GESTIUNEA UNEI FIRME DE DEZVOLTARE DE JOCURI

Maftei Ștefan

Grupa 142

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

O bază de date a unei firme de dezvoltare de jocuri ar putea fi utilizată pentru a gestiona informațiile referitoare la angajați, echipe, proiecte, departamente și alte aspecte relevante pentru activitatea companiei. Aceasta ar putea conține informații despre fiecare angajat, cum ar fi numele, poziția, experiența, abilitățile, salariul, etc. De asemenea, baza de date ar putea să includă informații despre echipele de dezvoltare, cum ar fi componența acestora și proiectele pe care le gestionează. Scopul bazei de date este de a ajuta firma să gestioneze informațiile sale într-un mod eficient și organizat, pentru a-și îndeplini obiectivele de business, să ia decizii și să optimizeze procesele. Restricțiile ar putea fi legate de protecția datelor, a securității și a confidențialității, pentru a asigura că datele sunt accesibile doar persoanelor autorizate în mod legal.

1. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.

* Fiecare angajat este membru al unei singure echipe, iar fiecare echipă are asignate proiecte. Un proiect este asignat unei singure echipe.
* Un angajat se va regăsi într-un singur departament.
* Fiecare departament are cel puțin un angajat
* Fiecare proiect asignate unui echipe are planificat un relase pentru rezultatul final.
* Indiferent de departament (HR, Vânzări, Programare) fiecare departament poate să aibe mai mulți angajați
* Fiecare angajat are o unică funcție în cadrul firmei.
* Fiecare departament are un singur director
* Fiecare director va semna contracte pentru proiecte
* Un director poate semna mai multe contracte
* Fiecare proiect are asignat un singur contract

1. Descrierea entităților, incluzând precizarea cheii primare.

Director = este responsabil cu conducerea firmei, cu semnarea de contracte. Cheia primară este ID\_DIRECTOR

Departament = reprezintă secțiile în care este împărțită firma. Cheia primară este ID\_DEPARTAMENT

Angajat = persoană care se ocupă de task-urile firmei. Cheia primară este ID\_ANGAJAT

Echipa = reprezintă un grup de angajați care se vor ocupa de proiectele firmei. Cheia primară este ID\_ECHIPA

Contract = conține detalii despre câștigurile firmei în urma unui proiect, acesta este semnat de director. Cheia primară este ID\_DIRECTOR

Proiect = conține detalii despre proiectul în desfășurare, etc. Cheia primară este ID\_PROIECT

1. Descrierea relațiilor, incluzând precizarea cardinalității acestora

* Relația dintre Departament și Angajat

Un departament poate avea asociați mai mulți angajați, iar un angajat este asociat unui singur departament.

Cardinalitate minimă 1 : 1 maximă 1 : N

* Relația dintre Departamenet și Director

Un departament poate avea un singur director, iar un director este asociat unui singur departament.

Cardinalitate 1 : 1

* Relația dintre Departament și Echipa

Un departament poate avea mai multe echipe, iar o echipă este asociată unui singur departament.

Cardinalitate N:N

* Relația dintre Echipa și Proiect.

Mai multe echipe pot avea mai multe proiecte, iar mai multe proiecte pot fi asignate mai multor echipe.

Cardinalitate 1 : 1

* Relația dintre Director și Contract

Un director poate semna mai multe contracte, iar un contract este asociat unui singur director.

Cardinalitate minimă 1 : 1 maximă 1 : N

* Relația dintre Contract și Proiect.

Un contract poatea avea mai multe proiecte, iar un proiect are un singur contract.

Cardinalitate minimă 1 : 1 maximă 1 : N

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

**Entitatea independentă Angajat:**

ID\_ANGAJAT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare angajat (PK).

NUME\_ANGAJAT: variabilă de tip caracter, de lungime maximă 24, este denumirea fiecărui angajat. (NOT NULL).

SALARIU\_ANGAJAT: variabilă de tip întreg de lungime maximă 6, reprezintă câștigul lunar pentru fiecare angajat. (NOT NULL).

DATA\_ANGAJARII: variabilă de tip calendaristică, care reprezintă data în care a fost angajată persoana respectivă (NOT NULL).

**Entitatea independentă Departament**:

ID\_DEPARTAMENT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare departament (PK).

NUME\_DEPARTAMENT: variabilă de tip caracter de lungime maximă 12, este denumirea fiecărui departament (NOT NULL).

**Entitatea independentă Director:**

ID\_DIRECTOR: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare director (PK).

NUME\_DIRECTOR: variabilă de tip caracter de lungime maximă 24, este denumirea fiecărui director (NOT NULL).

SALARIU\_DIRECTOR: variabilă de tip întreg de lungime maximă 6, reprezintă câștigul lunar pentru fiecare director (NOT NULL).

DATA\_FUNCTIE: varabilă de tip calendaristică, care reprezintă data în care angajatul respectiv a primit funcția de director (NOT NULL).

**Entitatea independentă Echipa:**

ID\_ECHIPA: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare echipă (PK).

ID\_DEPARTAMENT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare departament. (FK)

NUME\_ECHIPĂ: variabilă de tip caracter de lungime maximă 12, reprezintă denumirea fiecărei echipe (NOT NULL).

NUMAR\_TASKURI: variabilă de tip întreg de lungime maximă 4, care reprezintă numărul de task-uri care trebuiesc îndeplinite de echipă.

**Entitatea independentă Contract:**

ID\_CONTRACT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare contract (PK).

ID\_PROIECT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare proiect. Trebuie să corespundă cu cel din entitatea Contract (FK).

ID\_DIRECTOR: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare director (FK).

CASTIG\_CONTRACT: variabilă de tip întreg de lungime maximă 8, reprezintă numărul de bani pe care îi încasează firma în urma semnării contractului (NOT NULL)

DATA\_CONTRACT: variabilă de tip calendaristică, care reprezintă data în care contractul a fost semnat (NOT NULL).

**Entitatea independentă Proiect:**

ID\_PROIECT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare proiect (PK).

NUME\_PROIECT: variabilă de tip caracter de lungime maximă 20, reprezintă denumirea proiectului (NOT NULL).

STADIU\_PROIECT: variabilă de tip boolean care reprezintă stadiul proiectului (1 - dacă proiectul este terminat, iar 0 - dacă proiectul este în desfășurare) (NOT NULL).

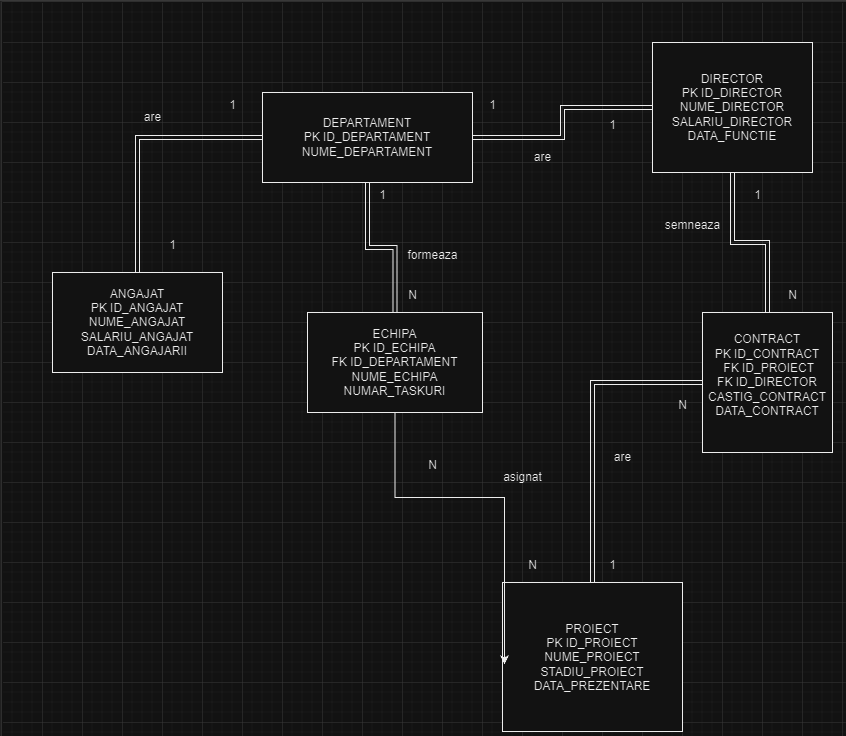
DATA\_PREZENTARE: variabilă de tip calendaristică, care reprezintă data în care proiectul trebuie prezentat (NOT NULL).

**Relația Relase (Echipa\_Proiect):**

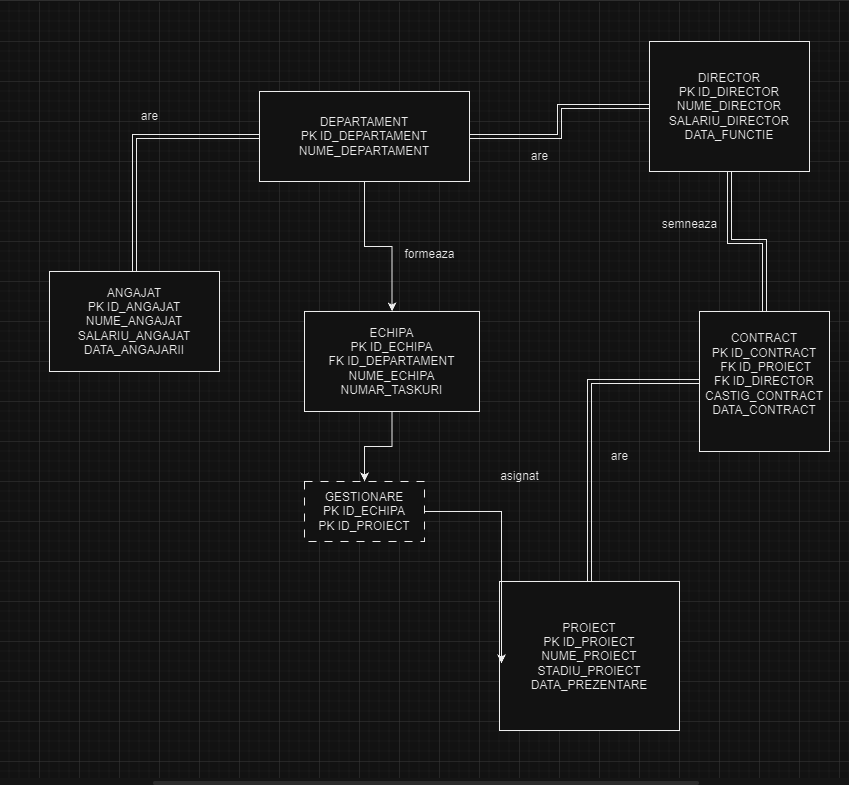
ID\_ECHIPA: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare echipă. Trebuie să corespundă cu cel puțin o valoare din entitatea Echipa (PK).

ID\_PROIECT: variabilă de tip întreg de lungime maximă 3, este id-ul de identificare pentru fiecare proiect. Trebuie să corespundă cu cel puțin o valoare din entitatea Proiect (PK).

1. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.



1. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obțiuntă trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



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

*ANGAJAT*(id\_angajat#, nume\_angajat, salariu\_angajat, data\_angajarii)

*DEPARTAMENT*(id\_departament#, nume\_departament)

*DIRECTOR*(id\_director#, nume\_director, salariu\_director, data\_functie)

*ECHIPA*(id\_echipa#, id\_departament, nume\_echipa, numar\_taskuri)

*CONTRACT*(id\_contract#, id\_proiect, id\_director, castig\_contract, data\_contract)

*RELASE*(id\_echipa#, id\_proiect#)

*PROIECT*(id\_proiect#, nume\_proiect, stadiu\_proiect, data\_prezentare)

1. Realizarea normalizării până la forma normala 3 (FN1-FN3).

FN1: Am identificat atributele atomice din tabel. Cum ar fi NUME\_ANGAJAT din tabela Angajat și l-am separat în 2 atribute separate nume și prenume sau NUME\_DIRECTOR din tabela Director și l-am separat în 2 atribute separate nume și prenume.

FN2: Normalizarea FN2 este deja făcută. Dar, de exemplu, dacă nu ar exista tabela Angajat, toate din cadrul tabelei s-ar duce în tabela Departament, iar astfel SALARIU\_ANGAJAT ar depinde tot de ID\_ANGAJAT, dar ID\_ANGAJAT nu mai există ca PK.

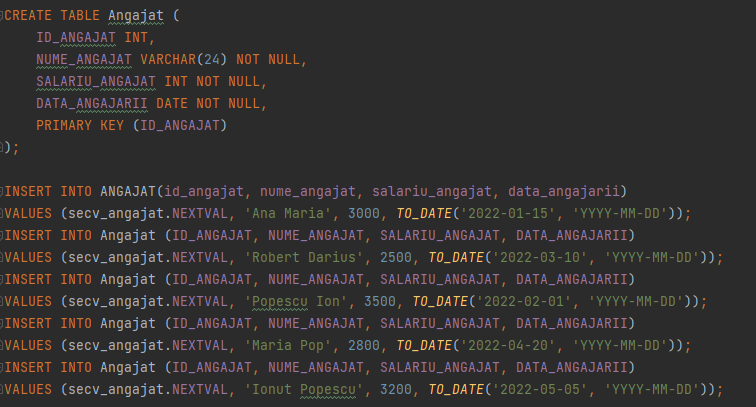
FN3: Normalizarea FN3 este deja făcută. Dar, de exemplu, dacă nu ar exista tabela Proiect și tabela Contract și am avea totul in Director, s-ar crea o relație de genul ID\_DIRECTOR determină ID\_CONTRACT care determină ID\_PROIECT, iar ID\_PROIECT determină DATA\_PREZENTARE, deci indirect ID\_DIRECTOR determină DATA\_PREZENTARE. Asta este o dependență tranzitivă care nu respectă FN3.

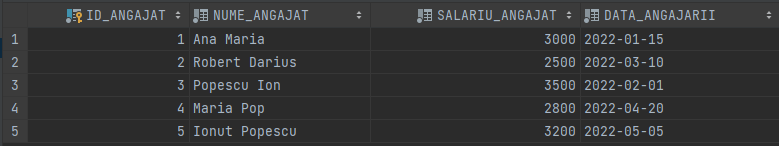
1. Creearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 11).



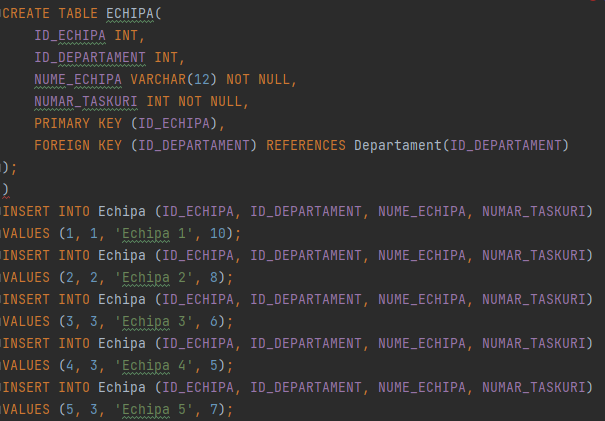
1. Creearea tabelolor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).

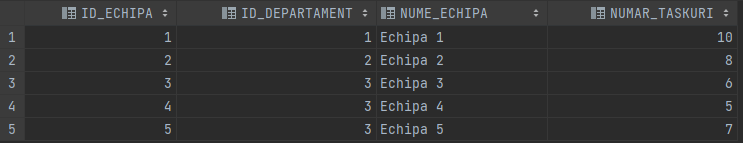
**ANGAJAT**



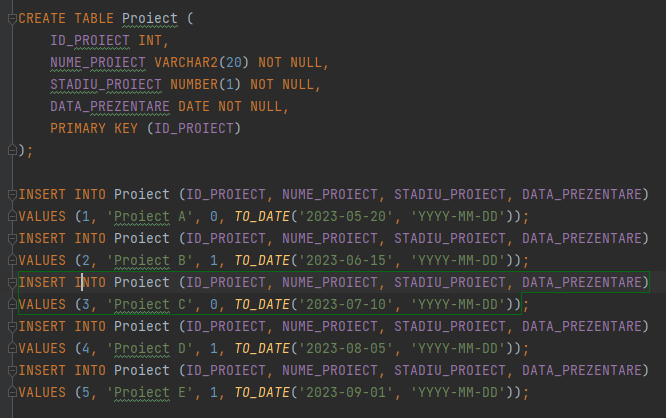


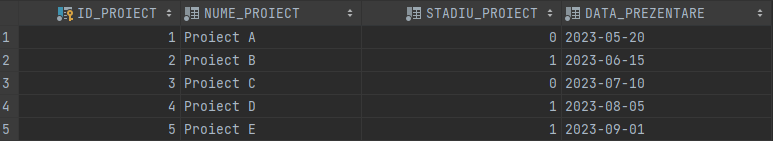
**ECHIPA**



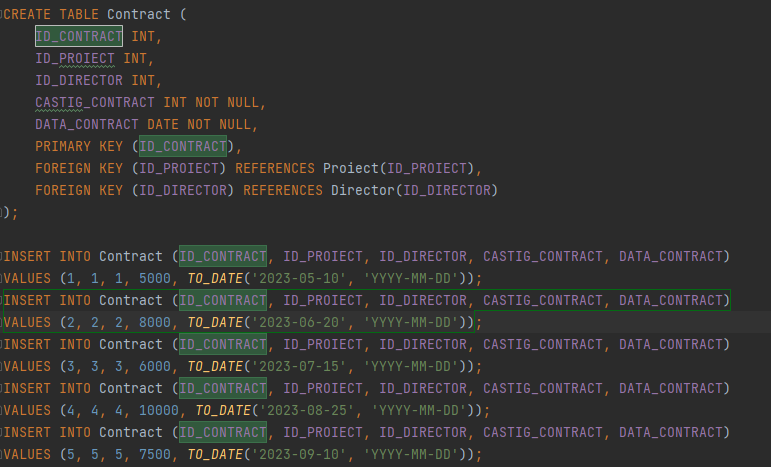


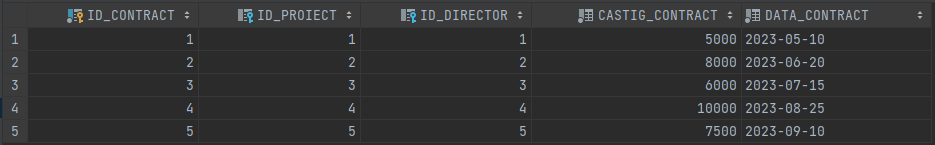
**PROIECT**



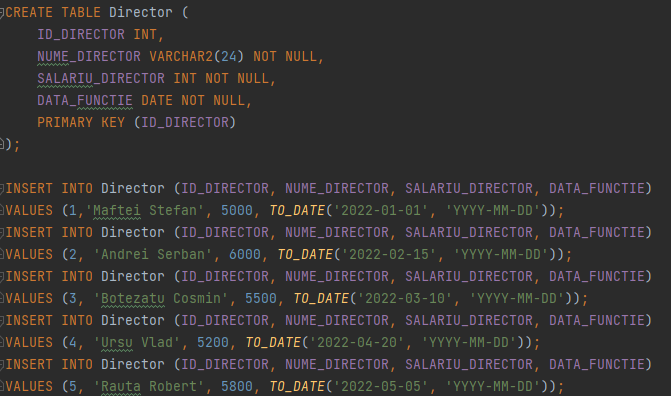


**CONTRACT**



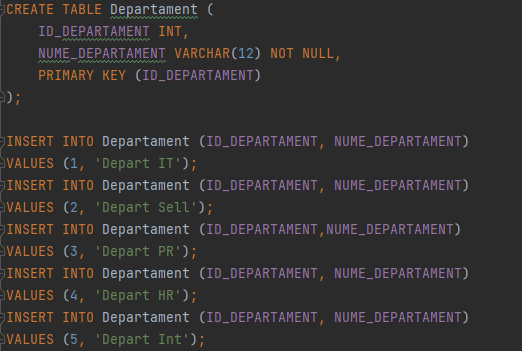


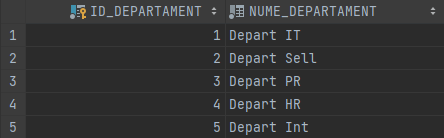
**DIRECTOR**



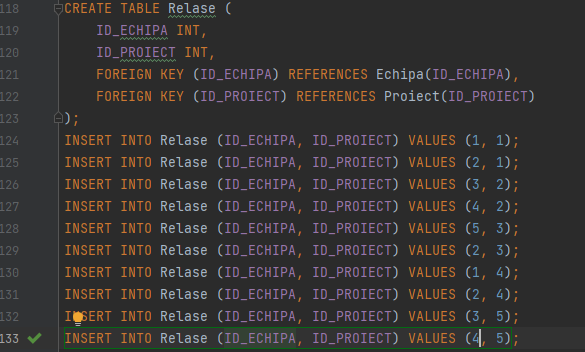


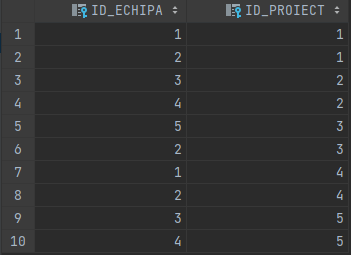
**DEPARTAMENT**





**RELASE**

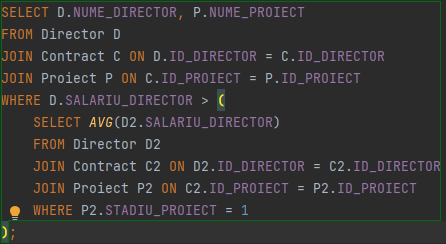


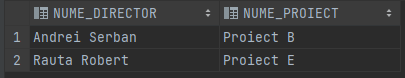


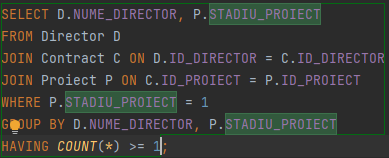
1. Formulați în limbaj natural și implementați 5 cereri sql complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

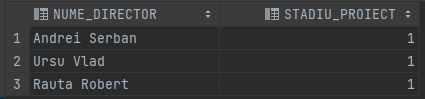
* Subcereri sincronizate în care intervin cel puțin 3 tabele.
* Subcereri nesincronizate în clauza FROM.
* Grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri (in cadrul aceleasi cereri).
* Ordonări și utilizarea funcțiilor NVL și DECODE (in cadrul aceleasi cereri)
* Utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE.
* Utilizarea a cel puțin 1 bloc de cerere (WITH).

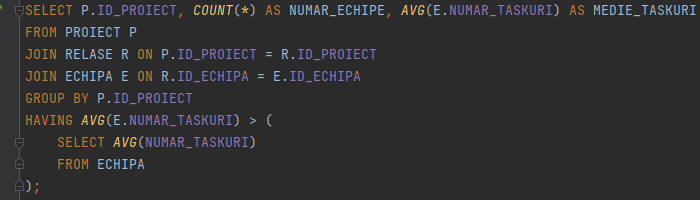
*EXEMPLE:*

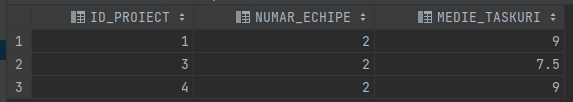


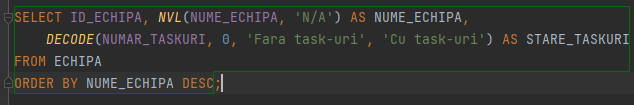


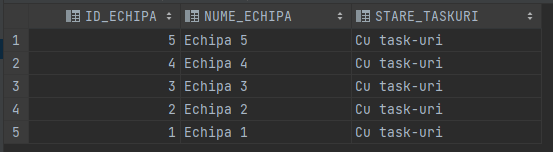


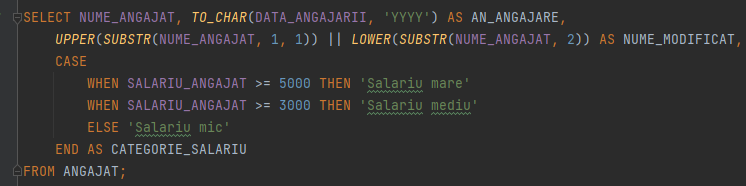


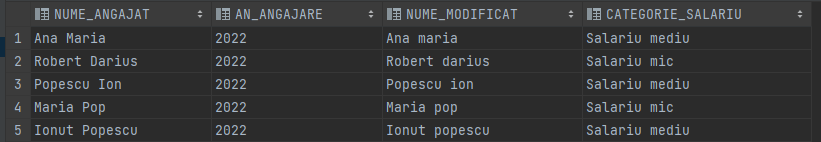


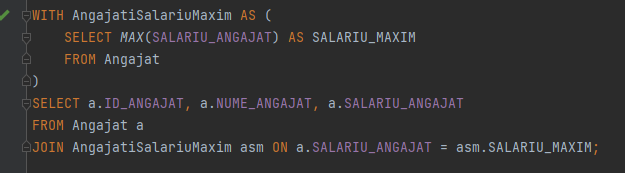






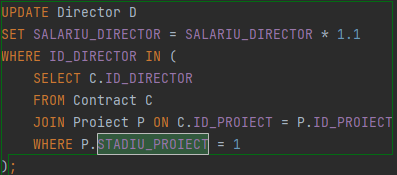




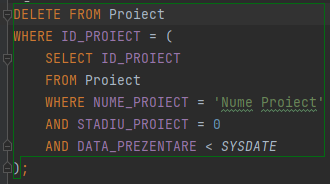




1. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.
2. Actualizare cu subcerere:



1. Ștergere cu subcerere:



1. Actualizare și ștergere cu subcereri combinate:

