**Nume Prenume**: Alexandru Norina-Ștefania

**Grupa**: 263

**Tema proiect**: Centru de meditații

Rezolvări capitolul III

a) Listare conținut cu posibilitatea de sortare (toate tabelele).

Opțiunile de sortare sunt prezente pe fiecare coloana din fiecare tabel, săgeata în jos specificând sortarea descrescătoare, iar săgeata în sus sortarea crescătoare. Tabelul pe care dorim să în vizualizăm se selectează din meniu, apăsându-se butonul de Submit pentru finalizarea alegerii.



Figura 1 – cerința a), tabelul JOBURI nesortat

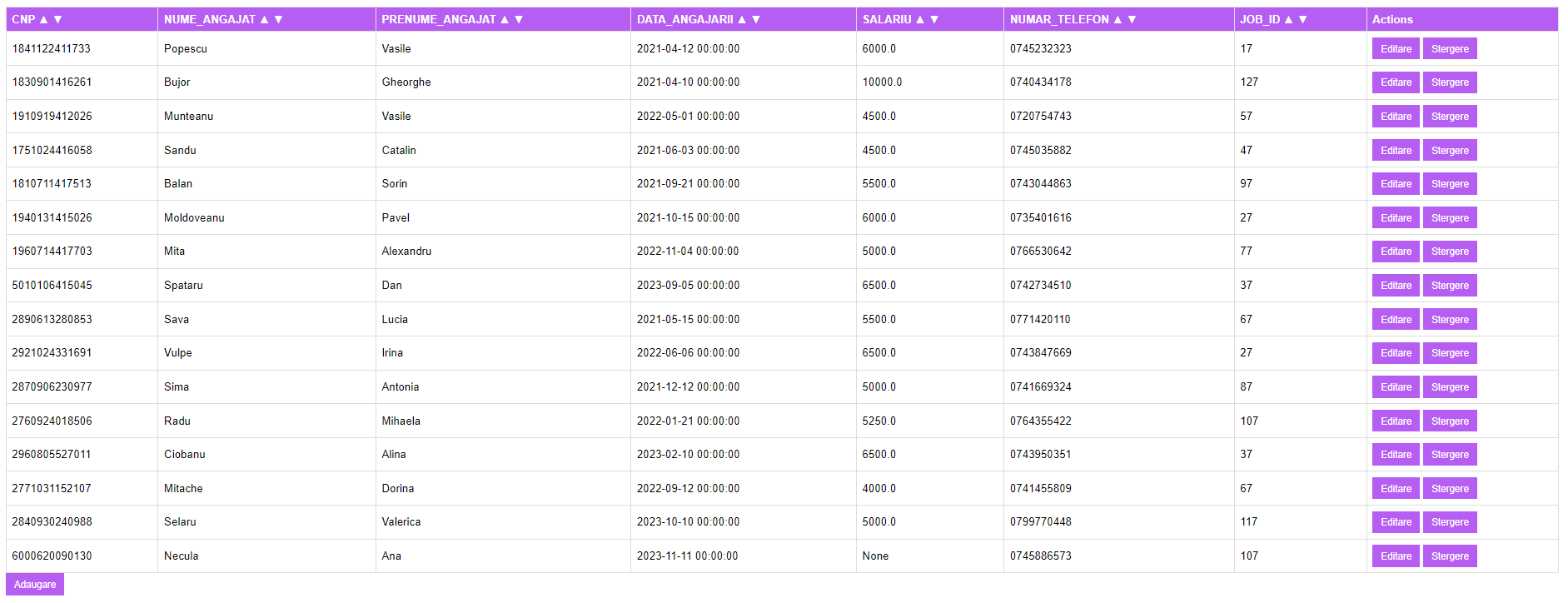


Figura 2 – cerința a), tabelul ANGAJATI nesortat

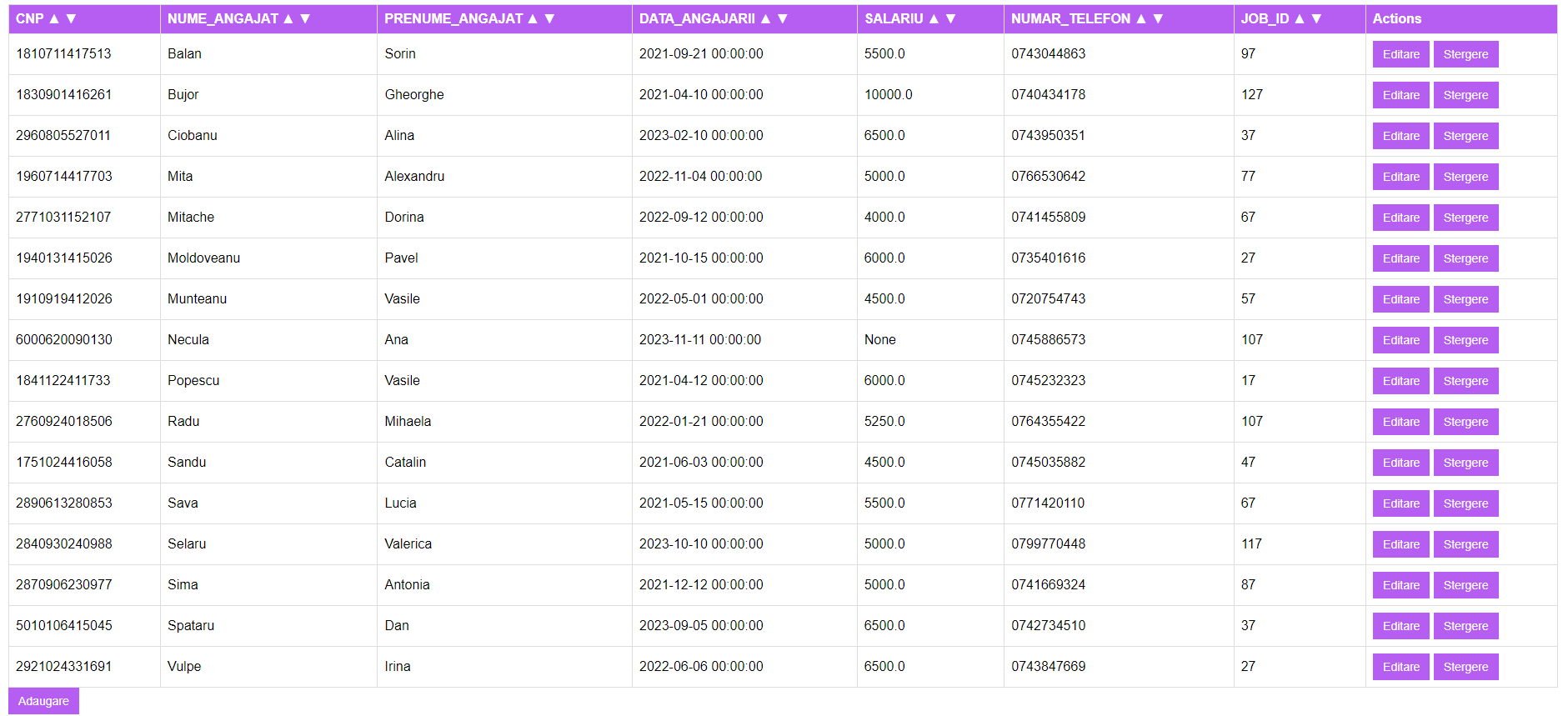


Figura 3 – cerința a), tabelul ANGAJATI sortat crescător după nume\_angajat

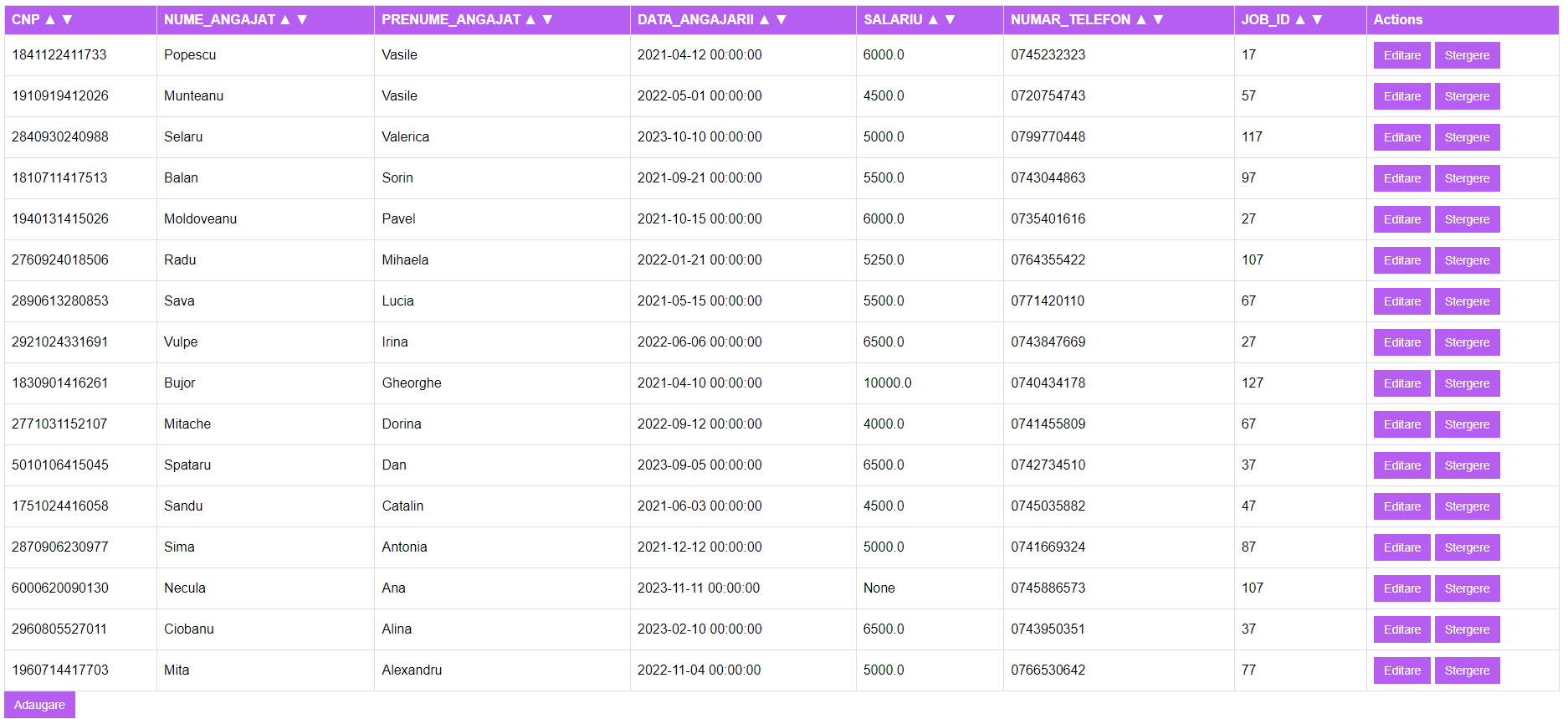


Figura 4 – cerința a), tabelul ANGAJATI, sortat descrescător după prenume\_angajat

b) Modificare informații (opțiune de edit/ștergere pentru fiecare înregistrare).

Pentru fiecare înregistrare din fiecare tabel avem opțiunea de editare și de ștergere. Accesând butonul de editare vom fi redirecționați către o nouă pagină unde putem edita câmpurile înregistrării. Opțiunea de ștergere, elimină înregistrarea și afișează tabelul modificat.

**Editare**

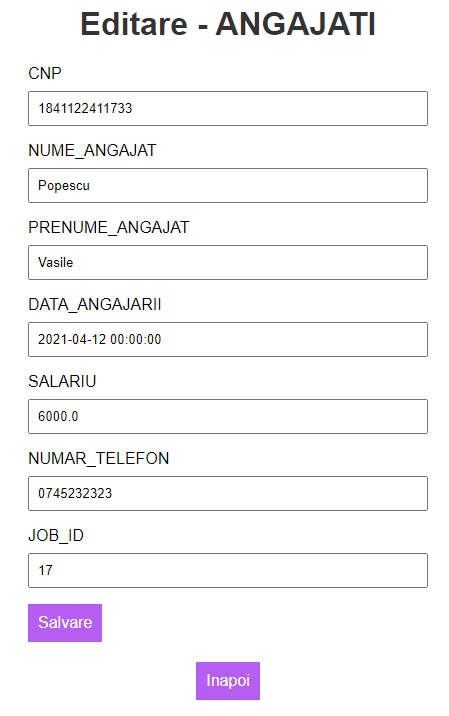
****

Figura 5 – editarea primei înregistrări din tabelul ANGAJATI

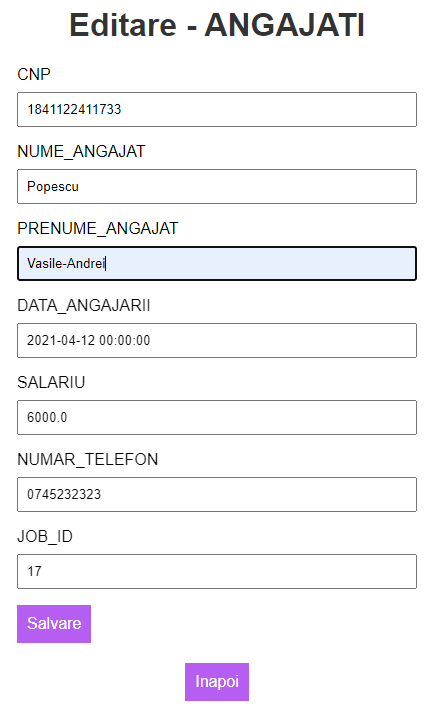


Figura 6 – modificarea numelui acestui ANGAJAT



Figura 7 – după apăsarea butonului Salvare ne redirecționează către tabelul modificat

**Ștergere**

****

Figura 7 – tabelul ANGAJATI înainte de ștergerea celui de-al doilea angajat

****

Figura 8 – tabelul ANGAJATI după ștergerea celui de-al doilea angajat

**Adăugare**

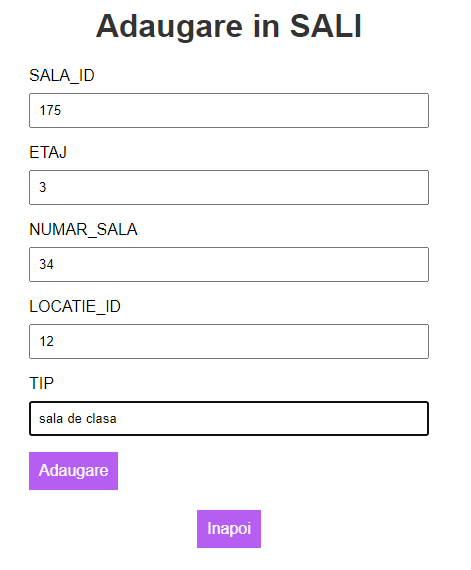


Figura 9 – Adăugarea unei înregistrării în tabelul SALI

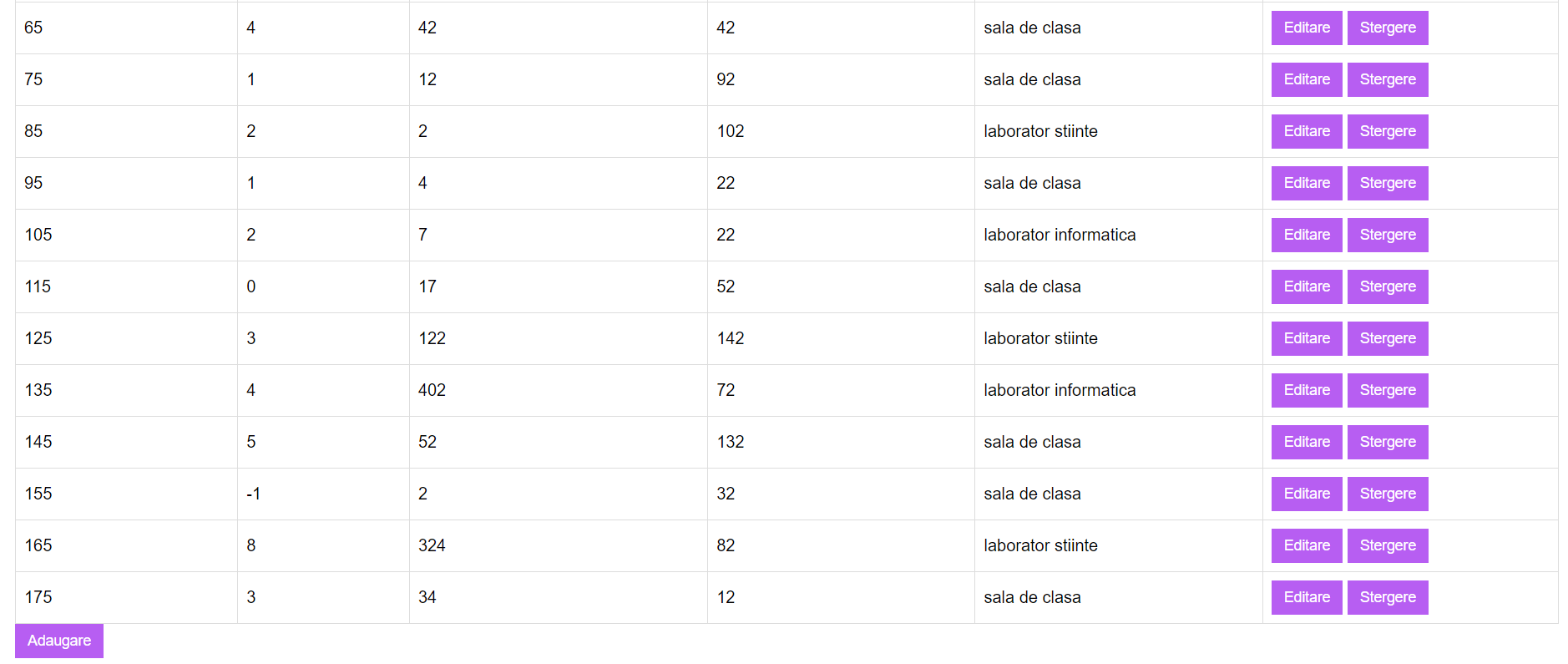


Figura 10 – tabelul SALI după adăugarea înregistrării

c) Afișarea rezultatului unei cereri care extrage informații din cel puțin 3 tabele şi le filtrează cu ajutorul a cel puțin 2 condiții.

**Enunț: Listați id-ul ședinței, numele profesorului și sala (adresă, etaj, numărul sălii) pentru toate ședințele desfășurate în București în anul 2023.**

select sedinta\_id,   
 nume\_angajat || ' ' || prenume\_angajat as Profesor,  
 'Str. ' || strada || ', nr. ' || numar\_locatie || ', et. ' || etaj || ', nr\_sala. ' || numar\_sala as Sala  
from sedinte join sali using (sala\_id)  
 join locatii using (locatie\_id)  
 join orase using (oras\_id)  
 join angajati on (CNP = profesor)  
where upper(denumire\_oras) = 'BUCURESTI'  
 and to\_char(data\_ora, 'yyyy') = '2023';



Figura 11 – rezolvarea cerinței C

d) Afișarea rezultatului unei cereri care folosește funcții grup și o clauză *having*

## Enunț: Listați elevii (nume și prenume) și suma totală pe care au achitat-o, pentru toți elevii care au de plătit mai mult de 200 lei.

select nume\_elev, prenume\_elev, sum(pret) as Suma  
from elevi join facturi on (elev = CNP)  
 join sedinte using (sedinta\_id)  
group by nume\_elev, prenume\_elev  
having sum(pret) > 200;



Figura 12 – rezolvarea cerinței D

e) Implementarea unei constrângeri de tipul *on delete cascade* şi exemplificare din interfață.

După cum este precizat și în documentația proiectului există implementate mai multe constrângeri de tipul *on delete cascade*. Un exemplu în acest caz ar fi următorul: dacă ștergem un oraș din baza de date (prin intermediul opțiunii de ștergere create la punctul b) se vor șterge toate locațiile din acel oraș și toate sălile care aparțineau acestor locații.

f) Utilizarea vizualizărilor (cel puțin 2 vizualizări: *compusă* care să permită operații LMD, respectiv *complexă*)

**Vizualizare compusă**

Pentru fiecare angajat care are jobul de profesor să se afișeze numele, prenumele, numărul de telefon și materia pe care acesta o predă.

create or replace view Profesori as  
select nume\_angajat, prenume\_angajat, materie\_predata, numar\_telefon  
from angajati join joburi using (job\_id)  
where lower(denumire\_job) = 'profesor';



Figura 13 – vizualizare compusă

**Vizualizare complexă**

Pentru fiecare profesor să se afișeze nume, prenumele, numărul de ședințe desfășurate, numărul de elevi pe care i-a meditat și suma totală pe care a încasat-o pentru aceste ședințe.

create or replace view Incasari\_profesori as  
select nume\_angajat, prenume\_angajat, count(distinct sedinta\_id) as Numar\_sedinte, count(factura\_id) as Numar\_elevi, sum(pret) as Suma\_incasata  
from angajati join sedinte on (CNP = profesor)  
 join facturi using (sedinta\_id)  
group by nume\_angajat, prenume\_angajat;

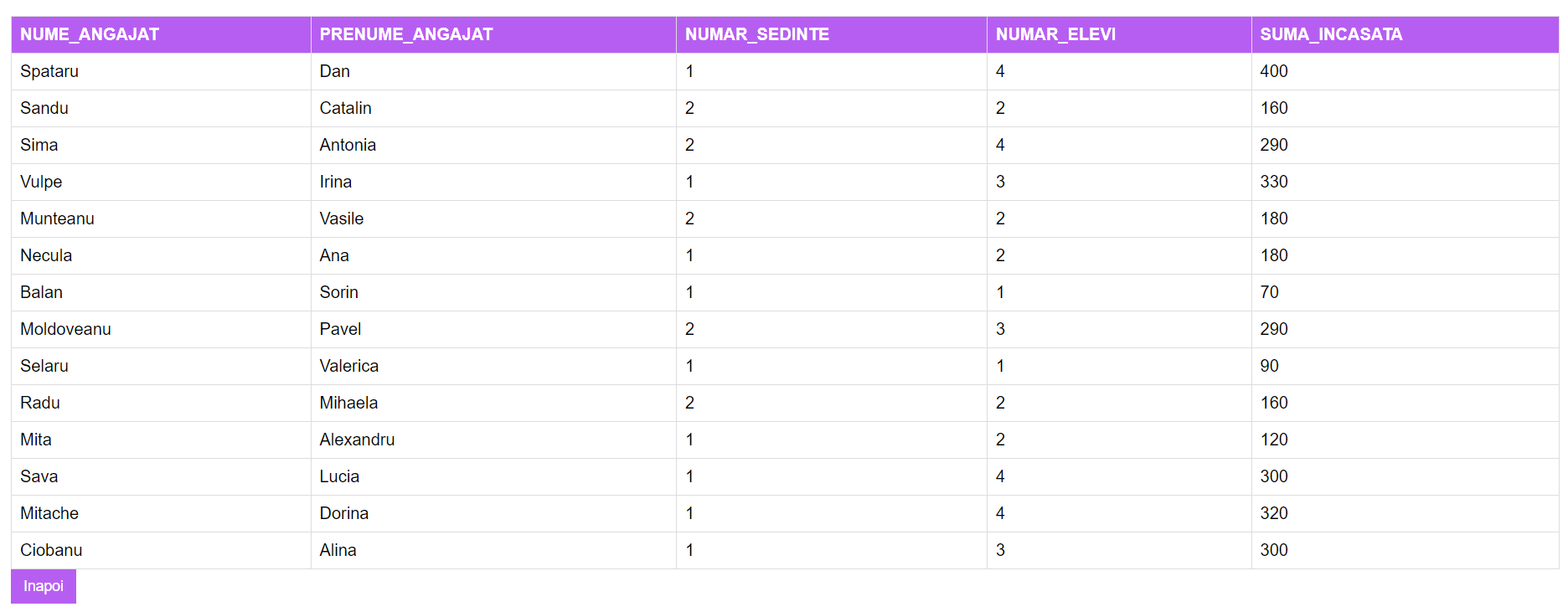


Figura 14 – vizualizare complexă