_angajat = a.id_manager
START WITH a.id_manager IS NULL
GROUP BY LEVEL, a.nume_angajat, a.prenume_angajat, a.salariul
HAVING COUNT(DISTINCT sub.id_angajat) > 2
ORDER BY NIVEL_IERARHIC, SALARIU_TOTAL_SUBORDONATI DESC;
SELECT USER FROM DUAL;
```

```
ORDER BY NIVEL_IERARHIC, SALARIU_MANAGER  
SELECT USER FROM DUAL;
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.026 seconds

USER
1 SIMONG_65

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0.026 seconds

NIVEL_IERARHIC	MANAGER	NUMAR_SUBORDONATI	SALARIU_TOTAL_SUBORDONATI	SALARIU_MANAGER
1	Ilie Alexandra	3	18100	8000
2	I.Toma Gabriela	3	17150	7600
3	I.Clobanu Marian	4	14850	8500

22)Sa se realizeze o analiza a clientilor, in functie de activitatea lor financiara (tranzactiile efectuate si suma acestora) realizata in anul trecut. In functie de numar de tranzacti, daca este mai mare sau egal cu 3 si se indeplineste simultan conditia ca suma sa fie mai mare de 1000 RON, se va afisa ‘Activitate intensa’

PROIECT – BAZE DE DATE – SIMON GEORGIANA

ca tip de comportament. Daca numarul de tranzactii se afla intre 1 si 3 si suma acestora este mai mica de 1000 RON se va afisa ‘Moderata’, altfel se va afisa ‘Normal’.

```

SELECT c.id_client, c.nume_client || ' ' || c.prenume_client AS nume_complet, COUNT(t.id_tranzactie)
AS numar_tranzactii,
SUM(t.suma_tranzactiei) AS suma_totala,
CASE
WHEN COUNT(t.id_tranzactie) >= 3
AND SUM(t.suma_tranzactiei) > 10000
THEN 'Activitate intensiva'
WHEN COUNT(t.id_tranzactie) BETWEEN 1 AND 3
AND SUM(t.suma_tranzactiei) < 1000
THEN 'Activitate moderata'
ELSE 'Normal'
END AS tip_comportament
FROM clienti_proiect c
JOIN conturi_proiect co ON c.id_client = co.id_client
JOIN tranzactii_proiect t ON co.id_cont = t.id_cont
WHERE EXTRACT(YEAR FROM t.data_tranzactiei) = 2024
GROUP BY c.id_client, c.nume_client, c.prenume_client
ORDER BY suma_totala DESC;
SELECT USER FROM DUAL;
```

```

Query Builder
SELECT c.id_client, c.nume_client || ' ' || c.prenume_client AS nume_complet, COUNT(t.id_tranzactie) AS numar_tranzactii,
SUM(t.suma_tranzactiei) AS suma_totala,
CASE
WHEN COUNT(t.id_tranzactie) >= 3
AND SUM(t.suma_tranzactiei) > 10000
THEN 'Activitate intensiva'
WHEN COUNT(t.id_tranzactie) BETWEEN 1 AND 3
AND SUM(t.suma_tranzactiei) < 1000
THEN 'Activitate moderata'
ELSE 'Normal'
END AS tip_comportament
FROM clienti_proiect c
JOIN conturi_proiect co ON c.id_client = co.id_client
JOIN tranzactii_proiect t ON co.id_cont = t.id_cont
WHERE EXTRACT(YEAR FROM t.data_tranzactiei) = 2024
GROUP BY c.id_client, c.nume_client, c.prenume_client
ORDER BY suma_totala DESC;
SELECT USER FROM DUAL;
```

```

Query Result
SQL | All Rows Fetched: 6 in 0.198 seconds
ID_CLIENT | NUME_COMPLET | NUMAR_TRANZACTII | SUMA_TOTALA | TIP_COMPORTAMENT
1 1005 Radu Mihai | 4 | 25000 | Activitate intensiva
2 1003 Dumitru Stefan | 4 | 19000 | Activitate intensiva
3 1002 Marin Ioana | 2 | 6500 | Normal
4 1004 Marin Ioana | 1 | 6000 | Normal
5 1001 Ilince Lucian | 3 | 6000 | Normal
6 1000 Simon Georgiana | 1 | 1200 | Normal
```

```

Query Result
SQL | All Rows Fetched: 1 in 0.094 seconds
USER
1 SIMONG_65
```

23)Sa se realizeze o analiza asupra tipurilor de tranzactii, in functie de luna si de dimensiunea acestora, daca s-au realizat tranzactii de maxim 1000 RON, atunci tipul de tranzactie va face parte din categoria ‘MICA’, daca s-au realizat tranzactii intre 1000 si 5000 RON, atunci e ava face parte din categoria ‘MEDIE’, altfel, va face parte din categoria ‘MARE’. Sa se ordoneze in functie de luna si suma totala in ordine descrescatoare.

```

SELECT EXTRACT(MONTH FROM t.data_tranzactiei) AS LUNA,
CASE
WHEN t.suma_tranzactiei < 1000 THEN 'MICĂ'
WHEN t.suma_tranzactiei BETWEEN 1000 AND 5000 THEN 'MEDIE'
ELSE 'MARE'
END AS CATEGORIE,
COUNT(t.id_tranzactie) AS NUMAR_TRANZACTII,
SUM(t.suma_tranzactiei) AS SUMA_TOTALA
```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```

FROM tranzactii_proiect t
GROUP BY EXTRACT(MONTH FROM t.data_tranzactiei),
CASE
WHEN t.suma_tranzactiei < 1000 THEN 'MICĂ'
WHEN t.suma_tranzactiei BETWEEN 1000 AND 5000 THEN 'MEDIE'
ELSE 'MARE'
END
ORDER BY LUNA, SUMA_TOTALA DESC;
SELECT USER FROM DUAL;

```

```

SELECT EXTRACT(MONTH FROM t.data_tranzactiei) AS LUNA,
CASE
WHEN t.suma_tranzactiei < 1000 THEN 'MICĂ'
WHEN t.suma_tranzactiei BETWEEN 1000 AND 5000 THEN 'MEDIE'
ELSE 'MARE'
END AS CATEGORIE,
COUNT(t.id_tranzactie) AS NUMAR_TRANZACTII,
SUM(t.suma_tranzactiei) AS SUMA_TOTALA
FROM tranzactii_proiect t
GROUP BY EXTRACT(MONTH FROM t.data_tranzactiei),
CASE
WHEN t.suma_tranzactiei < 1000 THEN 'MICĂ'
WHEN t.suma_tranzactiei BETWEEN 1000 AND 5000 THEN 'MEDIE'
ELSE 'MARE'
END
ORDER BY LUNA, SUMA_TOTALA DESC;
SELECT USER FROM DUAL;

```

LUNA	CATEGORIE	NUMAR_TRANZACTII	SUMA_TOTALA
1	MARE	3	27000
2	MEDIE	2	2050
3	MARE	6	40500
4	MEDIE	9	27200


```

SELECT USER

```

USER
SIMONG_65

24) Sa se realizeze un raport detaliat ce vizeaza activitatea bancilor, in functie de creditele acordate clientilor, doar in cazul in care statutul creditelor este ‘Achitat’.

```

SELECT b.denumire,
COUNT(DISTINCT c.id_client) AS NUMAR_CLIENTI,
COUNT(DISTINCT cr.id_credit) AS NUMAR_CREDITE,
SUM(cr.suma_credit) AS TOTAL_CREDITE,
SUM(NVL((SELECT SUM(r.suma_rata)
FROM rate_credit_bancare_proiect2 r
WHERE r.id_credit = cr.id_credit AND r.status_rata = 'Achitat'), 0)) AS TOTAL_RATE_ACHITATE
FROM banca_proiect b
JOIN banca_clienti_proiect c ON b.id_banca = c.id_banca
JOIN credite_proiect cr ON c.id_client = cr.id_client
GROUP BY b.denumire
ORDER BY TOTAL_CREDITE DESC;
SELECT USER FROM DUAL;

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

SELECT b.denumire,
       COUNT(DISTINCT c.id_client) AS NUMAR_CLIENTI,
       COUNT(DISTINCT cr.id_credit) AS NUMAR_CREDITE,
       SUM(cr.suma_credit) AS TOTAL_CREDITE,
       SUM(NVL((SELECT SUM(r.suma_rata)
FROM rate_credit_bancare_proiect r
WHERE r.id_credit = cr.id_credit AND r.status_rata = 'Achitat'), 0)) AS TOTAL_RATE_ACHITATE
FROM banca_proiect b
JOIN banca_clienti_proiect c ON b.id_banca = c.id_banca
JOIN credite_proiect cr ON c.id_client = cr.id_client
GROUP BY b.denumire
ORDER BY TOTAL_CREDITE DESC;
SELECT USER FROM DUAL;

Script Output | Query Result
SQL | All Rows Fetched: 5 in 0.026 seconds
| DENUMIRE | NUMAR_CLIENTI | NUMAR_CREDITE | TOTAL_CREDITE | TOTAL_RATE_ACHITATE |
| 1 BRD | 2 | 4 | 55000 | 3100 |
| 2 Revolut | 2 | 4 | 54000 | 1600 |
| 3 Banca Transilvania | 2 | 4 | 50000 | 3700 |
| 4 BCR | 2 | 4 | 44500 | 1050 |
| 5 Raiffeisen Bank | 2 | 4 | 42000 | 1500 |

SELECT b.denumire,
       COUNT(DISTINCT c.id_client) AS NUMAR_CLIENTI,
       COUNT(DISTINCT cr.id_credit) AS NUMAR_CREDITE,
       SUM(cr.suma_credit) AS TOTAL_CREDITE,
       SUM(NVL((SELECT SUM(r.suma_rata)
FROM rate_credit_bancare_proiect2 r
WHERE r.id_credit = cr.id_credit AND r.status_rata = 'Achitat'), 0)) AS TOTAL_RATE_ACHITATE
FROM banca_proiect b
JOIN banca_clienti_proiect c ON b.id_banca = c.id_banca
JOIN credite_proiect cr ON c.id_client = cr.id_client
GROUP BY b.denumire
ORDER BY TOTAL_CREDITE DESC;
SELECT USER FROM DUAL;

```

Script Output | Query Result
SQL | All Rows Fetched: 1 in 0.015 seconds
USER | 1 SIMONG_65

25) Sa se acorde un bonus la salariul angajatiilor, pe baza salariului mediu din departamentele in care lucreaza. Si sa afiseze angajatii carora li s-au acordat bonusuri cele mai mari.

ALTER TABLE angajati_PROIECT

MODIFY salariul NUMBER(8, 2);

UPDATE angajati_proiect a

SET a.salariul = a.salariul +

CASE

WHEN a.id_departamente IN (SELECT d.id_departamente FROM departamente_proiect d

JOIN angajati_proiect ang ON d.id_departamente = ang.id_departamente

GROUP BY d.id_departamente

HAVING AVG(ang.salariul) > 7000) THEN 700

ELSE 300

END;

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, d.denumire_departament FROM angajati_proiect a

JOIN departamente_proiect d ON a.id_departamente = d.id_departamente

WHERE a.id_departamente IN (SELECT d.id_departamente

FROM departamente_proiect d

JOIN angajati_proiect ang ON d.id_departamente = ang.id_departamente

GROUP BY d.id_departamente

HAVING AVG(ang.salariul) > 7000)

ORDER BY a.salariul DESC;

SELECT USER FROM DUAL;

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

ALTER TABLE angajati_PROJECT
MODIFY salariul NUMBER(5, 2);
UPDATE angajati_project a
SET a.salariul = a.salariul +
50;
END;
SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, d.denumire_departament
FROM angajati_project a
JOIN departamente_project d ON a.id_departamente = d.id_departamente
WHERE a.id_departamente IN (
SELECT d.id_departamente
FROM departamente_project d
GROUP BY d.id_departamente
HAVING AVG(ang.salariul) > 7000)
ORDER BY a.salariul DESC;
SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, d.denumire_departament
FROM angajati_project a
JOIN departamente_project d ON a.id_departamente = d.id_departamente
WHERE a.id_departamente IN (SELECT d.id_departamente
FROM departamente_project d
JOIN angajati_project ang ON d.id_departamente = ang.id_departamente
GROUP BY d.id_departamente
HAVING AVG(ang.salariul) > 7000)
ORDER BY a.salariul DESC;

```

Script Output | Query Result | Task completed in 0.06 seconds

Table ANGAJATI_PROJECT altered.

115 rows updated.

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIUL	DENUMIRE_DEPARTAMENT
55	2062 Rusu	Mirela	6400	Credite/Imprumuturi
56	2021 Ion	Andreea	6225	Marketing
57	2019 Dima	Alina	6200	IT si suport tehnic
58	2051 Neacsu	Florin	6200	Resurse umane
59	2054 Barbu	Elena	6200	Departament juridic
60	2029 Dragomir	Gabriel	5900	Credite/Imprumuturi
61	2050 Vasile	Mihai	5900	Resurse umane
62	2063 Popescu	Andrei	5900	Credite/Imprumuturi
63	2055 Gheorghiu	George	5800	Departament juridic
64	2056 Constantin	Irina	5700	Departament juridic

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, d.denumire_departament
FROM angajati_project a
JOIN departamente_project d ON a.id_departamente = d.id_departamente
WHERE a.id_departamente IN (SELECT d.id_departamente
FROM departamente_project d
JOIN angajati_project ang ON d.id_departamente = ang.id_departamente
GROUP BY d.id_departamente
HAVING AVG(ang.salariul) > 7000)
ORDER BY a.salariul DESC;
SELECT USER FROM DUAL;

```

Script Output | Query Result | Task completed in 0.028 seconds

USER

1 SIMONG_65

26) Sa se actualizeze functia angajatiilor(e reprezentata de idea de a fi manager), astfel incat, daca salariul unui angajat este sub media departamentului, i se atribuie un nou manager.

```

UPDATE angajati_project a
SET a.id_manager = (SELECT m.id_angajat
FROM angajati_project m
WHERE m.id_departamente = a.id_departamente
AND m.salariul = (SELECT MAX(salariul)
FROM angajati_project
WHERE id_departamente = a.id_departamente))
WHERE a.salariul < (SELECT AVG(salariul)
FROM angajati_project
WHERE id_departamente = a.id_departamente);
SELECT USER FROM DUAL;

```

```

UPDATE angajati_project a
SET a.id_manager = (SELECT m.id_angajat
FROM angajati_project m
WHERE m.id_departamente = a.id_departamente
AND m.salariul = (SELECT MAX(salariul)
FROM angajati_project
WHERE id_departamente = a.id_departamente))
WHERE a.salariul < (SELECT AVG(salariul)
FROM angajati_project
WHERE id_departamente = a.id_departamente);
SELECT USER FROM DUAL;

```

Script Output | Query Result | Task completed in 0.016 seconds

USER

1 SIMONG_65

```

UPDATE angajati_project a
SET a.id_manager = (SELECT m.id_angajat
FROM angajati_project m
WHERE m.id_departamente = a.id_departamente
AND m.salariul = (SELECT MAX(salariul)
FROM angajati_project
WHERE id_departamente = a.id_departamente))
WHERE a.salariul < (SELECT AVG(salariul)
FROM angajati_project
WHERE id_departamente = a.id_departamente);
SELECT USER FROM DUAL;

```

Script Output | Query Result | Task completed in 0.05 seconds

Table ANGAJATI_PROJECT altered.

115 rows updated.

59 rows updated.

27) Sa se afiseze angajatii cu cele mai mari salarii din fiecare departament.

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, d.denumire_departament,
(SELECT AVG(salariul) FROM angajati_project

```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```
WHERE id_departamente = a.id_departamente) AS salariu_mediu_departament  
FROM angajati_proiect a  
JOIN departamente_proiect d ON a.id_departamente = d.id_departamente  
WHERE a.salariul = (SELECT MAX(salariul)  
FROM angajati_proiect  
WHERE id_departamente = a.id_departamente)  
ORDER BY d.denumire_departament, a.salariul DESC;  
SELECT USER FROM DUAL;
```

29)Sa se actualizeze salariilor angajatilor care lucreaza in bancile Raiffeisen si Revolut, oferind un bonus in functie de numarul de tranzactii realizate de clientii acestora. Daca un client a facut intre 1 si 3 tranzactii, angajatul va primi un bonus de 500, iar daca a facut mai mult de 3 tranzactii, angajatul va primi un bonus de 1000. UPDATE angajati_proiect a

SET a.salariul = a.salariul +

CASE

WHEN a.id_banca IN (SELECT bc.id_banca FROM banca clienti proiect bc

JOIN clienti project c ON bc.id_client = c.id_client

JOIN tranzactii project t ON c.id = t.id client

JOIN banca project b ON bc.id = banca.b_id

WHERE b.denumire IN ('Raiffeisen', 'Revolut')

GROUP BY bc_id, banca, c_id, client

HAVING COUNT(t.id_tranzactie) BETWEEN 1 AND 3) THEN 500

WHEN a.id_banca IN (SELECT bc.id_banca FROM banca_clienti project bc

```
WHERE client_id = ?; (SELECT client_id FROM  
JOIN clienti project c ON bc.id_client = c.id_client
```

JOIN clienti_proiecte ON sc.id_client = c.id_client
JOIN tranzactii_proiecte ON s.id_client = t.id_client

JOIN transaction_project t ON t.id_client = b.id_client

```
JOIN banea_proiect b ON b.id_banea = b.id_t  
WHERE b.denumire IN ('Raiffeisen', 'Revolut')
```

WHERE b.dealerid IN (Rahnsen, GROUBBY, hanse, aid, client)

HAVING COUNT(transaction) > 2) TIEN 1000

HAVING COUNT(l.id_transaksi) > 3) THEN 1000
ELSE 0

ELSE 0
END

WHERE i1.i1 = D1.(SELECT i1

WHERE a.id_banca IN (SELECT id_banca FROM b WHERE b.id_cassa = a.id_cassa)

WHERE

SELH

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```


-- UPDATE angajati_proiect a
SET a.salariu = a.salariu +
CASE
    WHEN a.id_banca IN (
        SELECT bc.id_banca
        FROM banca_clienti_proiect bc
        JOIN clienti_proiect c ON bc.id_client = c.id_client
        JOIN tranzactii_proiect t ON c.id_client = t.id_client
        JOIN banca_proiect b ON bc.id_banca = b.id_banca
        WHERE b.denumire IN ('Raiffeisen', 'Revolut')
        GROUP BY bc.id_banca, c.id_client
        HAVING COUNT(t.id_tranzactie) BETWEEN 1 AND 3) THEN 500
    WHEN a.id_banca IN (
        SELECT bc.id_banca
        FROM clienti_proiect bc
        JOIN clienti_proiect c ON bc.id_client = c.id_client
        JOIN tranzactii_proiect t ON c.id_client = t.id_client
        JOIN banca_proiect b ON bc.id_banca = b.id_banca
        WHERE b.denumire IN ('Raiffeisen', 'Revolut')
        GROUP BY bc.id_banca, c.id_client
        HAVING COUNT(t.id_tranzactie) > 3) THEN 1000
    ELSE 0
END
WHERE a.id_banca IN (
    SELECT id_banca
    FROM banca_proiect
    WHERE denumire IN ('Raiffeisen', 'Revolut'));
SELECT USER FROM DUAL;


```

11 rows updated.

Script Output | Query Result | Task completed in 0.061 seconds

Script Output | Query Result | All Rows Fetched: 1 in 0.013 seconds

USER
SIMONG_65

30)Sa se afiseze pentru fiecare banca denumirea bancii, salariul mediu al angajatilor acesteia (rotunjit la doua zecimale) si numarul total de tranzactii efectuate de clientii bancii, dar doar pentru bancile care au mai mult de 3 tranzactii in total. Rezultatele sunt ordonate descrescator dupa salariul mediu al angajatilor.
 SELECT b.denumire AS banca, ROUND(AVG(a.salariu), 2) AS salariu_mediul, COUNT(t.id_tranzactie) AS numar_tranzactii FROM angajati_proiect a

```

JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
GROUP BY b.denumire
HAVING COUNT(t.id_tranzactie) > 3
ORDER BY salariu_mediul DESC;
SELECT USER FROM DUAL;

```

```


SELECT b.denumire AS banca,
ROUND(AVG(a.salariu), 2) AS salariu_mediul,
COUNT(t.id_tranzactie) AS numar_tranzactii
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
GROUP BY b.denumire
HAVING COUNT(t.id_tranzactie) > 3
ORDER BY salariu_mediul DESC;
SELECT USER FROM DUAL;


```

All Rows Fetched: 5 in 0.04 seconds

BANCA	SALARU_MEDIU	NUMAR_TRANZACTII
1 Revolut	6888.64	66
2 Banca Transilvania	6692.31	13
3 BRD	6612.22	18
4 BCR	6380	50
5 Raiffeisen Bank	4817.96	84

Script Output | Query Result | All Rows Fetched: 1 in 0.041 seconds

USER
SIMONG_65

31)Sa se afiseze detaliile angajatilor care au gestionat cel mai mare numar de clienti cu tranzactii, incluzand salariul acestora si banca la care sunt asociati, pentru bancile Raiffeisen, Revolut, BCR si BRD. Sa se afiseze rezultatele pentru angajatii care au gestionat mai mult de 1 clienti distinct si sa se ordoneze rezultatele descrescator in functie de numarul de clienti, urmat de salariu.

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariu, b.denumire AS banca,
COUNT(DISTINCT c.id_client) AS numar_clienti
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca

```

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```
WHERE b.denumire IN ('Raiffeisen', 'Revolut', 'BCR', 'BRD')
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire
HAVING COUNT(DISTINCT c.id_client) >=1
ORDER BY numar_clienti DESC, a.salariul DESC;
SELECT USER FROM DUAL;
```

```
SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire AS banca,
COUNT(DISTINCT c.id_client) AS numar_clienti
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
WHERE b.denumire IN ('Raiffeisen', 'Revolut', 'BCR', 'BRD')
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire
HAVING COUNT(DISTINCT c.id_client) >=1
ORDER BY numar_clienti DESC, a.salariul DESC;
```

```
SELECT USER FROM DUAL;
```

Script Output | Query Result | All Rows Fetched: 30 in 0.043 seconds

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARUL	BANCA	NUMAR_CLIENTI
13	2023 Morar	Iulia	6200	Revolut	2
14	2031 Stoica	Elena	6200	BCR	2
15	2016 Cristescu	Ioana	6050	Revolut	2
16	2024 Rusu	George	6000	Revolut	2
17	2027 Voinean	Cristian	5800	BCR	2
18	2029 Dragomir	Gabriel	5600	BCR	2
19	2032 Tudor	Andrei	5500	BCR	2
20	2033 Gheorghe	Maria	5300	BCR	2
21	2034 Pavel	Daniel	5200	BCR	2
22	2035 Ilie	Alexandra	5700	BRD	1
23	2036 Moldovan	Andrei	7400	BRD	1
24	2037 Ciucă	Ioana	6910	BRD	1
25	2038 Anghel	Mihai	6900	BRD	1
26	2040 Petrescu	Radu	6600	BRD	1
27	2041 Iacob	Simona	6300	BRD	1
28	2039 Dumitru	Gabriela	5800	BRD	1
29	2042 Rusu	Florin	5500	BRD	1
30	2043 Neagu	Elena	5400	BRD	1

```
SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire AS banca,
SUM(t.suma_tranzactiei) AS valoare_totala_tranzactii
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
WHERE t.suma_tranzactiei > 1000
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire
HAVING COUNT(DISTINCT c.id_client) >=1
ORDER BY valoare_totala_tranzactii DESC;
SELECT USER FROM DUAL;
```

Script Output | Query Result | All Rows Fetched: 1 in 0.035 seconds

USER
1 SIMONG_65

32)Sa se calculeze valoarea totala a tranzactiilor, dar numai pentru angajatii care au gestionat cel putin 3 clienti cu tranzactii ce au o valoare mai mare de 1000.

```
SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire AS banca,
SUM(t.suma_tranzactiei) AS valoare_totala_tranzactii
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
WHERE t.suma_tranzactiei > 1000
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire
HAVING COUNT(DISTINCT c.id_client) >=3
ORDER BY valoare_totala_tranzactii DESC;
SELECT USER FROM DUAL;
```

```
SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire AS banca,
SUM(t.suma_tranzactiei) AS valoare_totala_tranzactii
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
WHERE t.suma_tranzactiei > 1000
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, b.denumire
HAVING COUNT(DISTINCT c.id_client) >= 3
ORDER BY valoare_totala_tranzactii DESC;
SELECT USER FROM DUAL;
```

```
SELECT USER FROM DUAL;
```

Script Output | Query Result | All Rows Fetched: 14 in 0.055 seconds

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	BANCA	VALOARE_TOTALA_TRANZACTII
1	2000 Iobanici	Marian	Raiffeisen Bank	38700
2	2001 Enă	Mihaila	Raiffeisen Bank	38700
3	2002 Grivitea	Madalina	Raiffeisen Bank	38700
4	2003 Lungu	Mihai	Raiffeisen Bank	38700
5	2004 Oprea	Marius	Raiffeisen Bank	38700
6	2005 Popa	Adriana	Raiffeisen Bank	38700
7	2013 Petrescu	Catalin	Raiffeisen Bank	38700
8	2007 Marinescu	Ioana	Raiffeisen Bank	38700
9	2008 Grigorescu	Dan	Raiffeisen Bank	38700
10	2009 Ionescu	Radu	Raiffeisen Bank	38700
11	2010 Georgescu	Ana	Raiffeisen Bank	38700
12	2011 Stan	Mirela	Raiffeisen Bank	38700
13	2012 Dumitrescu	Laura	Raiffeisen Bank	38700
14	2006 Vasilescu	George	Raiffeisen Bank	38700

```
SELECT USER FROM DUAL;
```

Script Output | Query Result | All Rows Fetched: 1 in 0.032 seconds

USER
1 SIMONG_65

33)Sa se identifice angajatii care au gestionat cei mai multi clienti, dar doar pentru clientii din BRD, BCR sau Revolut, iar pentru acestia, se cere si calcularea unui bonus pe baza tranzactiilor.

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire AS banca,
COUNT(DISTINCT c.id_client) AS numar_clienti,
CASE
WHEN COUNT(t.id_tranzactie) BETWEEN 1 AND 3 THEN 500
WHEN COUNT(t.id_tranzactie) > 3 THEN 1000
ELSE 0 END AS bonus
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
WHERE b.denumire IN ('BRD', 'Revolut', 'BCR')
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire
HAVING COUNT(DISTINCT c.id_client) >=2
ORDER BY numar_clienti DESC; SELECT USER FROM DUAL;

```

The screenshot shows the SQL developer interface with two tabs: Script Output and Query Result.

Script Output:

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire AS banca,
COUNT(DISTINCT c.id_client) AS numar_clienti,
CASE
WHEN COUNT(t.id_tranzactie) BETWEEN 1 AND 3 THEN 500
WHEN COUNT(t.id_tranzactie) > 3 THEN 1000
ELSE 0
END AS bonus
FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
WHERE b.denumire IN ('BRD', 'Revolut', 'BCR')
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire
HAVING COUNT(DISTINCT c.id_client) >=2
ORDER BY numar_clienti DESC;
SELECT USER FROM DUAL;

```

Query Result:

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU	BANCA	NUMAR_CLIENTI	BONUS
8	2021Ion	Andrei	6425	Revolut	2	1000
9	2022Radu	Vlad	6600	Revolut	2	1000
10	2023Mihai	Iulia	6200	Revolut	2	1000
11	2024Nico	Georgiana	4000	Revolut	2	1000
12	2025Sandu	Alexandru	5500	BCR	2	1000
13	2026Matei	Iulia	7100	BCR	3	1000
14	2027Voines	Cristian	5800	BCR	2	1000
15	2028Marin	Roxana	4900	BCR	2	1000
16	2029Dragomir	Gabriel	5600	BCR	2	1000
17	2030Barbu	Vasile	7700	BCR	2	1000
18	2031Stoica	Elena	6200	BCR	2	1000
19	2032Tudor	Andrei	5500	BCR	2	1000
20	2033Gheorghe	Maria	5300	BCR	2	1000
21	2015Balan	Roberta	7700	Revolut	2	1000

Script Output X | Query Result X | All Rows Fetched: 21 in 0.093 seconds

User: SIMONG_65

34)Sa se afiseze cei mai activi angajati (pe baza numarului de tranzactii efectuate) pentru fiecare banca.

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire AS banca,
COUNT(t.id_tranzactie) AS numar_tranzactii FROM angajati_proiect a
JOIN banca_clienti_proiect bc ON a.id_banca = bc.id_banca
JOIN clienti_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire
HAVING COUNT(t.id_tranzactie) >5
ORDER BY numar_tranzactii DESC;
SELECT USER FROM DUAL;

```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire AS banca,
COUNT(t.id_tranzactie) AS numar_tranzactii
FROM angajati_proiect a
JOIN banca_client_proiect bc ON a.id_banca = bc.id_banca
JOIN client_proiect c ON bc.id_client = c.id_client
JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul, b.denumire
HAVING COUNT(t.id_tranzactie) >5
ORDER BY numar_tranzactii DESC;
SELECT USER FROM DUAL;

```

Script Output | Query Result | SQL | All Rows Fetched: 25 in 0.036 seconds

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARUL	BANCA	NUMAR_TRANZACTII
7	2006 Vasilescu	George	5000 Raiffeisen Bank	€	
8	2007 Marinescu	Ioana	4900 Raiffeisen Bank	€	
9	2008 Grigorescu	Dan	5400 Raiffeisen Bank	€	
10	2009 Ionescu	Radu	3400 Raiffeisen Bank	€	
11	2010 Georgescu	Ana	3200 Raiffeisen Bank	€	
12	2011 Stan	Mircea	5200 Raiffeisen Bank	€	
13	2012 Dumitrescu	Laura	4800 Raiffeisen Bank	€	
14	2013 Petrescu	Cătălin	4400 Raiffeisen Bank	€	
15	2014 Toma	Gabriela	8800 Revolut	€	
16	2015 Balan	Roberta	7700 Revolut	€	
17	2016 Cristescu	Ioana	6050 Revolut	€	
18	2017 Stanciu	Dumitru	7000 Revolut	€	
19	2018 Ropescu	Mihai	7400 Revolut	€	
20	2019 Ilnu	Alina	6400 Revolut	€	
21	2020 Vlad	Radu	7200 Revolut	€	
22	2021 Ion	Andreea	6425 Revolut	€	
23	2022 Enache	Vlad	6600 Revolut	€	
24	2023 Morar	Iulia	6200 Revolut	€	
25	2001 Ene	Mihaela	5200 Raiffeisen Bank	€	

```

JOIN tranzactii_proiect t ON c.id_client = t.id_client
JOIN banca_proiect b ON bc.id_banca = b.id_banca
GROUP BY a.id_angajat, a.nume_angajat, a.prenume_angajat, a.salariul
HAVING COUNT(t.id_tranzactie) >5
ORDER BY numar_tranzactii DESC;
SELECT USER FROM DUAL;

```

Script Output | Query Result | SQL | All Rows Fetched: 1 in 0.047 seconds

USER
1 SIMONG_65

35) Sa se calculeze tranzactiile efectuate intr-o perioada de timp, afisand suma tranzactiei si clasificarea acestora in functie de prima cifra a sumei (mic sau mare).

```

SELECT t.id_tranzactie, TO_CHAR(t.data_tranzactie, 'DD-Mon-YYYY') AS data_tranzactie_formatata,
t.suma_tranzactie,
DECODE(SUBSTR(TO_CHAR(t.suma_tranzactie), 1, 1),
'1',
'Mică', '2',
'Mare', '3',
'Alta') AS tip_tranzactie
FROM tranzactii_proiect t
WHERE t.data_tranzactieI BETWEEN TO_DATE('01-JAN-2025', 'DD-Mon-YYYY') AND SYSDATE
ORDER BY t.data_tranzactieI DESC;
SELECT USER FROM DUAL;

```

```

----  

SELECT t.id_tranzactie,  

TO_CHAR(t.data_tranzactie, 'DD-Mon-YYYY') AS data_tranzactie_formatata,  

t.suma_tranzactie,  

DECODE(SUBSTR(TO_CHAR(t.suma_tranzactie), 1, 1),  

'1', 'Mică',  

'2', 'Mare',  

'3', 'Alta') AS tip_tranzactie  

FROM tranzactii_proiect t  

WHERE t.data_tranzactieI BETWEEN TO_DATE('01-JAN-2025', 'DD-Mon-YYYY') AND SYSDATE  

ORDER BY t.data_tranzactieI DESC;  

SELECT USER FROM DUAL;

```

Script Output | Query Result | SQL | All Rows Fetched: 5 in 0.039 seconds

ID_TRANZACTIE	DATA_TRANZACTIE_FORMATATA	SUMA_TRANZACTIEI	TIPO_TRANZACTIE
1	600019 09-Jan-2025	1050	Mică
2	600018 07-Jan-2025	1000	Mică
3	600017 05-Jan-2025	9500	Alta
4	600016 03-Jan-2025	9000	Alta
5	600015 01-Jan-2025	8500	Alta

Script Output | Query Result | SQL | All Rows Fetched: 1 in 0.016 seconds

USER
1 SIMONG_65

36) Sa se creeze o tabela virtuala pentru angajati cu varsta mai mare de 30 de ani si care lucreaza in departamentele ce contin in denumirea lor "relatii". Sa se actualizeze salariul astfel incat, daca salariul este mai mic de 4000 si angajatul lucreaza in departamentul 10, va primi o majorare de 5000, daca salariul este mai mare de 4500 si angajatul lucreaza in departamentul 16, va primi o majorare de 2500.

CREATE OR REPLACE VIEW v_angajati_proiect3

AS

```

SELECT a.id_angajat, a.nume_angajat, a.prenume_angajat, a.varsta, a.id_departamente,
d.denumire_departament, a.salariul
FROM angajati_proiect a
JOIN departamente_proiect d ON a.id_departamente = d.id_departamente
WHERE a.varsta > 30

```

PROIECT – BAZE DE DATE – ȘIMON GEORGIANA

```

AND a.id_departamente IN
(SELECT id_departamente FROM departamente_proiect
WHERE UPPER(denumire_departament) LIKE '%RELATII%');
SELECT *FROM v_angajati_proiect3;
UPDATE v_angajati_proiect3
SET salariul = salariul+
CASE
WHEN salariul<4000 AND id_departamente=10 THEN 5000
WHEN salariul>4500 AND id_departamente=16 THEN 2500
ELSE 0
END;
SELECT USER FROM DUAL;

```

The screenshot shows the Oracle SQL Developer interface with several windows open:

- Script Output:** Shows the creation of a view named `V_ANGAJATI_PROJECT3` and a PL/SQL block that updates the `v_angajati_proiect3` view based on department ID and salary conditions.
- Query Result:** Shows the creation of the view `V_ANGAJATI_PROJECT3`.
- Script Output:** Shows the creation of a synonym `V_ANGAJATI_PROJECT3` for the view `V_ANGAJATI_PROJECT3`.
- Query Result:** Shows the creation of the synonym `V_ANGAJATI_PROJECT3`.
- Script Output:** Shows the creation of a user `SIMONG_65`.
- Query Result:** Shows the creation of the user `SIMONG_65`.

37) Sa se creeze un sinonim pentru tabela `angajati_PROIECT`, sa se utilizeze in 2 interogari si sa se stearga.

```

CREATE SYNONYM angajati_proiect_syn FOR angajati_proiect;
SELECT * FROM angajati_proiect_syn WHERE varsta=30;
SELECT *FROM angajati_proiect_syn WHERE id_departamente=10;
SELECT * FROM USER_SYNONYMS;
DROP SYNONYM angajati_proiect_syn;
SELECT USER FROM DUAL;

```

The screenshot shows the Oracle SQL Developer interface with several windows open:

- Script Output:** Shows the creation of a synonym `ANGAJATI_PROJECT_SYN` for the table `angajati_proiect`.
- Query Result:** Shows the creation of the synonym `ANGAJATI_PROJECT_SYN`.
- Script Output:** Shows the dropping of the synonym `ANGAJATI_PROJECT_SYN`.
- Query Result:** Shows the dropping of the synonym `ANGAJATI_PROJECT_SYN`.
- Script Output:** Shows the creation of a user `SIMONG_65`.
- Query Result:** Shows the creation of the user `SIMONG_65`.

PROIECT – BAZE DE DATE – SIMON GEORGIANA

```

CREATE SYNONYM angajati_proiect_syn FOR angajati_proiect;
SELECT * FROM angajati_proiect_syn WHERE varsta=30;
SELECT *FROM angajati_proiect_syn WHERE id_departamente=10;
SELECT * FROM USER_SYNONYMS;
DROP SYNONYM angajati_proiect_syn;

Script Output X | Query Result X
SQL | Task completed in 0.181 seconds

Synonym ANGAJATI_PROIECT_SYN created.

SELECT * FROM angajati_proiect_syn WHERE varsta=30;
SELECT *FROM angajati_proiect_syn WHERE id_departamente=10;
SELECT * FROM USER_SYNONYMS;
DROP SYNONYM angajati_proiect_syn;

Script Output X | Query Result X
SQL | All Rows Fetched: 5 in 0.016 seconds
ID_ANGAJAT | NUME_ANGAJAT | PRENUME_ANGAJAT | VARSTA | SALARIU | ID_MANAGER | ID_DEPARTAMENTE | ID_BANCA
1 | 2004 Oprea | Marius | 30 | 3700 | 2000 | 23 | 3000
2 | 2017 Stanciu | Dumitru | 30 | 7000 | 2014 | 18 | 3001
3 | 2028 Dragomir | Gabriel | 30 | 5600 | 2028 | 11 | 3002
4 | 2042 Rusu | Florin | 30 | 5500 | 2041 | 18 | 3003
5 | 2062 Rusu | Mirela | 30 | 6100 | 2061 | 11 | 3005

SELECT *FROM angajati_proiect_syn WHERE id_departamente=10;
SELECT * FROM USER_SYNONYMS;
DROP SYNONYM angajati_proiect_syn;

Script Output X | Query Result X
SQL | All Rows Fetched: 5 in 0.042 seconds
ID_ANGAJAT | NUME_ANGAJAT | PRENUME_ANGAJAT | VARSTA | SALARIU | ID_MANAGER | ID_DEPARTAMENTE | ID_BANCA
1 | 2026 Matei | Iulia | 35 | 7100 | 2025 | 10 | 3002
2 | 2027 Voinean | Cristian | 29 | 5800 | 2026 | 10 | 3002
3 | 2096 Dobre | Alina | 44 | 8500 | 2095 | 10 | 3008
4 | 2057 Vasile | Dan | 38 | 7400 | 2096 | 10 | 3008
5 | 2098 Baciu | Irina | 34 | 6400 | 2097 | 10 | 3008

SELECT * FROM USER_SYNONYMS;
DROP SYNONYM angajati_proiect_syn;

Script Output X | Query Result X
SQL | All Rows Fetched: 1 in 0.062 seconds
SYNONYM_NAME | TABLE_OWNER | TABLE_NAME | DB_LINK | ORIGIN_CON_ID
ANGAJATI_PROIECT_SYN SIMONG_65 | ANGAJATI_PROIECT (null) | 3

```