MONITORIZAREA ACTIVITĂȚII UNEI AGENȚII IMOBILIARE

Proiect Baze de Date

**Ciurescu Irina Alexandra**

Grupa 142

## Descrierea modelului real, a utilității acestuia și a regulilor de funcționare

Sistemul informatic al agenției imobiliare este conceput și pus la dispoziția utilizatorilor (personalul agenției). Aplicația permite utilizarea unei baze de date pentru gestiunea ofertelor și cererilor prezentate, a contractelor și tranzacțiilor încheiate.

Utilitatea acestei baze de date constă în furnizarea unui mijloc eficient de gestionare și accesare a informațiilor despre proprietățile imobiliare și clienții agenției. Aceasta poate fi folosită de către agenți pentru a găsi rapid proprietățile potrivite pentru clienții lor, precum și pentru a stoca informațiile despre tranzacțiile finalizate și stadiul acestora.

**Principalele servicii oferite de agenție sunt:**

* intermedieri în vederea vânzării, cumpărării sau închirierii de proprietăți imobiliare;
* consultanță în domeniul pieței imobiliare;
* evaluări imobiliare.

**Aplicația permite monitorizarea următoarelor activități:**

* activitatea de gestiune a utilizatorilor (personalul agenției)
* activitatea de gestiune a clienților
* activitatea de gestionare a cererilor și ofertelor
* activitatea de gestionare a contractelor și tranzacțiilor încheiate.
* **Activitatea de gestiune a utilizatorilor**

Grupul de utilizatori este constituit de personalul agenției (director, secretar, contabil, agenți imobiliari etc). Se vor memora datele personale ale angajaților (cod, nume, prenume, cnp, studii, funcție, adresă, telefon, email) dar și datele de logare în aplicație (cont și parolă).

* **Activitatea de gestiune a clienților**

Pentru fiecare client se vor memora datele personale (cod, nume, prenume, cnp, adresa, telefon, email) iar operațiile permise vor fi de adăugare, modificare, ștergere, căutare client.

* **Activitatea de gestionare a cererilor și ofertelor**

Pentru fiecare **ofertă** se vor memora datele proprietarului (cod client) și principalele caracteristici: cod proprietate, tipul operației (vânzare/închiriere), tipul imobilului (casa, apartament, teren), adresa, suprafață, an construcție, preț, descriere proprietate etc.

Pentru fiecare **cerere** se vor memora datele clientului (cod client) și informații despre proprietatea –imobiliară solicitată: cod cerere, tipul operației (cumpărare/închiriere), tipul imobilului (casa, apartament, teren), buget disponibil, preferințe de căutare (zona, suprafață minimă, etaj maxim, an construcție etc.).

Aplicația va permite agentului imobiliar să adauge, să modifice, să caute sau să șteargă o ofertă sau o cerere primită din partea unui client. Tot prin intermediul aplicației agentul poate consulta oferta de imobile (apartamente, case, terenuri) pe baza opțiunilor clientului și va oferi un răspuns prompt acestuia.

* **Activitatea de gestionare a contractelor și tranzacțiilor încheiate**

Pentru fiecare client se va încheia un **contract** între client și agenție care va conține următoarele date: număr contract, cod agent, cod client, data contract, comision.

Pentru fiecare **tranzacție** încheiată se vor memora: cod tranzacție, data, cod agent, cod client pentru vânzător, cod client pentru cumpărător, cod ofertă, cod cerere, suma plătită, comisioanele percepute pentru vânzător și cumpărător.

## Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului

În cazul agențiilor imobiliare se pot aplica următoarele constrângeri:

* constrângeri de tipul cheii primare pot fi aplicate pe cod agent, cod client, cod proprietate, cod cerere, cod tranzacție, cod contract, cont utilizator
* constrângeri de tipul cheii externe pot fi aplicate pe cod agent, cod client, cod proprietate, cod cerere
* constrângerile de validitate pot fi aplicate pe suprafața totală a proprietății, numărul de camere sau pe orice altă informație care trebuie să respecte anumite standarde
* constrângerile de tipul valorii implicite se pot aplica pentru datele calendaristice sau la prețul minim de vânzare acceptabil

## Descrierea entităților, incluzând precizarea cheii primare

* ANGAJAT
* Primary Key: ID\_ANAGAJAT
* Păstrează datele personale ale angajaților și datele de logare în aplicație
* CLIENT
* Primary Key: ID\_CLIENT
* Memorează datele personale ale clienților
* OFERTĂ
* Primary Key: ID\_OFERTA
* Memorează codul clientului și principalele caracteristici ale proprietății oferite spre vânzare/închiriere
* CERERE
* Primary Key: ID\_CERERE
* Memorează codul clientului și informații despre proprietatea imobiliară solicitată
* CONTRACT
* Primary Key: ID\_CONTRACT
* Memorează date despre contractele încheiate de agenție
* TRANZACȚIE
* Primary Key: ID\_TRANZACTIE
* Memorează date despre tranzacțiile încheiate de agenție
* PROPRIETATE
* Primary Key: ID\_PROPRIETATE
* Nomenclator pentru tipurile de proprietăți (casa, apartament, teren)
* OPERAȚIE
* Primary Key: ID\_OPERAȚIE
* Nomenclator pentru tipurile de operații (vânzare, cumpărare, închiriere)
* FUNCȚIE
* Primary Key: ID\_FUNCȚIE
* Nomenclator pentru funcțiile angajaților (director, contabil, agent etc.)
* STUDII
* Primary Key: ID\_STUDII
* Nomenclator pentru studiile angajaților (superioare, liceale, gimnaziale)
* JUDEȚ
* Primary Key: ID\_JUDET
* Nomenclator pentru județe
* LOCALITATE
* Primary Key: ID\_LOCALITATE
* Nomenclator pentru localități
* ANG\_STUDII
* Primary Key: ID\_ANGAJAT, ID\_STUDII
* Memorează pentru fiecare angajat ce formă/forme de studii a absolvit

## Descrierea relațiilor, incluzând precizarea cardinalității acestora

***ANGAJAT – are – FUNCTIE*** : Un angajat are exact o funcție în agenție. Relația este 1 to M(0).

***ANGAJAT – încheie – CONTRACT*** : Un angajat poate încheia mai multe contracte, eventual niciunul, iar un contract este încheiat de exclusiv un angajat. Relația este 1 to M(0).

***ANGAJAT – încheie – TRANZACȚIE*** : Un angajat poate încheia mai multe tranzacții, eventual niciuna, iar o tranzacție este încheiată de exclusiv un angajat. Relația este 1 to M(0).

***ANGAJAT – are studii – ANG\_STUDII***: Un angajat a absolvit cel puțin o formă de învățământ. Relația este 1 to M(1).

***CLIENT – prezintă – OFERTĂ*** : Un client poate prezenta mai multe oferte agenției, eventual niciuna, iar o ofertă e prezentată exclusiv de un client. Relația este 1 to M(0).

***CLIENT – are – CERERE***: Un client poate prezenta mai multe cereri agenției, eventual niciuna, iar o cerere e prezentată exclusiv de un client. Relația este 1 to M(0).

***CLIENT – încheie – CONTRACT*** : Un client poate încheia mai multe contracte, eventual niciunul, iar un contract e încheiat de un client. Relația este 1 to M(0).

***OFERTĂ – implică – OPERAȚIE*** : O ofertă implică un tip de operație, iar un tip de operație poate fi regăsit în mai multe oferte, eventual niciuna. Relația este M(0) to 1.

***OFERTĂ – implică – PROPRIETATE*** : O ofertă implică un tip de proprietate, iar un tip de proprietate poate fi regăsit în mai multe oferte, eventual în niciuna. Relația este M(0) to 1.

***OFERTĂ – se află în – LOCALITATE*** : O ofertă implică vânzarea unei proprietăți dintr-o localitate și pot exista mai multe oferte pentru o localitate. Nu este obligatoriu ca pentru toate localitățile să avem oferte. Relația este M(0) to 1.

***OFERTĂ – se încheie – TRANZACȚIE*** : Pentru o ofertă se poate încheia maxim o tranzacție, eventual niciuna, iar în cadrul unei tranzacții apare o singură ofertă. Relația este 1 to 1(0).

***CERERE – implică – OPERAȚIE*** : O cerere implică un tip de operație, iar un tip de operație poate fi regăsit în mai multe cereri. Unele tipuri de operații pot să nu apară printre cereri. Relația este M(0) to 1.

***CERERE – implică – PROPRIETATE*** : O cerere implică un tip de proprietate, iar un tip de proprietate poate fi regăsit în mai multe cereri. Unele tipuri de proprietate pot să nu apară. Relația este M(0) to 1.

***CERERE – se află în – LOCALITATE*** : O cerere implică cumpărarea/închirierea unei proprietăți dintr-o localitate și pot exista mai multe cereri pentru o localitate. Nu este obligatoriu ca pentru toate localitățile să avem cereri. Relația este M(0) to 1.

***CERERE – se încheie – TRANZACȚIE*** : Pentru o cerere se poate încheia maxim o tranzacție, eventual niciuna, iar în cadrul unei cereri apare o singură ofertă. Relația este 1 to 1(0).

***JUDEȚ – are – LOCALITATE*** : Într-un județ sunt mai multe localități, iar o localitate face parte dintr-un singur județ. Relația este 1 to M(1).

## Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor

**ANGAJAT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_angajat (PK) | Number(3) | Not null |  | 1,2,3... |
| Id\_functie (FK) | Number(2) | Not null |  | 1,2,3... |
| Nume | Varchar2(20) | Not null |  |  |
| Prenume | Varchar2(20) | Not null |  |  |
| Cnp | Varchar2(13) | Not null |  |  |
| Adresa | Varchar2(30) | Not null |  |  |
| Telefon | Varchar2(10) | Not null |  |  |
| email | Varchar2(20) | Not null |  |  |
| Cont | Varchar2(20) | Not null |  |  |
| parola | Varchar2(20) | Not null |  |  |

**CLIENT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_client (PK) | Number(5) | Not null |  | 1,2,3... |
| Nume | Varchar2(20) | Not null |  |  |
| Prenume | Varchar2(20) | Not null |  |  |
| Cnp | Varchar2(13) | Not null |  |  |
| Adresa | Varchar2(30) | Not null |  |  |
| Telefon | Varchar2(10) | Not null |  |  |
| email | Varchar2(20) |  |  |  |

**OFERTĂ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_oferta (PK) | Number(5) | Not null |  | 1,2,3... |
| Id\_client (FK) | Number(5) | Not null |  | 1,2,3... |
| Id\_operatie (FK) | Varchar2(1) | Not null |  | V, I |
| Id\_proprietate(FK) | Varchar2(1) | Not null |  | C, A, T |
| Id\_localitate (FK) | Number(10) | Not null |  | 1, 2, 3... |
| Strada | Varchar2(20) | Not null |  |  |
| Numar | Varchar2(5) |  |  |  |
| Suprafața | Number(10,2) | >0 |  |  |
| An\_construcție | Number(4) | >0 |  |  |
| Preț | Number(10) | >0 |  |  |
| Descriere | Varchar2(100) |  |  |  |

**CERERE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_cerere (PK) | Number(5) | Not null |  | 1,2,3... |
| Id\_client (FK) | Number(5) | Not null |  | 1,2,3... |
| Id\_operatie (FK) | Varchar2(1) | Not null |  | C, I |
| Id\_proprietate(FK) | Varchar2(1) | Not null |  | C, A, T |
| Id\_localitate (FK) | Number(10) | Not null |  |  |
| Buget | Number(10) | >0 |  |  |
| Suprafața\_minima | Number(10,2) | >0 |  |  |
| An\_construcție | Number(4) | >0 |  |  |

**CONTRACT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_contract (PK) | Number(5) | Not null |  | 1,2,3... |
| Id\_angajat(FK) | Number(3) | Not null |  | 1,2,3... |
| Id\_client (FK) | Number(5) | Not null |  | 1,2,3... |
| Data | Date | Not null | SYSDATE |  |
| Comision | Number(2) | >0 |  | 10, 20, 30... % |

**TRANZACȚIE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_tranzacție (PK) | Number(5) | Not null |  | 1,2,3... |
| Id\_angajat(FK) | Number(3) | Not null |  | 1,2,3... |
| Id\_cerere(FK) | Number(5) | Not null |  | 1,2,3... |
| Id\_oferta (FK) | Number(5) | Not null |  | 1,2,3... |
| Data | Date | Not null | SYSDATE |  |
| Suma | Number(10) | >0 |  |  |

**PROPRIETATE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_proprietate(PK) | Varchar2(1) | Not null |  | C, A, T |
| Denumire | Varchar2(20) | Not null |  | Casa, Apartament, Teren |

**OPERAȚIE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_operație(PK) | Varchar2(1) | Not null |  | V, C, I |
| Denumire | Varchar2(20) | Not null |  | Vanzare, Cumparare, Inchiriere |

**FUNCȚIE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_funcție (PK) | Number(2) | Not null |  | 1,2,3... |
| Denumire | Varchar2(20) | Not null |  | Director, Contabil, Agent, Secretar, Sofer |

**STUDII**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_studii(PK) | Varchar2(1) | Not null |  | S, L, G |
| Denumire | Varchar2(20) | Not null |  | Superioare, Liceale, Gimnaziale |

**JUDEȚ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_județ (PK) | Varchar2(2) | Not null |  | B, IS, BV... |
| Denumire | Varchar2(20) | Not null |  | Iasi ... |

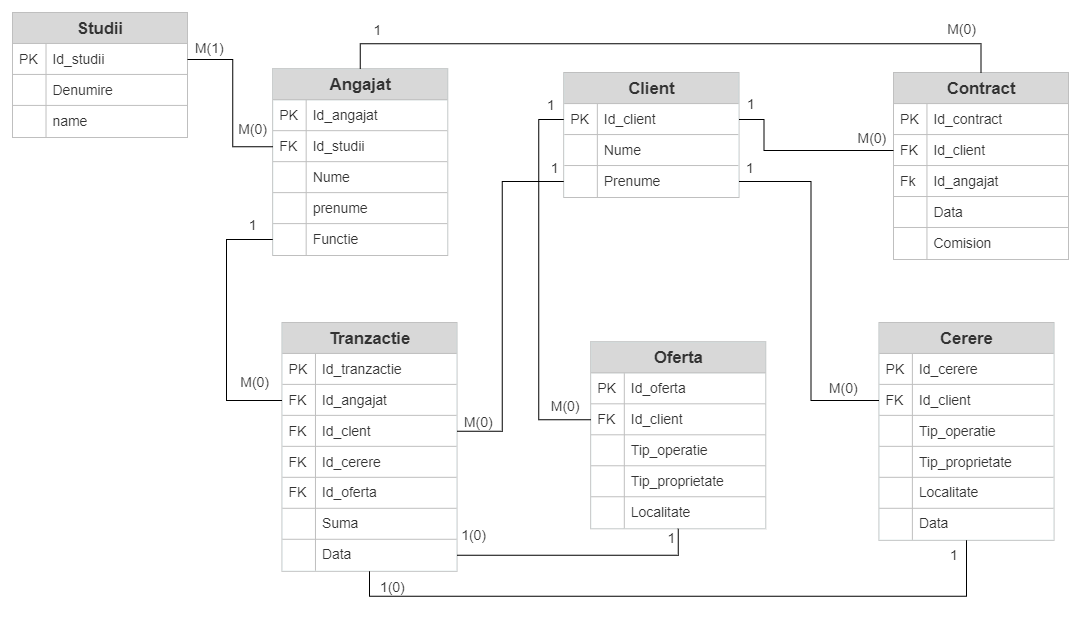
**LOCALITATE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_localitate(PK) | Number(10) | Not null |  | 1,2,3... |
| Id\_judet (FK) | Varchar2(2) | Not null |  | B, IS, BV... |
| Denumire | Varchar2(20) | Not null |  | Bucuresti... |

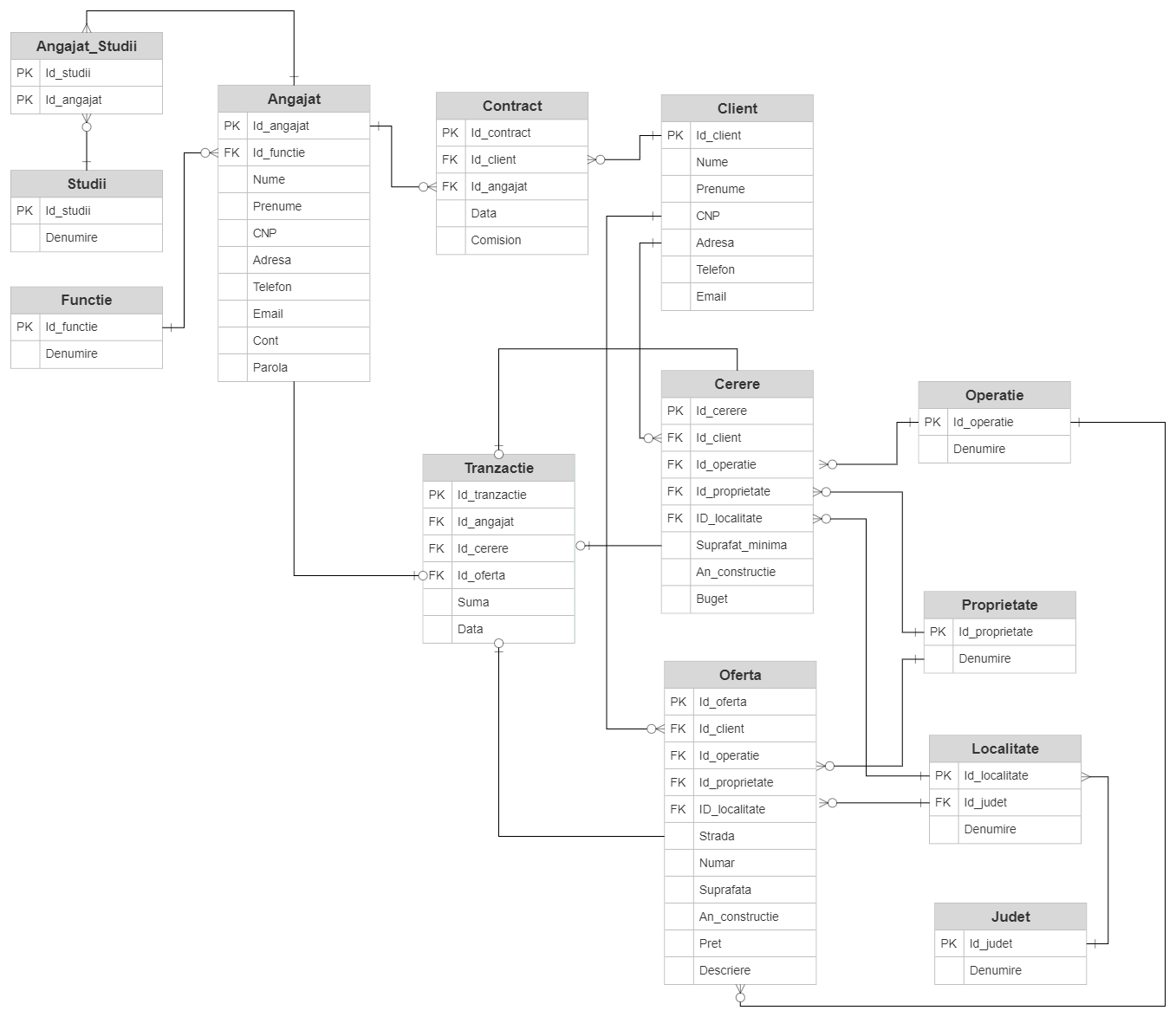
**ANG\_STUDII**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip date | Constrângeri | Valori implicite | Valori posibile |
| Id\_studii (PK) | Varchar2(1) | Not null |  | S, L, G |
| Id\_angajat (PK) | Number(3) | Not null |  | 1,2,3... |

## Realizarea diagramei entitate-relație



## Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6.



## Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

* **ANGAJAT** (Id\_angajat, Id\_functie, nume, prenume, cnp, telefon, adresa, email, cont, parola);
* **OFERTA** (Id\_oferta, Id\_client, Id\_operatie, Id\_proprietate, Id\_localitate, strada, numar, suprafata, an\_constructie, pret, descriere);
* **CERERE** (Id\_cerere, Id\_client, Id\_operatie, Id\_proprietate, Id\_localitate, buget, suprafata\_minima, an\_constructie);
* **CONTRACT** (Id\_contract, Id\_angajat, Id\_client, data, comision);
* **TRANZACTIE** (Id\_tranzactie, Id\_angajat, Id\_cerere, Id\_oferta, data, suma);
* **PROPRIETATE** (Id\_proprietate, denumire);
* **OPERATIE** (Id\_operatie, denumire);
* **FUNCTIE** (Id\_functie , denumire);
* **STUDII** (Id\_studii, denumire);
* **JUDETE** (Id\_judet, denumire);
* **LOCALITATE** (Id\_localitate, Id\_judet, denumire);
* **ANG\_STUDII** (Id\_angajat, Id\_studii);

## Realizarea normalizării până la forma normală 3

**FORMA NORMALA 1 (FN1)**

Valorile primite de atribute trebuie să fie atomice (nu pot fi liste, mulțimi sau alte structuri complexe) și atributele nu trebuie să fie repetitive.

**Exemplu forma non-FN1**

**ANGAJAT**

|  |  |
| --- | --- |
| **Angajat** | **Incadrare** |
| Popescu Ion | agent vanzari, studii liceale |
| Rosu Maria | agent vanzari, studii liceale |
| Rosu Maria | contabil, studii medii |

**Anomalii**

* mai multe valori semnificative în același câmp
* nu există o modalitate de a identifica în mod unic o înregistrare

**Transformare la forma FN1** (introducerea unor noi coloane, iar pentru a asigura unicitatea unei înregistrări se va utiliza cheia primară)

**ANGAJAT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id\_angajat** | **Nume** | **Prenume** | **Functie** | **Studii** |
| 1 | Popescu | Ion | agent vanzari | liceale |
| 2 | Rosu | Maria | agent vanzari | superioare |

**FORMA NORMALA 2 (FN2)**

Orice atribut este complet dependent de cheie. Presupune eliminarea redundanțelor și a anomaliilor care pot apare la actualizarea datelor.

**Exemplu forma non-FN2**

**CONTRACTE**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id\_Contract | NumeAngajat | PrenumeAngajat | CodAngajat | NumeClient | PrenumeClient | CodClient |
| 1 | Pop | Ioana | 1 | Avram | Ana | 1 |
| 2 | Ionescu | Mihai | 2 | Iancu | Rares | 2 |
| 3 | Pop | Ioana | 1 | Avram | Ana | 1 |
| 4 | Antoche | Ana | 3 | Iancu | Rares | 2 |

**Anomalii:**

* redundanța datelor (angajatul Pop si clienții Avram și Iancu au încheiat mai multe contracte și de fiecare dată memorăm datele acestora)
* limitate posibilități de inserare a datelor (de exemplu nu pot fi introduse date despre un agent de vânzări sau un client dacă nu a încheiat cel puțin un contract)
* pierderi de date la ștergere (dacă se șterge contractul cu id-ul 2 se pierd toate datele referitoare la agentul de vânzări cu codul 2)
* la modificarea datelor (dacă se modifică codul unui agent pot apare anomalii dacă nu se modifică codul acestuia în toate înregistrările)

**Transformare la forma FN2** (datele vor fi separate in tabele diferite)

**ANGAJAT**

|  |  |  |
| --- | --- | --- |
| Id\_angajat | Nume | Prenume |
| 1 | Pop | Ioana |
| 2 | Ionescu | Mihai |
| 3 | Antoche | Ana |

**CLIENŢI**

|  |  |  |
| --- | --- | --- |
| Id\_client | Nume | Prenume |
| 1 | Avram | Ana |
| 2 | Iancu | Rares |

**CONTRACT**

|  |  |  |
| --- | --- | --- |
| Id\_contract | Id\_angajat | Id\_client |
| 1 | 1 | 1 |
| 2 | 2 | 2 |
| 3 | 1 | 1 |
| 4 | 3 | 2 |

**FORMA NORMALA 3 (FN3)**

Atributele care nu sunt chei primare depind funcțional numai de cheile primare ale tabelului, nu există astfel nici un fel de dependențe tranzitive (adică, niciunul din atributele non-cheie nu este dependent de alt atribut, care la rândul său este dependent de cheia relației))

**Exemplu forma non-FN3**

**CERERI**

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_cerere | Id\_ client | Judet | Localitate |
| 1 | 2 | Iasi | Targu Frumos |
| 2 | 2 | Vrancea | Focsani |
| 3 | 4 | Iasi | TarguFrumos |
| 4 | 10 | Vrancea | Focsani |

**Anomalii:**

* atributul non-cheie Localitate este dependent de atributul non-cheie Judeţ

Transformare la forma **FN3** (reorganizăm datele adăugând două tabele noi)

**CERERI**

|  |  |  |
| --- | --- | --- |
| Id\_cerere | Id\_ client | Id\_Localitate |
| 1 | 2 | 1 |
| 2 | 2 | 2 |
| 3 | 4 | 1 |
| 4 | 10 | 2 |

**JUDET**

|  |  |
| --- | --- |
| Id\_judet | Denumire |
| IS | Iasi |
| VN | Vrancea |

**LOCALITATE**

|  |  |  |
| --- | --- | --- |
| Id\_Localitate | Id\_judet | Localitate |
| 1 | IS | Targu Frumos |
| 2 | VN | Focsani |

## Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele

CREATE SEQUENCE TRANZACTIE\_TRANZACTIE\_ID

INCREMENT BY 1

START WITH 1

MAXVALUE 9999

NOCACHE

NOCYCLE;

COMMIT;

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,1,1,1,SYSDATE,10000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,2,2,2,SYSDATE,20000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,3,3,5,SYSDATE,300000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,3,4,4,SYSDATE,40000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,3,5,3,SYSDATE,50000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,4,6,2,SYSDATE,90000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,7,1,6,SYSDATE,10000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,8,2,7,SYSDATE,20000);

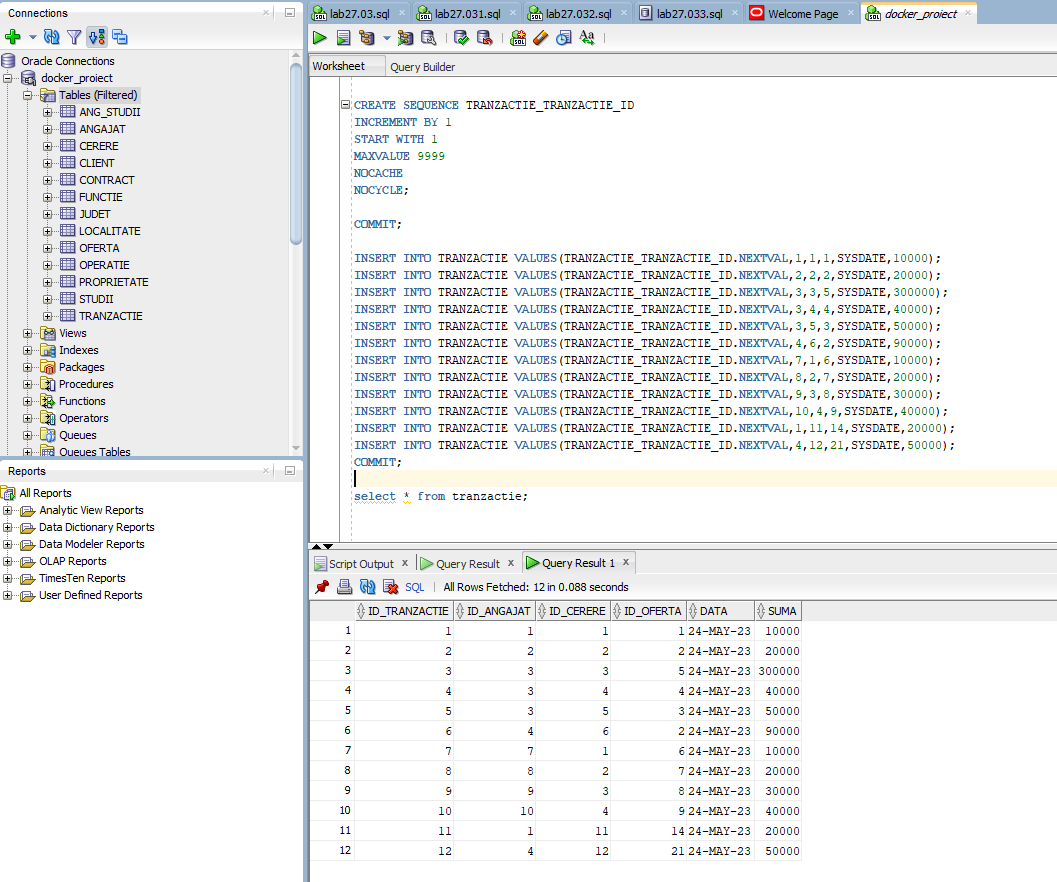
INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,9,3,8,SYSDATE,30000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,10,4,9,SYSDATE,40000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,1,11,14,SYSDATE,20000);

INSERT INTO TRANZACTIE VALUES(TRANZACTIE\_TRANZACTIE\_ID.NEXTVAL,4,12,21,SYSDATE,50000);

COMMIT;



## Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea

CREATE TABLE JUDET

(ID\_JUDET VARCHAR2(2) PRIMARY KEY,

DENUMIRE VARCHAR2(20) NOT NULL

);

COMMIT;

INSERT INTO JUDET VALUES ('AB', 'Alba');

INSERT INTO JUDET VALUES ('AR', 'Arad');

INSERT INTO JUDET VALUES ('AG', 'Arges');

INSERT INTO JUDET VALUES ('BC', 'Bacau');

INSERT INTO JUDET VALUES ('BH', 'Bihor');

INSERT INTO JUDET VALUES ('BN', 'Bistrita-Nasaudd');

INSERT INTO JUDET VALUES ('BT', 'Botosani');

INSERT INTO JUDET VALUES ('BR', 'Braila');

INSERT INTO JUDET VALUES ('BV', 'Brasov');

INSERT INTO JUDET VALUES ('BZ', 'Buzau');

INSERT INTO JUDET VALUES ('CL', 'Calarasi');

INSERT INTO JUDET VALUES ('CS', 'Caras-Severin');

INSERT INTO JUDET VALUES ('CJ', 'Cluj');

INSERT INTO JUDET VALUES ('CT', 'Constanta');

INSERT INTO JUDET VALUES ('CV', 'Covasna');

INSERT INTO JUDET VALUES ('DB', 'DamboviŁa');

INSERT INTO JUDET VALUES ('DJ', 'Dolj');

INSERT INTO JUDET VALUES ('GL', 'Galati');

INSERT INTO JUDET VALUES ('GR', 'Giurgiu');

INSERT INTO JUDET VALUES ('GJ', 'Gorj');

INSERT INTO JUDET VALUES ('HR', 'Harghita');

INSERT INTO JUDET VALUES ('HD', 'Hunedoara');

INSERT INTO JUDET VALUES ('IL', 'Ialomita');

INSERT INTO JUDET VALUES ('IS', 'Iasi');

INSERT INTO JUDET VALUES ('IF', 'Ilfov');

INSERT INTO JUDET VALUES ('MM', 'Maramures');

INSERT INTO JUDET VALUES ('MH', 'Mehedinti');

INSERT INTO JUDET VALUES ('MS', 'Mures');

INSERT INTO JUDET VALUES ('NT', 'Neamt');

INSERT INTO JUDET VALUES ('OT', 'Olt');

INSERT INTO JUDET VALUES ('PH', 'Prahova');

INSERT INTO JUDET VALUES ('SJ', 'Salaj');

INSERT INTO JUDET VALUES ('SM', 'Satu Mare');

INSERT INTO JUDET VALUES ('SB', 'Sibiu');

INSERT INTO JUDET VALUES ('SV', 'Suceava');

INSERT INTO JUDET VALUES ('TR', 'Teleorman');

INSERT INTO JUDET VALUES ('TM', 'Timis');

INSERT INTO JUDET VALUES ('TL', 'Tulcea');

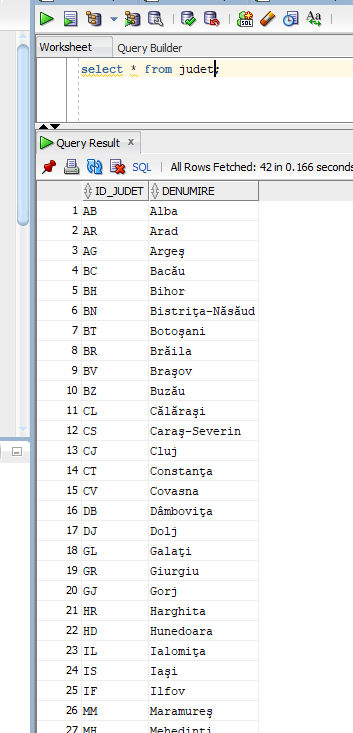
INSERT INTO JUDET VALUES ('VL', 'Valcea');

INSERT INTO JUDET VALUES ('VS', 'Vaslui');

INSERT INTO JUDET VALUES ('VN', 'Vrancea');

INSERT INTO JUDET VALUES ('B', 'Bucurestii');

COMMIT;



-- ##########################################

CREATE TABLE LOCALITATE

(ID\_LOCALITATE NUMBER(10) PRIMARY KEY,

ID\_JUDET VARCHAR2(2),

DENUMIRE VARCHAR2(20) NOT NULL

);

COMMIT;

ALTER TABLE LOCALITATE

ADD CONSTRAINT FKJUDET FOREIGN KEY

(

ID\_JUDET

)

REFERENCES JUDET

(

ID\_JUDET

);

INSERT INTO LOCALITATE VALUES (1, 'IS', 'Iasi');

INSERT INTO LOCALITATE VALUES (2, 'B', 'Bucuresti');

INSERT INTO LOCALITATE VALUES (3, 'CT', 'Constanta');

INSERT INTO LOCALITATE VALUES (4, 'CT', 'Mangalia');

INSERT INTO LOCALITATE VALUES (5, 'GL', 'Galati');

INSERT INTO LOCALITATE VALUES (6, 'GL', 'Tecuci');

INSERT INTO LOCALITATE VALUES (7, 'VS', 'Vaslui');

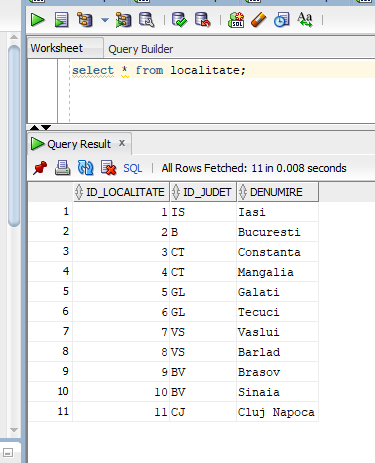
INSERT INTO LOCALITATE VALUES (8, 'VS', 'Barlad');

INSERT INTO LOCALITATE VALUES (9, 'BV', 'Brasov');

INSERT INTO LOCALITATE VALUES (10, 'BV', 'Sinaia');

INSERT INTO LOCALITATE VALUES (11, 'CJ', 'Cluj Napoca');

commit;



-- ##########################################

CREATE TABLE STUDII

(ID\_STUDII VARCHAR2(1) PRIMARY KEY,

DENUMIRE VARCHAR2(20) NOT NULL

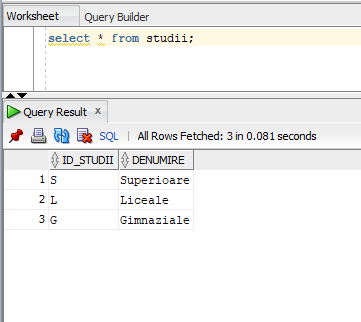
);

COMMIT;

INSERT INTO STUDII VALUES ('S','Superioare');

INSERT INTO STUDII VALUES ('L','Liceale');

INSERT INTO STUDII VALUES ('G','Gimnaziale');



-- ##########################################

CREATE TABLE FUNCTIE

(ID\_FUNCTIE NUMBER(2) PRIMARY KEY,

DENUMIRE VARCHAR2(20) NOT NULL

);

COMMIT;

INSERT INTO FUNCTIE VALUES (1,'Agent');

INSERT INTO FUNCTIE VALUES (2,'Contabil');

INSERT INTO FUNCTIE VALUES (3,'Director');

INSERT INTO FUNCTIE VALUES (4,'Sofer');

INSERT INTO FUNCTIE VALUES (5,'Secretara');

INSERT INTO FUNCTIE VALUES (6,'Ingrijitor');

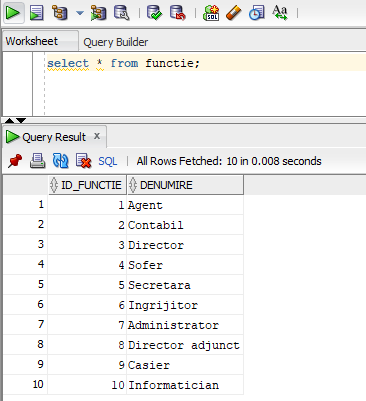
INSERT INTO FUNCTIE VALUES (7,'Administrator');

INSERT INTO FUNCTIE VALUES (8,'Director adjunct');

INSERT INTO FUNCTIE VALUES (9,'Casier');

INSERT INTO FUNCTIE VALUES (10,'Informatician');

commit;



-- ##########################################

CREATE TABLE OPERATIE

(ID\_OPERATIE VARCHAR2(1) PRIMARY KEY,

DENUMIRE VARCHAR2(20) NOT NULL

);

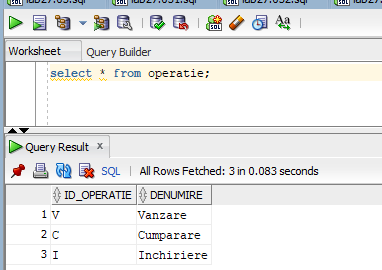
COMMIT;

INSERT INTO OPERATIE VALUES ('V','Vanzare');

INSERT INTO OPERATIE VALUES ('C','Cumparare');

INSERT INTO OPERATIE VALUES ('I','Inchiriere');

commit;



-- ##########################################

CREATE TABLE PROPRIETATE

(ID\_PROPRIETATE VARCHAR2(1) PRIMARY KEY,

DENUMIRE VARCHAR2(20) NOT NULL

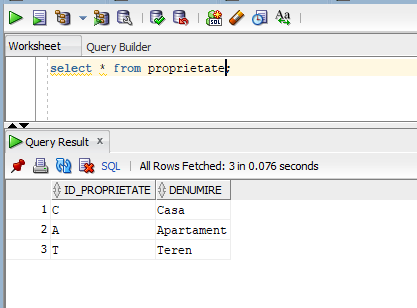
);

COMMIT;

INSERT INTO PROPRIETATE VALUES ('C','Casa');

INSERT INTO PROPRIETATE VALUES ('A','Apartament');

INSERT INTO PROPRIETATE VALUES ('T','Teren');

commit;

-- ##########################################

CREATE TABLE ANGAJAT

(Id\_angajat Number(3) PRIMARY KEY,

Id\_functie Number(2) Not null,

Nume Varchar2(20) Not null,

Prenume Varchar2(20) Not null,

Cnp Varchar2(13) Not null,

Adresa Varchar2(30) Not null,

Telefon Varchar2(10) Not null,

email Varchar2(20) Not null,

Cont Varchar2(20) Not null,

parola Varchar2(20) Not null

);

COMMIT;

ALTER TABLE ANGAJAT

ADD CONSTRAINT FKFUNCTIE FOREIGN KEY

(

ID\_FUNCTIE

)

REFERENCES FUNCTIE

(

ID\_FUNCTIE

);

COMMIT;

INSERT INTO ANGAJAT VALUES(1,1,'Ionescu','Vasile','1710612370028','Str. Fagului Nr. 2', '075142561','ionescuv@fx.ro','vionescu','sdkcn');

INSERT INTO ANGAJAT VALUES(2,1,'Popescu','Ionut','1710612370828','Str. Carpenului Nr. 3', '0751427140','popescui@fx.ro','ipopescu','sdkln');

INSERT INTO ANGAJAT VALUES(3,1,'Chirita','Maria','2710912370028','Str. Marului Nr. 12', '075199561','chiritam@fx.ro','mchirita','sdkcnio');

INSERT INTO ANGAJAT VALUES(4,1,'Mincu','Ion','1710612373028','Str. Stejarului Nr. 2', '072142561','mincui@fx.ro','imincu','sdttttkcn');

INSERT INTO ANGAJAT VALUES(5,2,'Cornea','Doina','2710618870028','Str. Macinului Nr. 28', '071192561','cornead@fx.ro','dcornea','aaadkcn');

INSERT INTO ANGAJAT VALUES(6,3,'Ciubotaru','Cornel','1710616660028','Str. Pacii Nr. 101', '075142001','ciubotaruc@fx.ro','cciubotaru','siiidkcn');

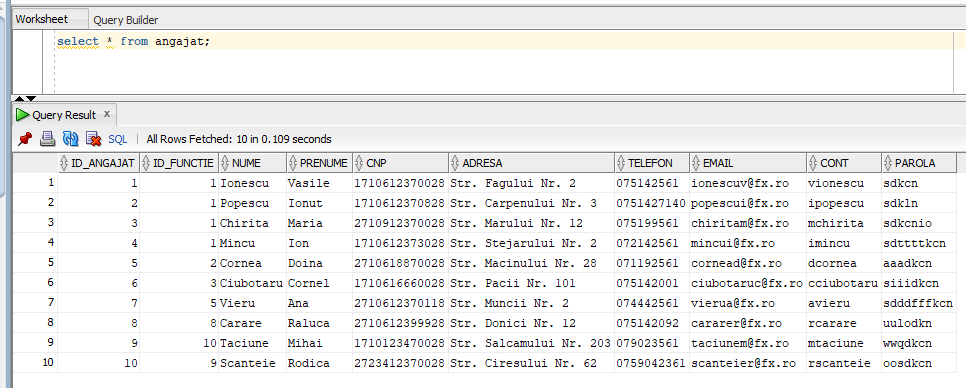
INSERT INTO ANGAJAT VALUES(7,5,'Vieru','Ana','2710612370118','Str. Muncii Nr. 2', '074442561','vierua@fx.ro','avieru','sdddfffkcn');

INSERT INTO ANGAJAT VALUES(8,8,'Carare','Raluca','2710612399928','Str. Donici Nr. 12', '075142092','cararer@fx.ro','rcarare','uulodkn');

INSERT INTO ANGAJAT VALUES(9,10,'Taciune','Mihai','1710123470028','Str. Salcamului Nr. 203', '079023561','taciunem@fx.ro','mtaciune','wwqdkcn');

INSERT INTO ANGAJAT VALUES(10,9,'Scanteie','Rodica','2723412370028','Str. Ciresului Nr. 62', '0759042361','scanteier@fx.ro','rscanteie','oosdkcn');

commit;



-- ##########################################

CREATE TABLE ANG\_STUDII

(Id\_angajat Number(3) Not null,

Id\_studii Varchar2(1) Not null

);

ALTER TABLE ANG\_STUDII ADD (

CONSTRAINT ANG\_STUD\_PK PRIMARY KEY (ID\_ANGAJAT,ID\_STUDII),

CONSTRAINT AS\_ID\_ANGAJAT\_FK FOREIGN KEY (ID\_ANGAJAT) REFERENCES ANGAJAT(ID\_ANGAJAT),

CONSTRAINT AS\_ID\_SUDII\_FK FOREIGN KEY (ID\_STUDII) REFERENCES STUDII(ID\_STUDII)

);

commit;

INSERT INTO ANG\_STUDII VALUES(1,'S');

INSERT INTO ANG\_STUDII VALUES(2,'S');

INSERT INTO ANG\_STUDII VALUES(3,'S');

INSERT INTO ANG\_STUDII VALUES(4,'S');

INSERT INTO ANG\_STUDII VALUES(5,'S');

INSERT INTO ANG\_STUDII VALUES(6,'S');

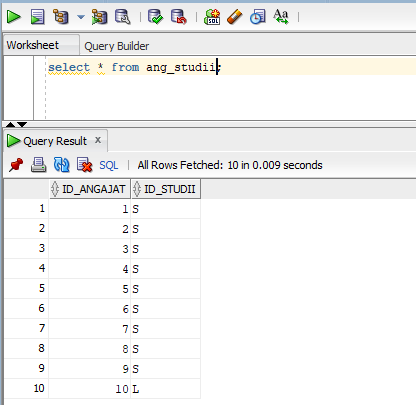
INSERT INTO ANG\_STUDII VALUES(7,'S');

INSERT INTO ANG\_STUDII VALUES(8,'S');

INSERT INTO ANG\_STUDII VALUES(9,'S');

INSERT INTO ANG\_STUDII VALUES(10,'L');

commit;



-- ##########################################

CREATE TABLE CLIENT

(Id\_client Number(5) primary key,

Nume Varchar2(20) Not null,

Prenume Varchar2(20) Not null,

Cnp Varchar2(13) Not null,

Adresa Varchar2(30) Not null,

Telefon Varchar2(10) Not null,

email Varchar2(20)

);

COMMIT;

INSERT into client VALUES (1,'Cimpoesu','Adrian','1720318374050','Str. Ion Creanga Nr. 4', '0761234518','acimpoesu@gmail.com');

INSERT into client VALUES (2,'Margineanu','Cosmin','1550318374050','Str. Pacii Nr. 4', '0721277518','margcos@gmail.com');

INSERT into client VALUES (3,'Purice','Adrian','1720318374234','Str. Muntelui Nr. 89', '0761890518','apurice@gmail.com');

INSERT into client VALUES (4,'Tarnoveanu','Costel','1720312174050','Str. Soarelui Nr. 5', '0788734518','costelt@gmail.com');

INSERT into client VALUES (5,'Maciuca','Tina','2720311234050','Str. Calugareni Nr. 74', '0761234255','tinamac@gmail.com');

INSERT into client VALUES (6,'Caragea','Adina','2730318974050','Str. Bratianu Nr. 101', '0761231234','adinuta@gmail.com');

INSERT into client VALUES (7,'Frunza','George','1751018374050','Str. Tezaurului Nr. 18', '0761480518','gfrunza@gmail.com');

INSERT into client VALUES (8,'Bulbuc','Elena','2720318366450','Str. Liliacului Nr. 90', '0791560518','bulbuce@gmail.com');

INSERT into client VALUES (9,'Pricope','Diana','2721295374050','Str. Bucuresti Nr. 49', '0712384518','pricopedia@gmail.com');

INSERT into client VALUES (10,'Stoica','Beatrice','2725140774050','Str. Traian Nr. 10', '0761291508','beastoi@gmail.com');

INSERT into client VALUES ('11','Modoranu','Florin','1700318374050','Str. Crizantemelor 3', '0746123451','fmodo@gmail.com');

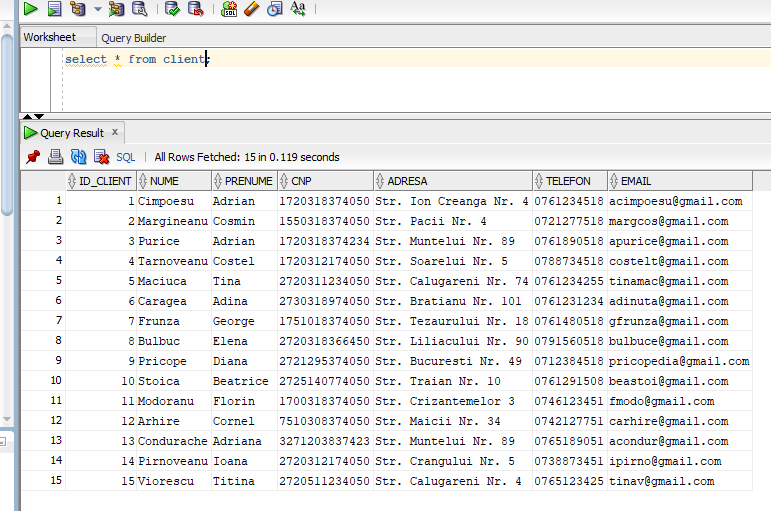
INSERT into client VALUES ('12','Arhire','Cornel','7510308374050','Str. Maicii Nr. 34', '0742127751','carhire@gmail.com');

INSERT into client VALUES ('13','Condurache','Adriana','3271203837423','Str. Muntelui Nr. 89', '0765189051','acondur@gmail.com');

INSERT into client VALUES ('14','Pirnoveanu','Ioana','2720312174050','Str. Crangului Nr. 5', '0738873451','ipirno@gmail.com');

INSERT into client VALUES ('15','Viorescu','Titina','2720511234050','Str. Calugareni Nr. 4', '0765123425','tinav@gmail.com');

COMMIT;



-- ##########################################

CREATE TABLE OFERTA

(Id\_oferta Number(5) PRIMARY KEY,

Id\_client Number(5) Not null,

Id\_operatie Varchar2(1) Not null,

Id\_proprietate Varchar2(1) Not null,

Id\_localitate Number(10) Not null,

Strada Varchar2(20) Not null,

Suprafata Number(10,2) CHECK (SUPRAFATA >0),

An\_constructie Number(4) CHECK (An\_constructie >0),

Pret Number(10) CHECK (Pret >0),

Descriere Varchar2(100),

Numar Varchar2(5)

);

COMMIT;

ALTER TABLE oferta

ADD CONSTRAINT FKCLIENT FOREIGN KEY

(

ID\_CLIENT

)

REFERENCES CLIENT

(

ID\_CLIENT

);

ALTER TABLE oferta

ADD CONSTRAINT FKOPERATIE FOREIGN KEY

(

ID\_OPERATIE

)

REFERENCES OPERATIE

(

ID\_OPERATIE

);

ALTER TABLE oferta

ADD CONSTRAINT FKPROPRIETATE FOREIGN KEY

(

ID\_PROPRIETATE

)

REFERENCES PROPRIETATE

(

ID\_PROPRIETATE

);

ALTER TABLE oferta

ADD CONSTRAINT FKLOCALITATE FOREIGN KEY

(

ID\_LOCALITATE

)

REFERENCES LOCALITATE

(

ID\_LOCALITATE

);

COMMIT;

INSERT INTO OFERTA VALUES (1,1,'V','A',1,'Pinului', 150,1999,20000,'Mobilat','3');

INSERT INTO OFERTA VALUES (2,3,'I','C',2,'Stejar', 300,2001,70000,'Mobilat','6');

INSERT INTO OFERTA VALUES (3,2,'V','C',5,'Pacii', 700,2004,100000,'Mobilat','43');

INSERT INTO OFERTA VALUES (4,5,'V','T',6,'Muncii', 500,1995,10000,'Mobilat','89');

INSERT INTO OFERTA VALUES (5,6,'I','A',4,'Teiului', 200,1997,40000,'Mobilat','13');

INSERT INTO OFERTA VALUES (6,3,'V','A',9,'Tecuci', 150,2010,50000,'Mobilat','90');

INSERT INTO OFERTA VALUES (7,4,'V','C',5,'Florilor', 800,2004,100000,'La rosu','50');

INSERT INTO OFERTA VALUES (8,8,'V','T',6,'Smardan', 500,1995,10000,'Mobilat','56');

INSERT INTO OFERTA VALUES (9,7,'I','A',4,'Donici', 100,1997,40000,'Nemobilat','13');

INSERT INTO OFERTA VALUES (10,9,'V','A',9,'Unirii', 150,2010,50000,'Mobilat','100');

INSERT INTO OFERTA VALUES (11,10,'V','A',7,'Pinului', 180,1999,20000,null,'3');

INSERT INTO OFERTA VALUES (12,11,'V','C',2,'Stejar', 300,2001,70000,'Mobilat','6');

INSERT INTO OFERTA VALUES (13,12,'V','C',2,'Pacii', 700,2004,100000,'Mobilat','43');

INSERT INTO OFERTA VALUES (14,13,'V','A',2,'Muncii', 500,1995,10000,'Mobilat','89');

INSERT INTO OFERTA VALUES (15,14,'V','A',7,'Teiului', 200,1997,40000,'Mobilat','13');

INSERT INTO OFERTA VALUES (16,15,'V','A',9,'Tecuci', 150,2010,55000,'Mobilat','90');

INSERT INTO OFERTA VALUES (17,6,'V','C',7,'Florilor', 800,2004,100000,'La rosu','50');

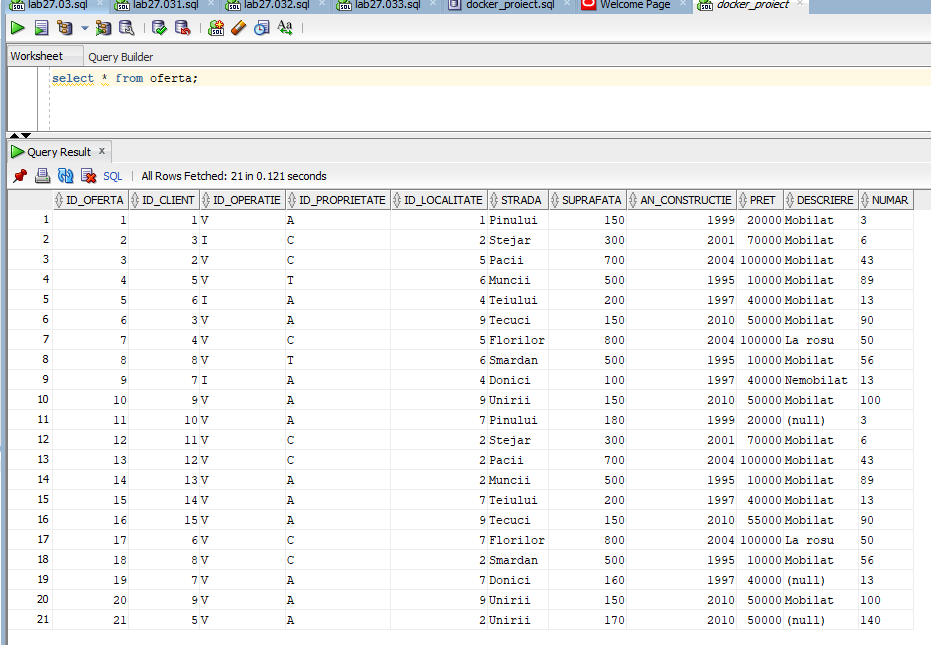
INSERT INTO OFERTA VALUES (18,8,'V','C',2,'Smardan', 500,1995,10000,'Mobilat','56');

INSERT INTO OFERTA VALUES (19,7,'V','A',7,'Donici', 160,1997,40000,null,'13');

INSERT INTO OFERTA VALUES (20,9,'V','A',9,'Unirii', 150,2010,50000,'Mobilat','100');

INSERT INTO OFERTA VALUES (21,5,'V','A',2,'Unirii', 170,2010,50000,null,'140');

COMMIT;



-- ##########################################

CREATE TABLE CERERE

( Id\_cerere Number(5) PRIMARY KEY,

Id\_client Number(5) Not null,

Id\_operatie Varchar2(1) Not null,

Id\_proprietate Varchar2(1) Not null,

Id\_localitate Number(10) Not null,

Buget Number(10) check( buget>0),

Suprafata\_minima Number(10,2) CHECK(SUPRAFATA\_MINIMA>0),

An\_construcTie Number(4) CHECK(AN\_CONSTRUCTIE>0)

);

COMMIT;

ALTER TABLE CERERE

ADD CONSTRAINT FKCLIENT2 FOREIGN KEY

(

ID\_CLIENT

)

REFERENCES CLIENT

(

ID\_CLIENT

);

ALTER TABLE CERERE

ADD CONSTRAINT FKOPERATIE2 FOREIGN KEY

(

ID\_OPERATIE

)

REFERENCES OPERATIE

(

ID\_OPERATIE

);

ALTER TABLE CERERE

ADD CONSTRAINT FKPROPRIETATE2 FOREIGN KEY

(

ID\_PROPRIETATE

)

REFERENCES PROPRIETATE

(

ID\_PROPRIETATE

);

ALTER TABLE CERERE

ADD CONSTRAINT FKLOCALITATE2 FOREIGN KEY

(

ID\_LOCALITATE

)

REFERENCES LOCALITATE

(

ID\_LOCALITATE

);

COMMIT;

INSERT INTO cerere VALUES (1,1,'C','A',1,20000,150,2021);

INSERT INTO cerere VALUES (2,3,'I','C',3,80000,300,2001);

INSERT INTO cerere VALUES (3,8,'C','C',7,100000, 700,2004);

INSERT INTO cerere VALUES (4,10,'C','T',6,100000, 500,2020);

INSERT INTO cerere VALUES (5,2,'I','A',4,40000, 200,2000);

INSERT INTO cerere VALUES (6,7,'C','A',9,50000, 150,2010);

INSERT INTO cerere VALUES (7,4,'C','C',2,100000, 700,2018);

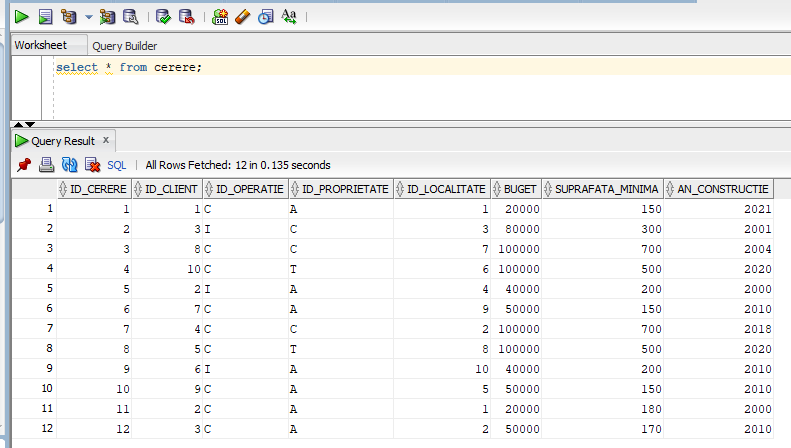
INSERT INTO cerere VALUES (8,5,'C','T',8,100000, 500,2020);

INSERT INTO cerere VALUES (9,6,'I','A',10,40000, 200,2010);

INSERT INTO cerere VALUES (10,9,'C','A',5,50000, 150,2010);

INSERT INTO cerere VALUES (11,2,'C','A',1,20000,180,2000);

INSERT INTO cerere VALUES (12,3,'C','A',2,50000,170,2010);

COMMIT;

-- ##########################################

CREATE TABLE CONTRACT

(Id\_contract Number(5) PRIMARY KEY,

Id\_angajat Number(3) Not null,

Id\_client Number(5) Not null,

Data Date DEFAULT (SYSDATE),

Comision Number(2) CHECK(COMISION>0)

);

COMMIT;

ALTER TABLE CONTRACT

ADD CONSTRAINT FKANGAJAT3 FOREIGN KEY

(

ID\_ANGAJAT

)

REFERENCES ANGAJAT

(

ID\_ANGAJAT

);

ALTER TABLE CONTRACT

ADD CONSTRAINT FKCLIENT3 FOREIGN KEY

(

ID\_CLIENT

)

REFERENCES CLIENT

(

ID\_CLIENT

);

COMMIT;

INSERT INTO contract VALUES (1,2,1,SYSDATE,10);

INSERT INTO contract VALUES (2,4,2,SYSDATE,10);

INSERT INTO contract VALUES (3,4,6,SYSDATE,10);

INSERT INTO contract VALUES (4,4,7,SYSDATE,5);

INSERT INTO contract VALUES (5,2,2,SYSDATE,10);

INSERT INTO contract VALUES (6,3,7,SYSDATE,10);

INSERT INTO contract VALUES (7,3,10,SYSDATE,10);

INSERT INTO contract VALUES (8,4,2,SYSDATE,20);

INSERT INTO contract VALUES (9,4,6,SYSDATE,10);

INSERT INTO contract VALUES (10,1,1,SYSDATE,5);

INSERT INTO contract VALUES (11,1,11,SYSDATE,10);

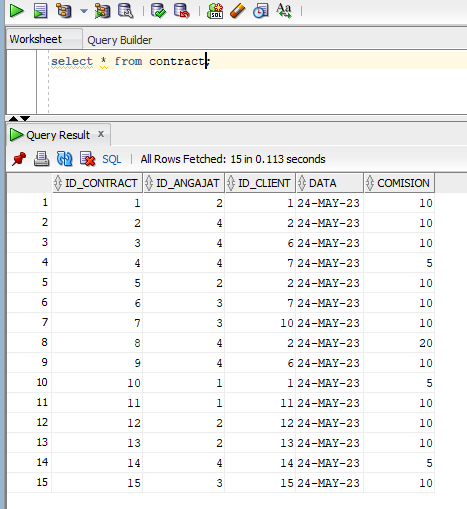
INSERT INTO contract VALUES (12,2,12,SYSDATE,10);

INSERT INTO contract VALUES (13,2,13,SYSDATE,10);

INSERT INTO contract VALUES (14,4,14,SYSDATE,5);

INSERT INTO contract VALUES (15,3,15,SYSDATE,10);

COMMIT;



## Formulați în limbaj natural și implementați 5 cereri SQL

1. **Afisati clienții care au oferit spre vânzare proprietăți cu prețul mai mic decât media prețurilor proprietăților de același tip din aceeași localitate, oferta pentru care s-a încheiat deja o tranzacție în ultimele 3 luni** (subcereri corelate în care intervin 4 tabele: OFERTA, CLIENTI, LOCALITATE, TRANZACTIE; funcții pentru date calendaristice ADD\_MONTHS, SYSDATE)

SELECT C.NUME, C.PRENUME, O.ID\_PROPRIETATE, L.DENUMIRE LOC, O.PRET

FROM OFERTA O

JOIN CLIENT C ON O.ID\_CLIENT=C.ID\_CLIENT

JOIN LOCALITATE L ON O.ID\_LOCALITATE=L.ID\_LOCALITATE

WHERE O.ID\_OPERATIE='V' AND O.PRET <

( SELECT AVG(PRET)

FROM OFERTA O1

WHERE O.ID\_OPERATIE='V' AND O1.ID\_PROPRIETATE=O.ID\_PROPRIETATE AND

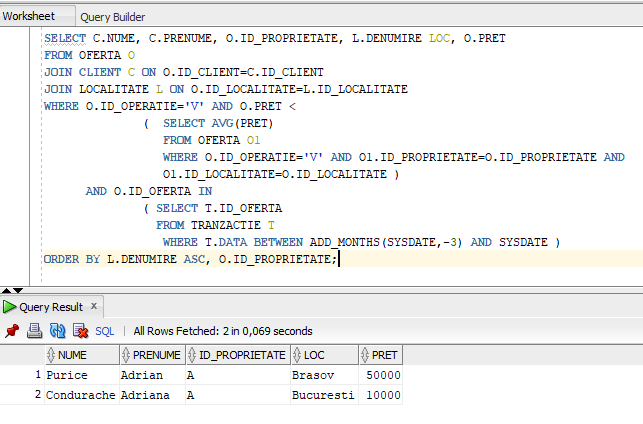
O1.ID\_LOCALITATE=O.ID\_LOCALITATE )

AND O.ID\_OFERTA IN

( SELECT T.ID\_OFERTA

FROM TRANZACTIE T

WHERE T.DATA BETWEEN ADD\_MONTHS(SYSDATE,-3) AND SYSDATE )

ORDER BY L.DENUMIRE ASC, O.ID\_PROPRIETATE;

1. **Afișați numele și funcția angajaților care au încheiat cele mai multe tranzacții** (3 subcereri necorelate din care una in clauza FROM, funcții grup, filtrare la nivel de grup)

SELECT A.NUME, A.PRENUME, F.DENUMIRE FUNCTIA, TR.NR\_TRANZ

FROM ANGAJAT A

JOIN FUNCTIE F ON A.ID\_FUNCTIE=F.ID\_FUNCTIE

JOIN (SELECT T.ID\_ANGAJAT, COUNT(T.ID\_ANGAJAT) AS NR\_TRANZ

FROM TRANZACTIE T

GROUP BY T.ID\_ANGAJA ) TR ON A.ID\_ANGAJAT=TR.ID\_ANGAJAT

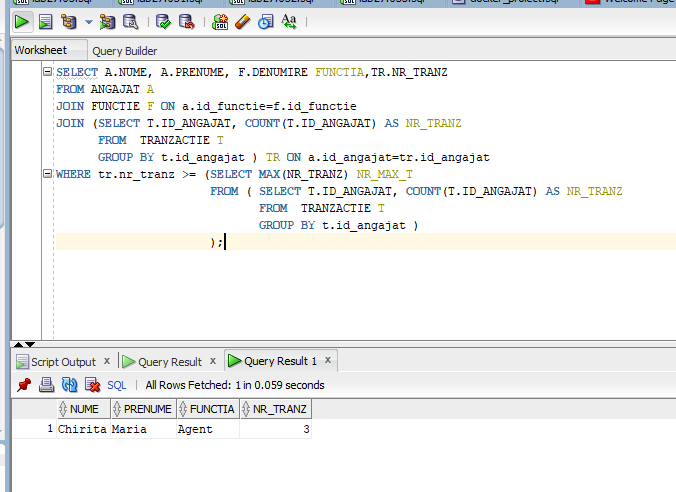
WHERE TR.NR\_TRANZ >=

(SELECT MAX(NR\_TRANZ) NR\_MAX\_T

FROM ( SELECT T.ID\_ANGAJAT, COUNT(T.ID\_ANGAJAT) AS NR\_TRANZ

FROM TRANZACTIE T

GROUP BY T.ID\_ANGAJAT

) );

1. **Afișați ofertele existente în acest moment în ordine pe localități, tip operație, tip proprietate, stradă** (subcerere corelată OFERTA cu TRANZACTIE in clauza WHERE / utilizare NVL si DECODE / funcții pe șiruri de caractere TRIM si INITCAP / ordonare)

SELECT J.ID\_JUDET JUD, L.DENUMIRE LOC, INITCAP(TRIM(C.NUME)) || ' ' ||

INITCAP(TRIM(C.PRENUME)) OFERTANT, STRADA, NUMAR, SUPRAFATA, AN\_CONSTRUCTIE, PRET, ID\_OPERATIE OP, DECODE(ID\_PROPRIETATE,'C','CASA','A','APARTAMENT','T','TEREN') TIP\_PROPR, NVL(O.DESCRIERE,'') DESCR

FROM OFERTA O

JOIN CLIENT C ON O.ID\_CLIENT=C.ID\_CLIENT

JOIN LOCALITATE L ON O.ID\_LOCALITATE=L.ID\_LOCALITATE

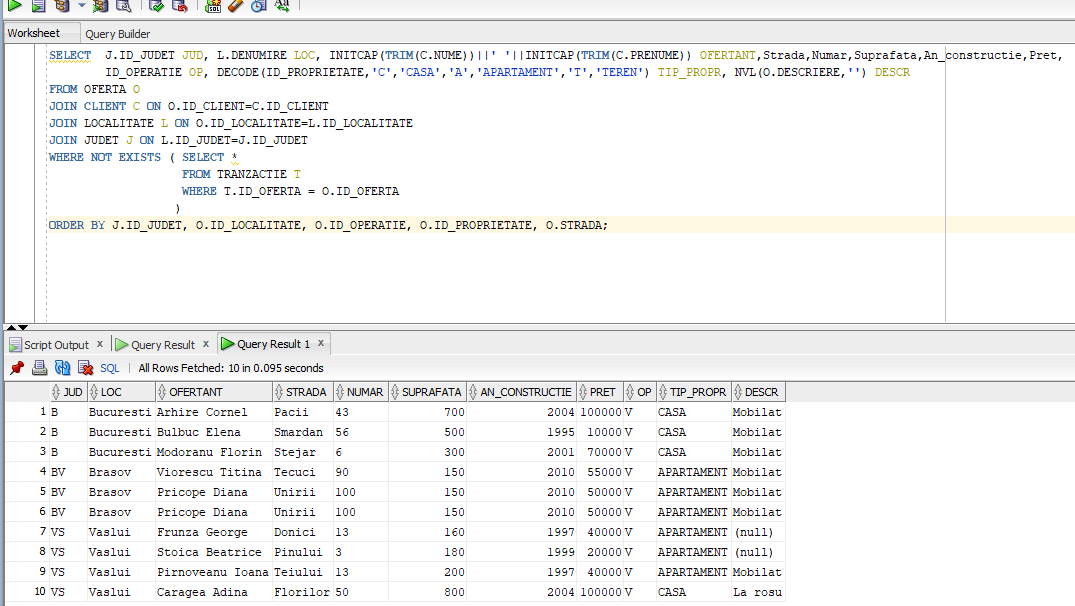
JOIN JUDET J ON L.ID\_JUDET=J.ID\_JUDET

WHERE NOT EXISTS ( SELECT \* FROM TRANZACTIE T

WHERE T.ID\_OFERTA = O.ID\_OFERTA

)

ORDER BY J.ID\_JUDET, O.ID\_LOCALITATE, O.ID\_OPERATIE, O.ID\_PROPRIETATE, O.STRADA;



1. **Ofertele de vânzare case pe categorii de preț din 'Bucuresti' (**CASE**)**

SELECT

CASE

WHEN O.PRET > 0 AND O.PRET < 50000

THEN 'Ieftin'

WHEN O.PRET >= 50000 AND O.PRET < 100000

THEN 'Accesibil'

WHEN O.PRET >= 100000 AND O.PRET < 150000

THEN 'Scump'

WHEN O.PRET >= 150000 AND O.PRET < 200000

THEN 'Lux'

ELSE

'Mare Lux'

END CATEGORIA , O.PRET, O.Suprafata, O.An\_constructie, C.NUME, C.PRENUME, Strada, Numar

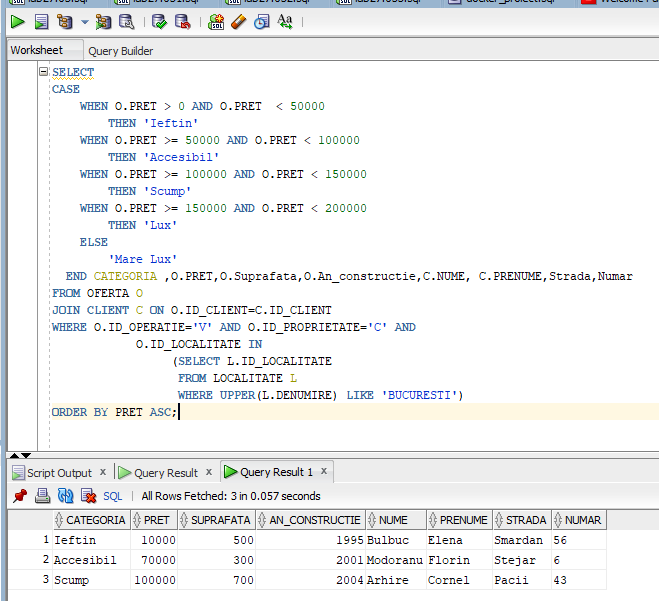
FROM OFERTA O

JOIN CLIENT C ON O.ID\_CLIENT=C.ID\_CLIENT

WHERE O.ID\_OPERATIE='V' AND O.ID\_PROPRIETATE='C' AND O.ID\_LOCALITATE IN

(SELECT L.ID\_LOCALITATE FROM LOCALITATE L WHERE UPPER(L.DENUMIRE) LIKE 'BUCURESTI')

ORDER BY PRET ASC;



1. **Cererile de cumpărare case cu buget mai mare sau egal decât media prețului caselor oferite (**clauzaWITH)

WITH AVG\_OFERTA

AS (SELECT AVG(PRET) AVGP FROM OFERTA WHERE ID\_PROPRIETATE='C' AND ID\_OPERATIE='V')

SELECT CL.NUME, CL.PRENUME , L.DENUMIRE LOC, Suprafata\_minima, An\_constructie, C.BUGET,

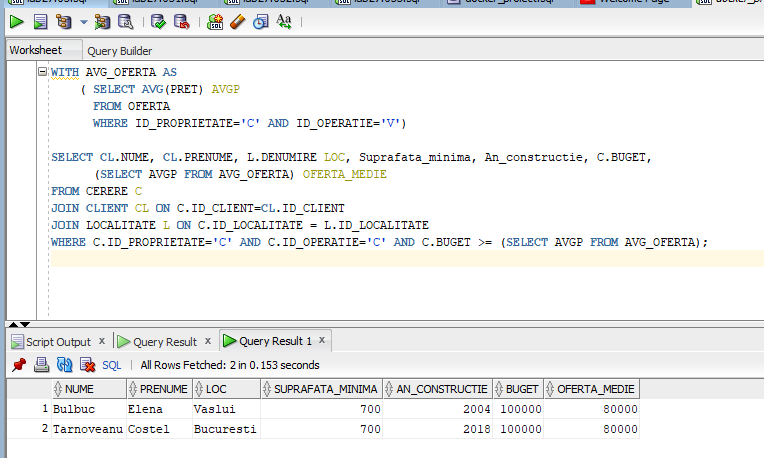
(SELECT AVGP FROM AVG\_OFERTA) OFERTA\_MEDIE

FROM CERERE C

JOIN CLIENT CL ON C.ID\_CLIENT=CL.ID\_CLIENT

JOIN LOCALITATE L ON C.ID\_LOCALITATE = L.ID\_LOCALITATE

WHERE C.ID\_PROPRIETATE='C' AND C.ID\_OPERATIE='C' AND C.BUGET >= (SELECT AVGP FROM AVG\_OFERTA);



## Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri

***ACTUALIZĂRI***

1. **Modificarea parolei pentru toți angajații cu funcția informatician**

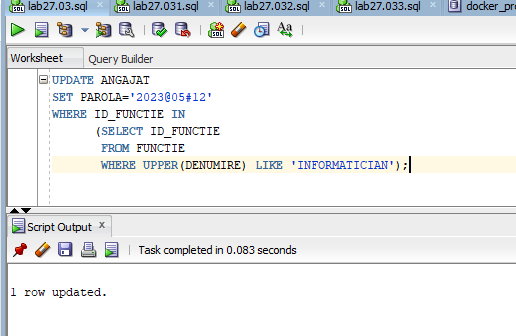
UPDATE ANGAJAT

SET PAROLA='2023@05#12'

WHERE ID\_FUNCTIE IN

(SELECT ID\_FUNCTIE

FROM FUNCTIE

 WHERE UPPER(DENUMIRE) LIKE 'INFORMATICIAN');

1. **Reducerea comisionului cu 2% pt. clienții care au mai multe contracte**

UPDATE CONTRACT

SET COMISION=(COMISION-2)

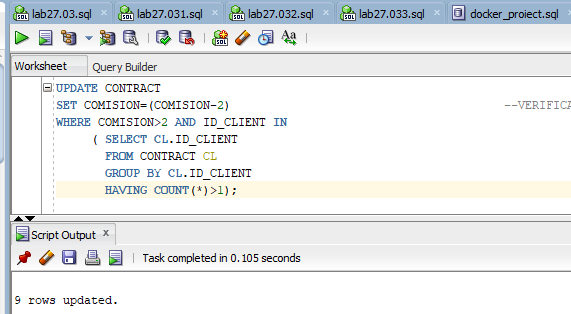
WHERE COMISION>2 AND ID\_CLIENT IN

(SELECT CL.ID\_CLIENT

FROM CONTRACT CL

GROUP BY CL.ID\_CLIENT

HAVING COUNT(\*)>1);



1. **Completarea automată în OFERTE a câmpului DESCRIERE cu un prefix de genul *#Inchiriere Loc. Vaslui - Casa Suprafata 500 mp, construit dupa 2000#* pentru apartamentele sau casele din localitatea Vaslui care nu au DESCRIERE.**

UPDATE OFERTA O

SET O.DESCRIERE= '#'||DECODE(ID\_OPERATIE,'V','Vanzare ','I','Inchiriere ') || 'Loc. Vaslui - ' ||

DECODE(ID\_PROPRIETATE,'C','CASA','A','APARTAMENT','T','TEREN') || ' Suprafata ' ||

TO\_CHAR(SUPRAFATA,'999999.99') || ' mp, construit ' || CASE WHEN AN\_CONSTRUCTIE>=2000

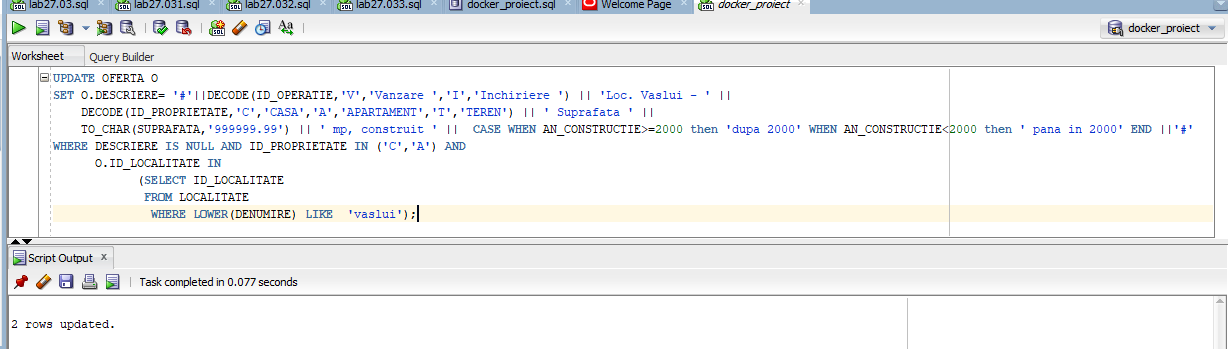
THEN 'dupa 2000' WHEN AN\_CONSTRUCTIE<2000 then ' pana in 2000' END ||'#'

WHERE DESCRIERE IS NULL AND ID\_PROPRIETATE IN ('C','A') AND O.ID\_LOCALITATE IN

(SELECT ID\_LOCALITATE

FROM LOCALITATE

WHERE LOWER(DENUMIRE) LIKE 'vaslui');

******

**SUPRIMĂRI**

1. **Ștergerea tranzacțiilor încheiate cu un client care nu are contract încheiat cu agenția.**

DELETE FROM TRANZACTIE

WHERE ID\_CERERE IN

(SELECT CR.ID\_CERERE

FROM CERERE CR

WHERE NOT EXISTS

(SELECT CT.ID\_CLIENT

FROM CONTRACT CT

WHERE CT.ID\_CLIENT=CR.ID\_CLIENT ))

OR ID\_OFERTA IN

( SELECT OFR.ID\_OFERTA

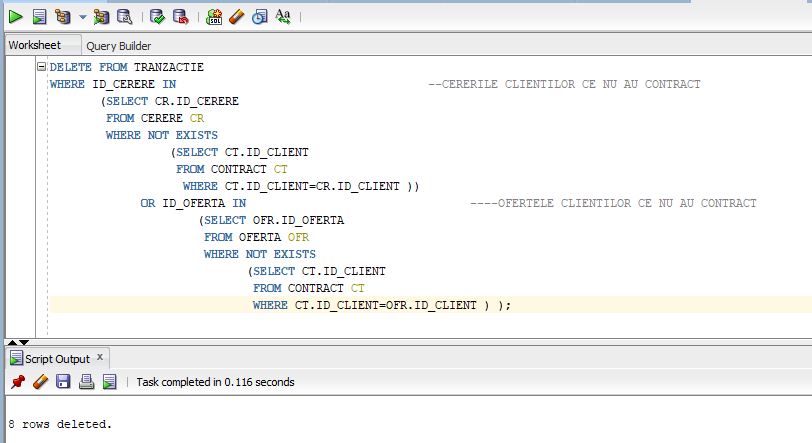
FROM OFERTA OFR

WHERE NOT EXISTS

(SELECT CT.ID\_CLIENT

FROM CONTRACT CT

WHERE CT.ID\_CLIENT=OFR.ID\_CLIENT ));



1. **Ștergerea ofertelor clienților ce nu au contract încheiat cu agenția (**are sens doar dacă s-a rulat cererea anterioara, ștergerea tranzacțiilor încheiate cu un client care nu are contract încheiat cu agenția, deoarece altfel apare eroare constrângere cheie externă)

DELETE FROM OFERTA

WHERE ID\_OFERTA IN

( SELECT OFR.ID\_OFERTA

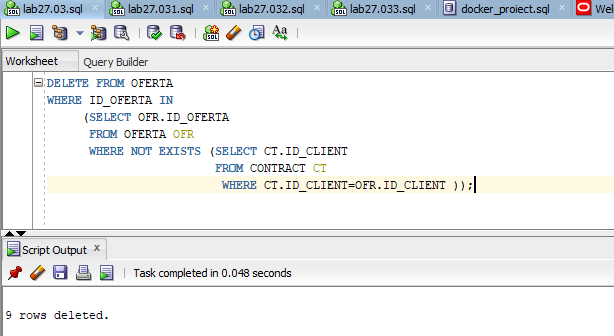
FROM OFERTA OFR

WHERE NOT EXISTS

( SELECT CT.ID\_CLIENT

FROM CONTRACT CT

WHERE CT.ID\_CLIENT=OFR.ID\_CLIENT ));



1. **Ștergerea cererilor clienților ce nu au contract încheiat cu agenția (**are sens doar dacă s-a rulat cererea anterioara, ștergerea tranzacțiilor încheiate cu un client care nu are contract încheiat cu agenția, deoarece altfel apare eroare constrângere cheie externă)

DELETE FROM CERERE

WHERE ID\_CERERE IN

(SELECT CR.ID\_CERERE

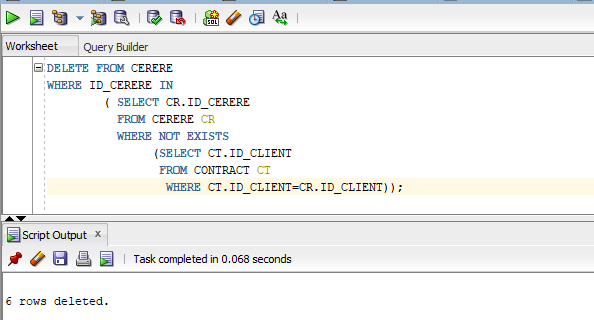
FROM CERERE CR

WHERE NOT EXISTS

(SELECT CT.ID\_CLIENT

FROM CONTRACT CT

WHERE CT.ID\_CLIENT=CR.ID\_CLIENT));



## Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.

## Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outerjoin pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n

## Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebrică, arbore algebric și limbaj (SQL), atât anterior cât și ulterior optimizării

## Realizarea normalizării BCNF, FN4, FN5. Aplicarea denormalizării, justificând necesitatea acesteia.

Cuprins

[1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare 1](#_Toc136023260)

[2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului 2](#_Toc136023261)

[3. Descrierea entităților, incluzând precizarea cheii primare 2](#_Toc136023262)

[4. Descrierea relațiilor, incluzând precizarea cardinalității acestora 4](#_Toc136023263)

[5. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor 5](#_Toc136023264)

[6. Realizarea diagramei entitate-relație 7](#_Toc136023265)

[7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. 8](#_Toc136023266)

[8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7. 9](#_Toc136023267)

[9. Realizarea normalizării până la forma normală 3 9](#_Toc136023268)

[10. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele 12](#_Toc136023269)

[11. Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea 13](#_Toc136023270)

[12. Formulați în limbaj natural și implementați 5 cereri SQL 26](#_Toc136023271)

[13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri 29](#_Toc136023272)

[14. Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă. 32](#_Toc136023273)

[15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outerjoin pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n 32](#_Toc136023274)

[16. Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebrică, arbore algebric și limbaj (SQL), atât anterior cât și ulterior optimizării 32](#_Toc136023275)

[17. Realizarea normalizării BCNF, FN4, FN5. Aplicarea denormalizării, justificând necesitatea acesteia. 33](#_Toc136023276)