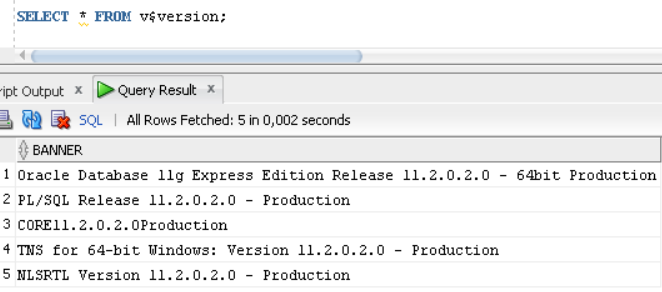
**Bejenaru Andrada-Elena Gr.344**

**Proiect Sisteme De Gestiune A Bazelor De Date 2024 - 2025**

**Administrarea Facultății de Matematică Informatică**

**Cuprins:**

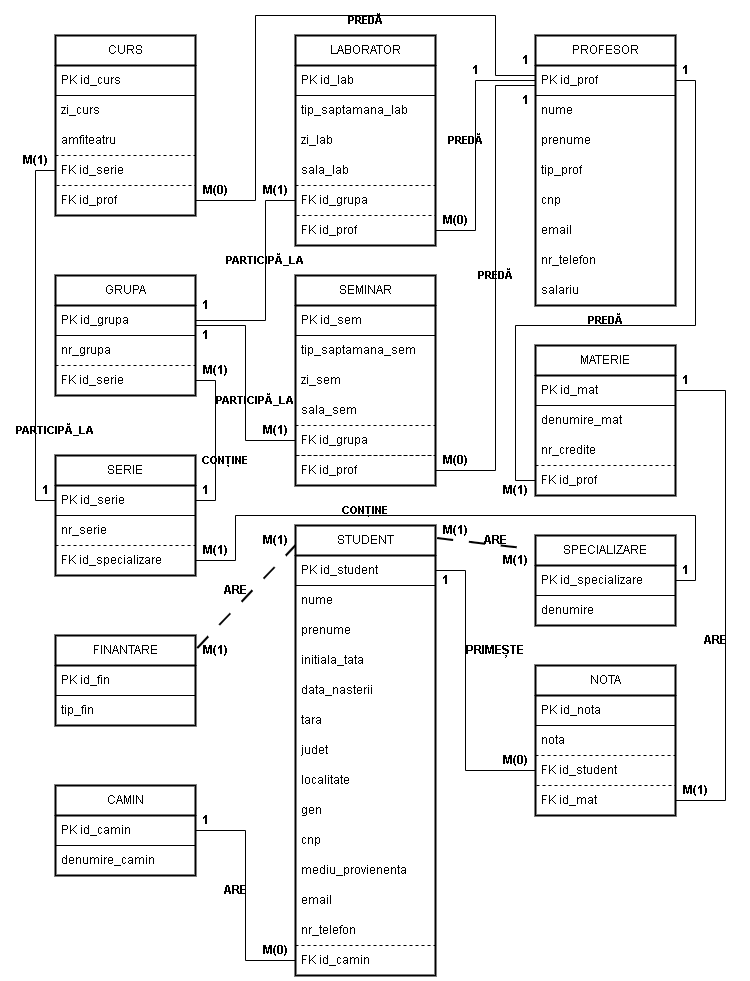
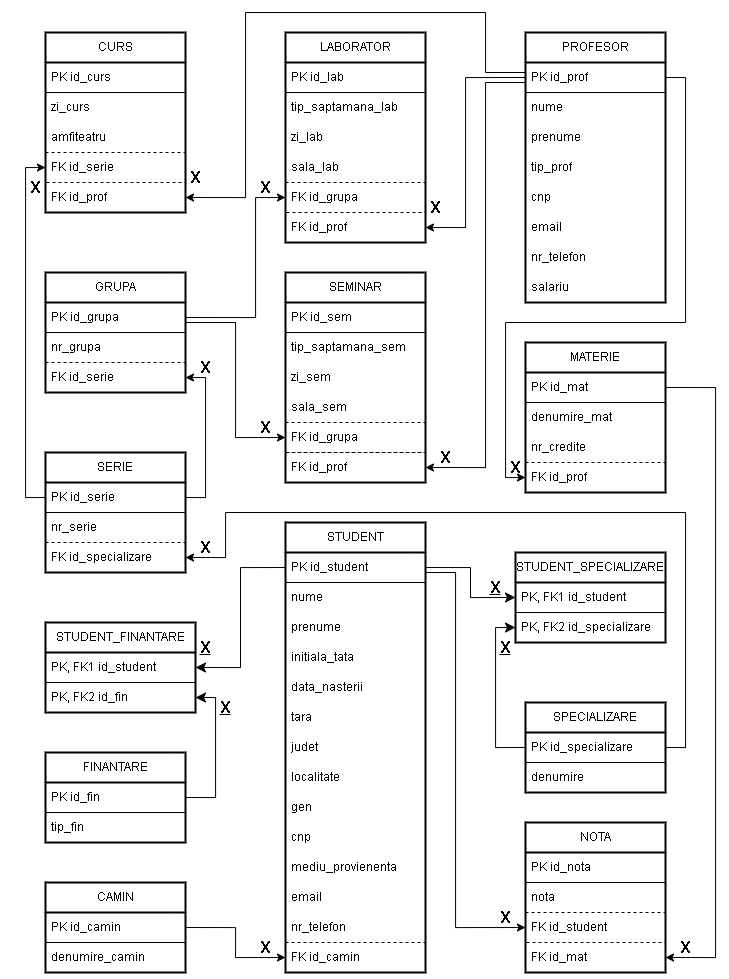
* [Cerința 1](#pb_1) – pag 3
* [Cerința 2](#pb_2) – pag 4
* [Cerința 3](#pb_3) – pag 5
* [Cerința 4](#pb_4) – pag 6
* [Cerința 5](#pb_5) – pag 9
* [Cerința 6](#pb_6) – pag 29
* [Cerința 7](#pb_7) – pag 32
* [Cerința 8](#pb_8) – pag 35
* [Cerința 9](#pb_9) – pag 40
* [Cerința 10](#pb_10) – pag 44
* [Cerința 11](#pb_11) – pag 46
* [Cerința 12](#pb_12) – pag 47



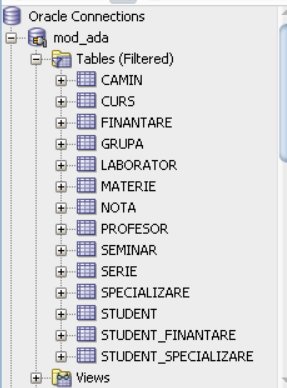
1. Prezentați pe scurt baza de date (utilitatea ei).

Proiectul constă într-o bază de date pentru Facultatea de Matematică și Informatică. Aceasta ține evidența studenților, profesorilor, repartizarea pe specializări, serii, grupe, organizarea cursurilor, seminarelor, cât și informații despre materie, cămin și forma de finanțare a studenților.

O astfel de bază de date permite facultății o gestiune mai bună a datelor și informaților despre studenți, profesori și a resurselor umane, lectori, laboranți, seminariști. Aceasta ar permite studenților, cât și profesorilor, urmărirea situației academice (note, tipul de finanțare), dar și informații despre orar. Un plus îl reprezintă și ușurarea procesului de repartizare buget/taxă, acordarea locurilor la cămin.

1. Realizați diagrama entitate-relație (ERD): entitățile, relațiile și atributele trebuie definite în limba română.
2. Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrând toate atributele necesare: entitățile, relațiile și atributele trebuie definite în limba română.
3. Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, adăugând toate constrângerile de integritate necesare (chei primare, cheile externe etc.).

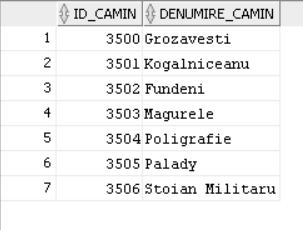
|  |
| --- |
| -- Crearea tabelelor  CREATE TABLE Student (  id\_student INT PRIMARY KEY,  nume VARCHAR2(50) NOT NULL,  prenume VARCHAR2(50) NOT NULL,  initiala\_tata VARCHAR2(5) NOT NULL,  data\_nasterii DATE,  tara VARCHAR2(50) NOT NULL,  judet VARCHAR2(50),  localitate VARCHAR2(50),  gen CHAR(1),  cnp VARCHAR2(50) NOT NULL,  mediu\_provenienta VARCHAR2(50) NOT NULL,  email VARCHAR2(100) NOT NULL,  nr\_telefon VARCHAR2(10) NOT NULL,  id\_camin INT,  FOREIGN KEY (id\_camin) REFERENCES Camin (id\_camin)  );  CREATE TABLE Profesor (  id\_prof INT PRIMARY KEY,  nume VARCHAR2(50) NOT NULL,  prenume VARCHAR2(50) NOT NULL,  tip\_prof VARCHAR2(50) NOT NULL,  cnp VARCHAR2(50) NOT NULL,  email VARCHAR2(50) NOT NULL,  nr\_telefon VARCHAR2(10) NOT NULL,  salariu INT NOT NULL  );  CREATE TABLE Materie (  id\_mat INT PRIMARY KEY,  denumire\_mat VARCHAR2(50) NOT NULL,  nr\_credite INT,  id\_prof INT NOT NULL,  FOREIGN KEY (id\_prof) REFERENCES Profesor (id\_prof)  );  CREATE TABLE Specializare (  id\_specializare INT PRIMARY KEY,  denumire VARCHAR2(50) NOT NULL  );  CREATE TABLE Serie (  id\_serie INT PRIMARY KEY,  nr\_serie INT,  id\_specializare INT NOT NULL,  FOREIGN KEY (id\_specializare) REFERENCES Specializare (id\_specializare)  );  CREATE TABLE Grupa (  id\_grupa INT PRIMARY KEY,  nr\_grupa INT,  id\_serie INT NOT NULL,  FOREIGN KEY (id\_serie) REFERENCES Serie (id\_serie)  );  CREATE TABLE Curs (  id\_curs INT PRIMARY KEY,  zi\_curs VARCHAR2(10),  amfiteatru VARCHAR2(50),  id\_serie INT NOT NULL,  id\_prof INT,  FOREIGN KEY (id\_serie) REFERENCES Serie (id\_serie),  FOREIGN KEY (id\_prof) REFERENCES Profesor (id\_prof)  );  CREATE TABLE Laborator (  id\_lab INT PRIMARY KEY,  tip\_saptamana\_lab VARCHAR2(10),  zi\_lab VARCHAR2(10),  sala\_lab VARCHAR2(20),  id\_grupa INT NOT NULL,  id\_prof INT,  FOREIGN KEY (id\_grupa) REFERENCES Grupa (id\_grupa),  FOREIGN KEY (id\_prof) REFERENCES Profesor (id\_prof)  );  CREATE TABLE Seminar (  id\_sem INT PRIMARY KEY,  tip\_saptamana\_sem VARCHAR2(10),  zi\_sem VARCHAR2(10),  sala\_sem VARCHAR2(20),  id\_grupa INT NOT NULL,  id\_prof INT,  FOREIGN KEY (id\_grupa) REFERENCES Grupa (id\_grupa),  FOREIGN KEY (id\_prof) REFERENCES Profesor (id\_prof)  );  CREATE TABLE Finantare (  id\_fin INT PRIMARY KEY,  tip\_fin VARCHAR2(50)  );  CREATE TABLE Student\_Specializare (  id\_student INT,  id\_specializare INT,  PRIMARY KEY (id\_student, id\_specializare),  FOREIGN KEY (id\_student) REFERENCES Student (id\_student),  FOREIGN KEY (id\_specializare) REFERENCES Specializare (id\_specializare)  );  CREATE TABLE Student\_Finantare (  id\_student INT,  id\_fin INT,  PRIMARY KEY (id\_student, id\_fin),  FOREIGN KEY (id\_student) REFERENCES Student (id\_student),  FOREIGN KEY (id\_fin) REFERENCES Finantare (id\_fin)  );  CREATE TABLE Camin (  id\_camin INT PRIMARY KEY,  denumire\_camin VARCHAR2(50)  );  CREATE TABLE Nota (  id\_nota INT PRIMARY KEY,  nota INT,  id\_student INT,  id\_mat INT,  FOREIGN KEY (id\_student) REFERENCES Student (id\_student),  FOREIGN KEY (id\_mat) REFERENCES Materie (id\_mat)  ); |

Tabelele au fost create:

1. Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru fiecare tabelă asociativă).

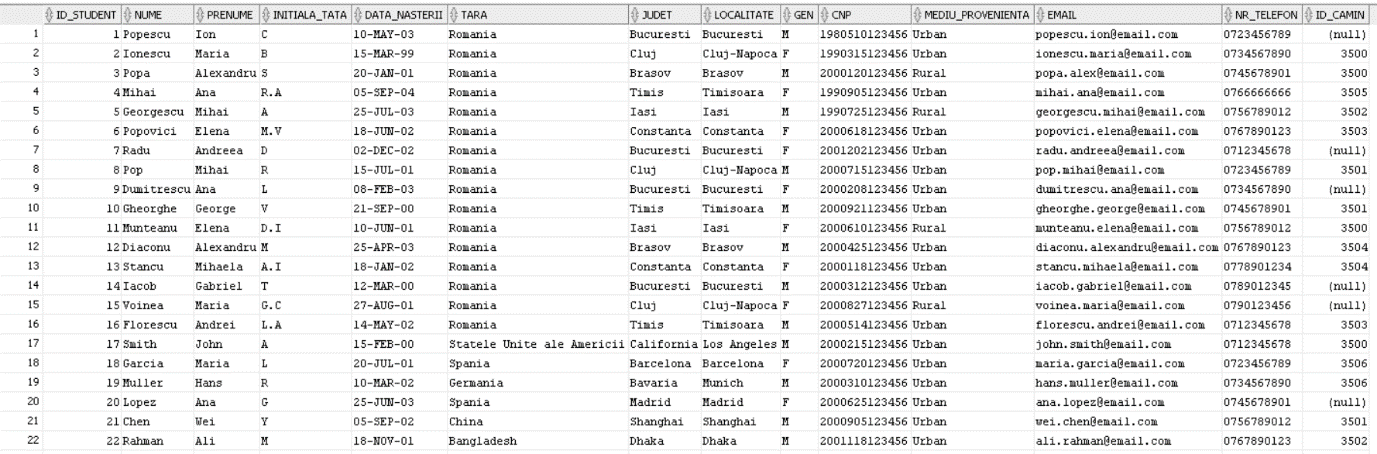
Inserarea datelor în tabela CAMIN:

|  |
| --- |
| INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Grozavesti');  INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Kogalniceanu');  INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Fundeni');  INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Magurele');  INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Poligrafie');  INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Palady');  INSERT INTO Camin (id\_camin, denumire\_camin)  VALUES (seq\_camin\_id.NEXTVAL, 'Stoian Militaru'); |

Date din tabela CAMIN:

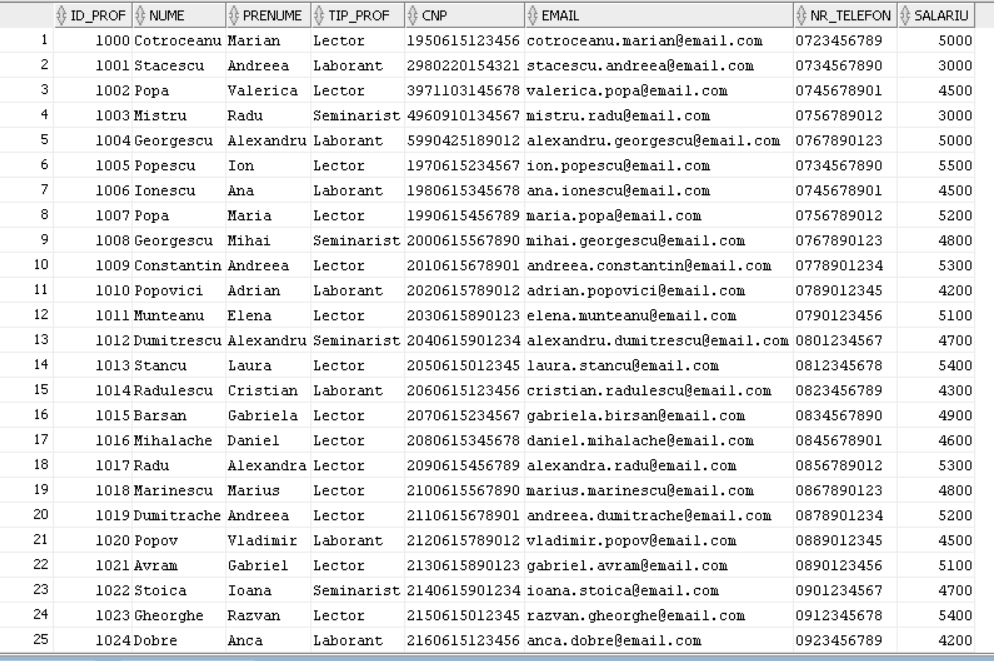
Inserarea datelor în tabela STUDENT:

|  |
| --- |
| INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL , 'Popescu', 'Ion', 'C', TO\_DATE('2003-05-10', 'YYYY-MM-DD'), 'Romania', 'Bucuresti', 'Bucuresti', 'M', '1980510123456', 'Urban', 'popescu.ion@email.com', '0723456789');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Ionescu', 'Maria', 'B', TO\_DATE('1999-03-15', 'YYYY-MM-DD'), 'Romania', 'Cluj', 'Cluj-Napoca', 'F', '1990315123456', 'Urban', 'ionescu.maria@email.com', '0734567890', 3500);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Popa', 'Alexandru', 'S', TO\_DATE('2001-01-20', 'YYYY-MM-DD'), 'Romania', 'Brasov', 'Brasov', 'M', '2000120123456', 'Rural', 'popa.alex@email.com', '0745678901', 3500);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Mihai', 'Ana', 'R.A', TO\_DATE('2004-09-05', 'YYYY-MM-DD'), 'Romania', 'Timis', 'Timisoara', 'F', '1990905123456', 'Urban', 'mihai.ana@email.com', '0766666666', 3505);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Georgescu', 'Mihai', 'A', TO\_DATE('2003-07-25', 'YYYY-MM-DD'), 'Romania', 'Iasi', 'Iasi', 'M', '1990725123456', 'Rural', 'georgescu.mihai@email.com', '0756789012', 3502);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Popovici', 'Elena', 'M.V', TO\_DATE('2002-06-18', 'YYYY-MM-DD'), 'Romania', 'Constanta', 'Constanta', 'F', '2000618123456', 'Urban', 'popovici.elena@email.com', '0767890123');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Radu', 'Andreea', 'D', TO\_DATE('2002-12-02', 'YYYY-MM-DD'), 'Romania', 'Bucuresti', 'Bucuresti', 'F', '2001202123456', 'Urban', 'radu.andreea@email.com', '0712345678');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Pop', 'Mihai', 'R', TO\_DATE('2001-07-15', 'YYYY-MM-DD'), 'Romania', 'Cluj', 'Cluj-Napoca', 'M', '2000715123456', 'Urban', 'pop.mihai@email.com', '0723456789', 3501);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Dumitrescu', 'Ana', 'L', TO\_DATE('2003-02-08', 'YYYY-MM-DD'), 'Romania', 'Bucuresti', 'Bucuresti', 'F', '2000208123456', 'Urban', 'dumitrescu.ana@email.com', '0734567890');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Gheorghe', 'George', 'V', TO\_DATE('2000-09-21', 'YYYY-MM-DD'), 'Romania', 'Timis', 'Timisoara', 'M', '2000921123456', 'Urban', 'gheorghe.george@email.com', '0745678901', 3501);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Munteanu', 'Elena', 'D.I', TO\_DATE('2001-06-10', 'YYYY-MM-DD'), 'Romania', 'Iasi', 'Iasi', 'F', '2000610123456', 'Rural', 'munteanu.elena@email.com', '0756789012', 3500);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Diaconu', 'Alexandru', 'M', TO\_DATE('2003-04-25', 'YYYY-MM-DD'), 'Romania', 'Brasov', 'Brasov', 'M', '2000425123456', 'Urban', 'diaconu.alexandru@email.com', '0767890123', 3504);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Stancu', 'Mihaela', 'A.I', TO\_DATE('2002-01-18', 'YYYY-MM-DD'), 'Romania', 'Constanta', 'Constanta', 'F', '2000118123456', 'Urban', 'stancu.mihaela@email.com', '0778901234', 3504);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Iacob', 'Gabriel', 'T', TO\_DATE('2000-03-12', 'YYYY-MM-DD'), 'Romania', 'Bucuresti', 'Bucuresti', 'M', '2000312123456', 'Urban', 'iacob.gabriel@email.com', '0789012345');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Voinea', 'Maria', 'G.C', TO\_DATE('2001-08-27', 'YYYY-MM-DD'), 'Romania', 'Cluj', 'Cluj-Napoca', 'F', '2000827123456', 'Rural', 'voinea.maria@email.com', '0790123456');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Florescu', 'Andrei', 'L.A', TO\_DATE('2002-05-14', 'YYYY-MM-DD'), 'Romania', 'Timis', 'Timisoara', 'M', '2000514123456', 'Urban', 'florescu.andrei@email.com', '0712345678', 3503);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Smith', 'John', 'A', TO\_DATE('2000-02-15', 'YYYY-MM-DD'), 'Statele Unite ale Americii', 'California', 'Los Angeles', 'M', '2000215123456', 'Urban', 'john.smith@email.com', '0712345678', 3500);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Garcia', 'Maria', 'L', TO\_DATE('2001-07-20', 'YYYY-MM-DD'), 'Spania', 'Barcelona', 'Barcelona', 'F', '2000720123456', 'Urban', 'maria.garcia@email.com', '0723456789', 3506);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Muller', 'Hans', 'R', TO\_DATE('2002-03-10', 'YYYY-MM-DD'), 'Germania', 'Bavaria', 'Munich', 'M', '2000310123456', 'Urban', 'hans.muller@email.com', '0734567890', 3506);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Lopez', 'Ana', 'G', TO\_DATE('2003-06-25', 'YYYY-MM-DD'), 'Spania', 'Madrid', 'Madrid', 'F', '2000625123456', 'Urban', 'ana.lopez@email.com', '0745678901');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Chen', 'Wei', 'Y', TO\_DATE('2002-09-05', 'YYYY-MM-DD'), 'China', 'Shanghai', 'Shanghai', 'M', '2000905123456', 'Urban', 'wei.chen@email.com', '0756789012', 3501);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Rahman', 'Ali', 'M', TO\_DATE('2001-11-18', 'YYYY-MM-DD'), 'Bangladesh', 'Dhaka', 'Dhaka', 'M', '2001118123456', 'Urban', 'ali.rahman@email.com', '0767890123', 3502);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Kim', 'Min-Ji', 'S', TO\_DATE('2000-04-14', 'YYYY-MM-DD'), 'Korea de Sud', 'Seoul', 'Seoul', 'F', '2000414123456', 'Urban', 'minji.kim@email.com', '0778901234');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Dubois', 'Pierre', 'M', TO\_DATE('2001-08-27', 'YYYY-MM-DD'), 'Franta', 'Paris', 'Paris', 'M', '2000827123456', 'Rural', 'pierre.dubois@email.com', '0789012345', 3502);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'González', 'Sofía', 'R', TO\_DATE('2000-03-12', 'YYYY-MM-DD'), 'Spania', 'Valencia', 'Valencia', 'F', '2000312123456', 'Urban', 'sofia.gonzalez@email.com', '0790123456');  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Silva', 'Isabella', 'M', TO\_DATE('2002-05-14', 'YYYY-MM-DD'), 'Brazilia', 'Sao Paulo', 'Sao Paulo', 'F', '2000514123456', 'Urban', 'isabella.silva@email.com', '0723456789', 3500);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon, id\_camin)  VALUES (seq\_student\_id.NEXTVAL, 'Negure', 'Silvia', 'M.V', TO\_DATE('1997-03-10', 'YYYY-MM-DD'), 'Romania', 'Bacau', 'Moinesti', 'F', '2000168323456', 'Urban', 'negure.silvia@email.com', '0767861423', 3506);  INSERT INTO Student (id\_student, nume, prenume, initiala\_tata, data\_nasterii, tara, judet, localitate, gen, cnp, mediu\_provenienta, email, nr\_telefon)  VALUES (seq\_student\_id.NEXTVAL, 'Potcoava', 'Valentin', 'C', TO\_DATE('1998-08-19', 'YYYY-MM-DD'), 'Romania', 'Iasi', 'Iasi', 'M', '7496618123456', 'Rural', 'potcoava.valentin@email.com', '0767899658'); |

Date din tabela STUDENT:

Inserarea datelor în tabela PROFESOR:

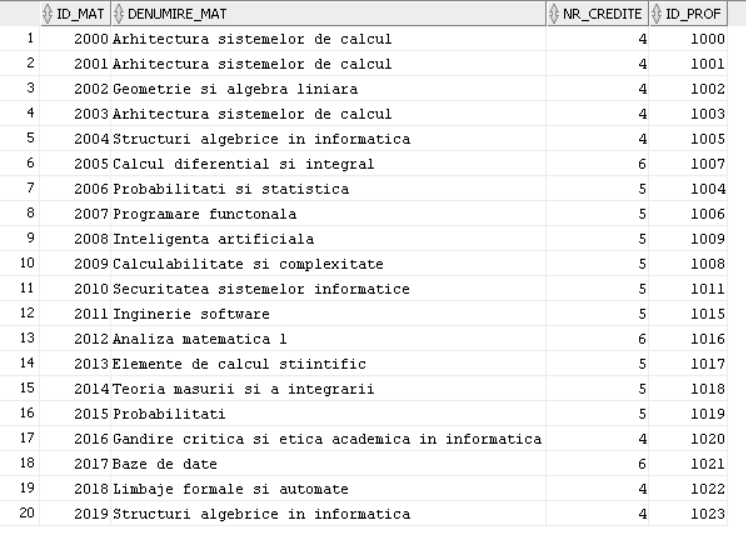
|  |
| --- |
| INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Cotroceanu', 'Marian', 'Lector', '1950615123456', 'cotroceanu.marian@email.com', '0723456789', 5000);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Stacescu', 'Andreea', 'Laborant', '2980220154321', 'stacescu.andreea@email.com', '0734567890', 3000);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Popa', 'Valerica', 'Lector', '3971103145678', 'valerica.popa@email.com', '0745678901', 4500);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Mistru', 'Radu', 'Seminarist', '4960910134567', 'mistru.radu@email.com', '0756789012', 3000);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Georgescu', 'Alexandru', 'Laborant', '5990425189012', 'alexandru.georgescu@email.com', '0767890123', 5000);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Popescu', 'Ion', 'Lector', '1970615234567', 'ion.popescu@email.com', '0734567890', 5500);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Ionescu', 'Ana', 'Laborant', '1980615345678', 'ana.ionescu@email.com', '0745678901', 4500);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Popa', 'Maria', 'Lector', '1990615456789', 'maria.popa@email.com', '0756789012', 5200);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Georgescu', 'Mihai', 'Seminarist', '2000615567890', 'mihai.georgescu@email.com', '0767890123', 4800);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Constantin', 'Andreea', 'Lector', '2010615678901', 'andreea.constantin@email.com', '0778901234', 5300);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Popovici', 'Adrian', 'Laborant', '2020615789012', 'adrian.popovici@email.com', '0789012345', 4200);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Munteanu', 'Elena', 'Lector', '2030615890123', 'elena.munteanu@email.com', '0790123456', 5100);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Dumitrescu', 'Alexandru', 'Seminarist', '2040615901234', 'alexandru.dumitrescu@email.com', '0801234567', 4700);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Stancu', 'Laura', 'Lector', '2050615012345', 'laura.stancu@email.com', '0812345678', 5400);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Radulescu', 'Cristian', 'Laborant', '2060615123456', 'cristian.radulescu@email.com', '0823456789', 4300);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Barsan', 'Gabriela', 'Lector', '2070615234567', 'gabriela.birsan@email.com', '0834567890', 4900);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Mihalache', 'Daniel', 'Lector', '2080615345678', 'daniel.mihalache@email.com', '0845678901', 4600);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Radu', 'Alexandra', 'Lector', '2090615456789', 'alexandra.radu@email.com', '0856789012', 5300);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Marinescu', 'Marius', 'Lector', '2100615567890', 'marius.marinescu@email.com', '0867890123', 4800);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Dumitrache', 'Andreea', 'Lector', '2110615678901', 'andreea.dumitrache@email.com', '0878901234', 5200);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Popov', 'Vladimir', 'Laborant', '2120615789012', 'vladimir.popov@email.com', '0889012345', 4500);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Avram', 'Gabriel', 'Lector', '2130615890123', 'gabriel.avram@email.com', '0890123456', 5100);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Stoica', 'Ioana', 'Seminarist', '2140615901234', 'ioana.stoica@email.com', '0901234567', 4700);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Gheorghe', 'Razvan', 'Lector', '2150615012345', 'razvan.gheorghe@email.com', '0912345678', 5400);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Dobre', 'Anca', 'Laborant', '2160615123456', 'anca.dobre@email.com', '0923456789', 4200);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Malaita', 'Crina', 'Laborant', '2180825123456', 'crina.malaita@email.com', '0856789012', 3600);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Garcea', 'Mirela', 'Laborant', '2171119123456', 'mirela.garcea@email.com', '0867890123', 3500);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Penciu', 'Iulian', 'Seminarist', '2160615123456', 'iulian.penciu@email.com', '0923456789', 4000);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Sincu', 'Daniela', 'Seminarist', '2190405123456', 'daniela.sincu@email.com', '0934567890', 4100);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Iancu', 'Laurentiu', 'Seminarist', '2171119123456', 'laurentiu.iancu@email.com', '0967890123', 4400);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Popovici', 'George', 'Seminarist', '2160615193456', 'george.popovici@email.com', '0923456752', 4100);  INSERT INTO Profesor (id\_prof, nume, prenume, tip\_prof, cnp, email, nr\_telefon, salariu)  VALUES (seq\_prof\_id.NEXTVAL, 'Cornaceanu', 'Valerica', 'Laborant', '2789615567890', 'valerica.cornaceanu@email.com', '0767894586', 5000); |

Date din tabela PROFESOR:

Inserarea datelor în tabela MATERIE:

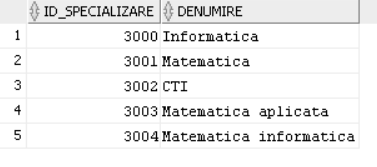
|  |
| --- |
| INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Arhitectura sistemelor de calcul', 4, 1000);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Arhitectura sistemelor de calcul', 4, 1001);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Geometrie si algebra liniara', 4, 1002);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Arhitectura sistemelor de calcul', 4, 1003);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Structuri algebrice in informatica', 4, 1005);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Calcul diferential si integral', 6, 1007);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Probabilitati si statistica', 5, 1004);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Programare functonala', 5, 1006);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Inteligenta artificiala', 5, 1009);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Calculabilitate si complexitate', 5, 1008);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Securitatea sistemelor informatice', 5, 1011);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Inginerie software', 5, 1015);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Analiza matematica 1', 6, 1016);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Elemente de calcul stiintific', 5, 1017);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Teoria masurii si a integrarii', 5, 1018);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Probabilitati', 5, 1019);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Gandire critica si etica academica in informatica', 4, 1020);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Baze de date', 6, 1021);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Limbaje formale si automate', 4, 1022);  INSERT INTO Materie (id\_mat, denumire\_mat, nr\_credite, id\_prof)  VALUES (seq\_mat\_id.NEXTVAL, 'Structuri algebrice in informatica', 4, 1023); |

Date din tabela MATERIE:



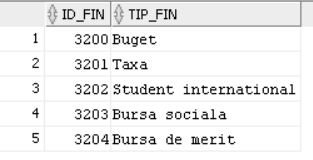
Inserarea datelor în tabela SPECIALIZARE:

|  |
| --- |
| INSERT INTO Specializare (id\_specializare, denumire)  VALUES (seq\_spec\_id.NEXTVAL, 'Informatica');  INSERT INTO Specializare (id\_specializare, denumire)  VALUES (seq\_spec\_id.NEXTVAL, 'Matematica');  INSERT INTO Specializare (id\_specializare, denumire)  VALUES (seq\_spec\_id.NEXTVAL, 'CTI');  INSERT INTO Specializare (id\_specializare, denumire)  VALUES (seq\_spec\_id.NEXTVAL, 'Matematica aplicata');  INSERT INTO Specializare (id\_specializare, denumire)  VALUES (seq\_spec\_id.NEXTVAL, 'Matematica informatica'); |

Date din tabela SPECIALIZARE:

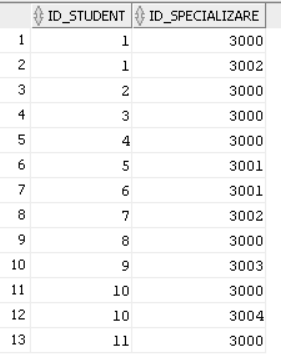
Inserarea datelor în tabela FINANTARE:

|  |
| --- |
| INSERT INTO Finantare (id\_fin, tip\_fin)  VALUES (seq\_fin\_id.NEXTVAL, 'Buget');  INSERT INTO Finantare (id\_fin, tip\_fin)  VALUES (seq\_fin\_id.NEXTVAL, 'Taxa');  INSERT INTO Finantare (id\_fin, tip\_fin)  VALUES (seq\_fin\_id.NEXTVAL, 'Student international');  INSERT INTO Finantare (id\_fin, tip\_fin)  VALUES (seq\_fin\_id.NEXTVAL, 'Bursa sociala');  INSERT INTO Finantare (id\_fin, tip\_fin)  VALUES (seq\_fin\_id.NEXTVAL, 'Bursa de merit'); |

Date din tabela FINANTARE:

Inserarea datelor în tabela STUDENT\_SPECIALIZARE:

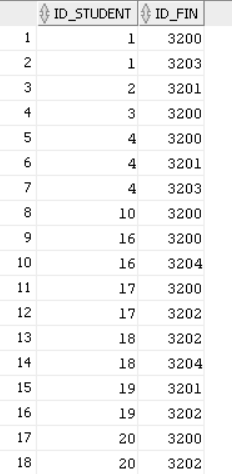
|  |
| --- |
| INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (1, 3000);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (2, 3000);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (1, 3002);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (3, 3000);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (4, 3000);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (5, 3001);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (6, 3001);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (7,3002);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (8, 3000);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (9, 3003);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (10, 3004);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (11, 3000);  INSERT INTO Student\_Specializare (id\_student, id\_specializare)  VALUES (10, 3000); |

Date din tabela STUDENT\_SPECIALIZARE:

Inserarea datelor în tabela STUDENT\_FINANTARE:

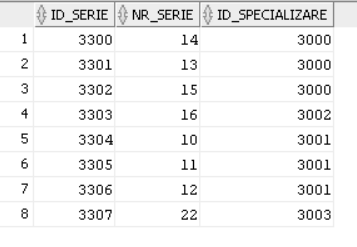
|  |
| --- |
| INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (1, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (10, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (2, 3201);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (3, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (4, 3201);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (1, 3203);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (16, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (17, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (18, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (19, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (20, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (21, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (22, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (23, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (24, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (25, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (26, 3202);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (16, 3204);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (17, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (18, 3204);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (19, 3201);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (20, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (21, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (22, 3204);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (23, 3204);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (24, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (25, 3201);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (26, 3201);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (4, 3200);  INSERT INTO Student\_Finantare (id\_student, id\_fin)  VALUES (4, 3203); |

Date din tabela STUDENT\_FINANTARE:



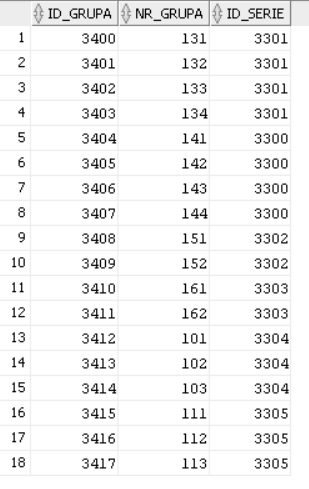
Inserarea datelor în tabela SERIE:

|  |
| --- |
| INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 14, 3000);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 13, 3000);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 15, 3000);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 16, 3002);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 10, 3001);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 11, 3001);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 12, 3001);  INSERT INTO Serie (id\_serie, nr\_serie, id\_specializare)  VALUES (seq\_serie\_id.NEXTVAL, 22, 3003); |

Date din tabela SERIE:

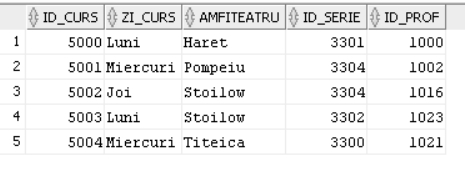
Inserarea datelor în tabela GRUPA:

|  |
| --- |
| INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 131, 3301);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 132, 3301);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 133, 3301);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 134, 3301);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 141, 3300);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 142, 3300);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 143, 3300);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 144, 3300);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 151, 3302);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 152, 3302);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 161, 3303);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 162, 3303);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 101, 3304);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 102, 3304);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 103, 3304);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 111, 3305);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 112, 3305);  INSERT INTO Grupa (id\_grupa, nr\_grupa, id\_serie)  VALUES (seq\_grupa\_id.NEXTVAL, 113, 3305); |

Date din tabela GRUPA:

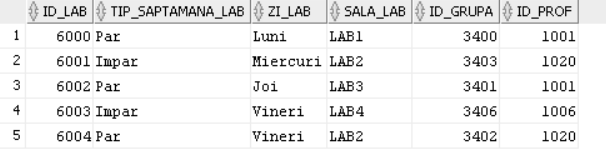
Inserarea datelor în tabela CURS:

|  |
| --- |
| INSERT INTO Curs (id\_curs, zi\_curs, amfiteatru, id\_serie, id\_prof)  VALUES (seq\_curs\_id.NEXTVAL, 'Luni', 'Haret', 3301, 1000);  INSERT INTO Curs (id\_curs, zi\_curs, amfiteatru, id\_serie, id\_prof)  VALUES (seq\_curs\_id.NEXTVAL, 'Miercuri', 'Pompeiu', 3304, 1002);  INSERT INTO Curs (id\_curs, zi\_curs, amfiteatru, id\_serie, id\_prof)  VALUES (seq\_curs\_id.NEXTVAL, 'Joi', 'Stoilow', 3304, 1016);  INSERT INTO Curs (id\_curs, zi\_curs, amfiteatru, id\_serie, id\_prof)  VALUES (seq\_curs\_id.NEXTVAL, 'Luni', 'Stoilow', 3302, 1023);  INSERT INTO Curs (id\_curs, zi\_curs, amfiteatru, id\_serie, id\_prof)  VALUES (seq\_curs\_id.NEXTVAL, 'Miercuri', 'Titeica', 3300, 1021); |

Date din tabela CURS:

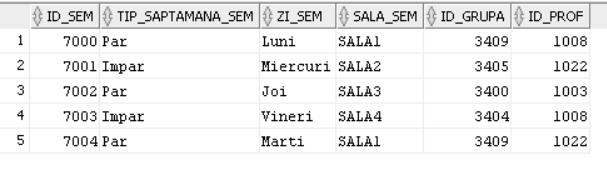
Inserarea datelor în tabela LABORATOR:

|  |
| --- |
| INSERT INTO Laborator (id\_lab, tip\_saptamana\_lab, zi\_lab, sala\_lab, id\_grupa, id\_prof)  VALUES (seq\_lab\_id.NEXTVAL, 'Par', 'Luni', 'LAB1', 3400, 1001);  INSERT INTO Laborator (id\_lab, tip\_saptamana\_lab, zi\_lab, sala\_lab, id\_grupa, id\_prof)  VALUES (seq\_lab\_id.NEXTVAL, 'Impar', 'Miercuri', 'LAB2', 3403, 1020);  INSERT INTO Laborator (id\_lab, tip\_saptamana\_lab, zi\_lab, sala\_lab, id\_grupa, id\_prof)  VALUES (seq\_lab\_id.NEXTVAL, 'Par', 'Joi', 'LAB3', 3401, 1001);  INSERT INTO Laborator (id\_lab, tip\_saptamana\_lab, zi\_lab, sala\_lab, id\_grupa, id\_prof)  VALUES (seq\_lab\_id.NEXTVAL, 'Impar', 'Vineri', 'LAB4', 3406, 1006);  INSERT INTO Laborator (id\_lab, tip\_saptamana\_lab, zi\_lab, sala\_lab, id\_grupa, id\_prof)  VALUES (seq\_lab\_id.NEXTVAL, 'Par', 'Vineri', 'LAB2', 3402, 1020); |

Date din tabela LABORATOR:

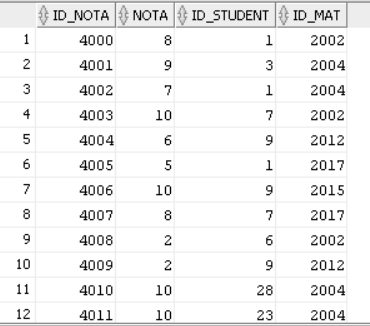
Inserarea datelor în tabela SEMINAR:

|  |
| --- |
| INSERT INTO Seminar (id\_sem, tip\_saptamana\_sem, zi\_sem, sala\_sem, id\_grupa, id\_prof)  VALUES (seq\_sem\_id.NEXTVAL, 'Par', 'Luni', 'SALA1', 3409, 1008);  INSERT INTO Seminar (id\_sem, tip\_saptamana\_sem, zi\_sem, sala\_sem, id\_grupa, id\_prof)  VALUES (seq\_sem\_id.NEXTVAL, 'Impar', 'Miercuri', 'SALA2', 3405, 1022);  INSERT INTO Seminar (id\_sem, tip\_saptamana\_sem, zi\_sem, sala\_sem, id\_grupa, id\_prof)  VALUES (seq\_sem\_id.NEXTVAL, 'Par', 'Joi', 'SALA3', 3400, 1003);  INSERT INTO Seminar (id\_sem, tip\_saptamana\_sem, zi\_sem, sala\_sem, id\_grupa, id\_prof)  VALUES (seq\_sem\_id.NEXTVAL, 'Impar', 'Vineri', 'SALA4', 3404, 1008);  INSERT INTO Seminar (id\_sem, tip\_saptamana\_sem, zi\_sem, sala\_sem, id\_grupa, id\_prof)  VALUES (seq\_sem\_id.NEXTVAL, 'Par', 'Marti', 'SALA1', 3409, 1022); |

Date din tabela SEMINAR:

Inserarea datelor în tabela NOTA:

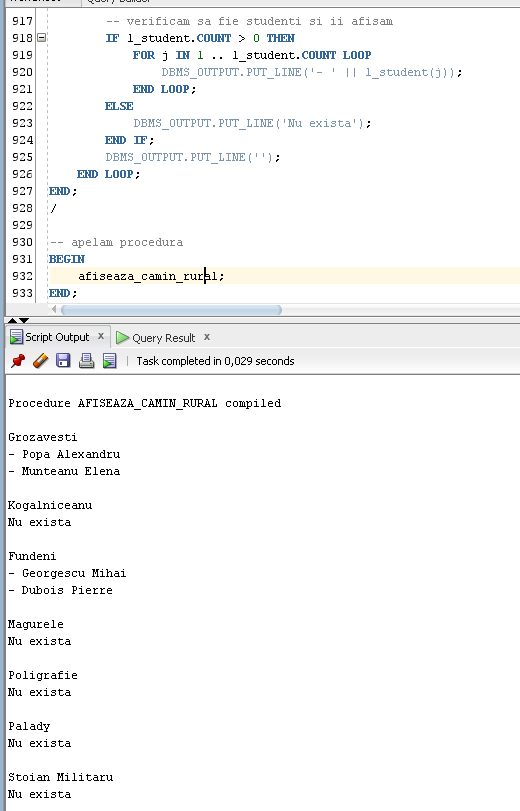
|  |
| --- |
| INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 8, 1, 2002);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 9, 3, 2004);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 7, 1, 2004);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 10, 7, 2002);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 6, 9, 2012);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 5, 1, 2017);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 10, 9, 2015);  INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 8, 7, 2017); |

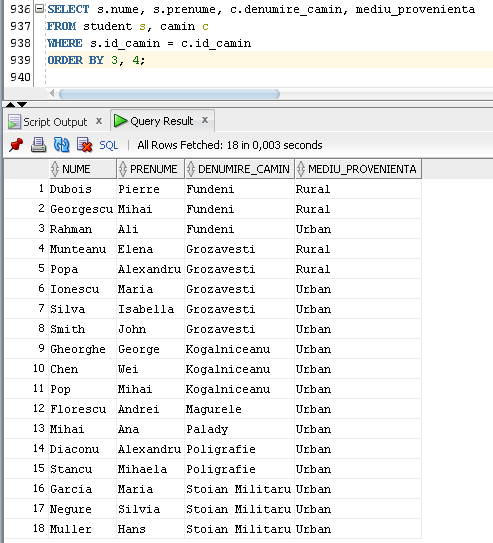
Date din tabela NOTA:

1. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze toate cele 3 tipuri de colecții studiate. Apelați subprogramul.

**CERINȚA:** Să se afișeze pentru fiecare cămin toți studenții din mediul rural.

|  |
| --- |
| CREATE OR REPLACE PROCEDURE afiseaza\_camin\_rural IS  -- definim un tablou indexat pt id\_camin  TYPE t\_id\_camin IS TABLE OF NUMBER INDEX BY PLS\_INTEGER;  l\_id\_camin t\_id\_camin;    -- definim un vector pt denumirile caminelor  TYPE t\_camin IS VARRAY(100) OF VARCHAR2(100);  l\_camin t\_camin := t\_camin();    -- definim un tablou imbricat pt nume studenti  TYPE t\_student IS TABLE OF VARCHAR2(200);  l\_student t\_student;    BEGIN  -- salvam id urile de camin in tabloul indexat  SELECT id\_camin  BULK COLLECT INTO l\_id\_camin  FROM camin;    -- salvam nume camine in vector  SELECT denumire\_camin  BULK COLLECT INTO l\_camin  FROM camin;    -- pt fiecare camin cautam studentii din rural  FOR i IN 1 .. l\_id\_camin.COUNT LOOP  SELECT s.nume || ' ' || s.prenume  BULK COLLECT INTO l\_student  FROM student s  WHERE s.id\_camin = l\_id\_camin(i)  AND LOWER(s.mediu\_provenienta) = 'rural';    -- afisam nume camin  DBMS\_OUTPUT.PUT\_LINE(l\_camin(i));    -- verificam sa fie studenti si ii afisam  IF l\_student.COUNT > 0 THEN  FOR j IN 1 .. l\_student.COUNT LOOP  DBMS\_OUTPUT.PUT\_LINE('- ' || l\_student(j));  END LOOP;  ELSE  DBMS\_OUTPUT.PUT\_LINE('Nu exista');  END IF;  DBMS\_OUTPUT.PUT\_LINE('');  END LOOP;  END;  /  -- apelam procedura  BEGIN  afiseaza\_camin\_rural;  END;  / |

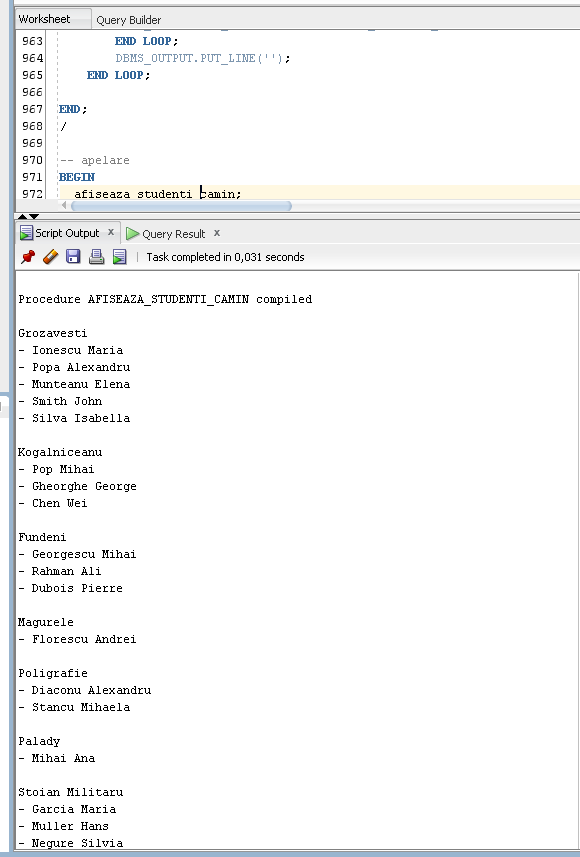


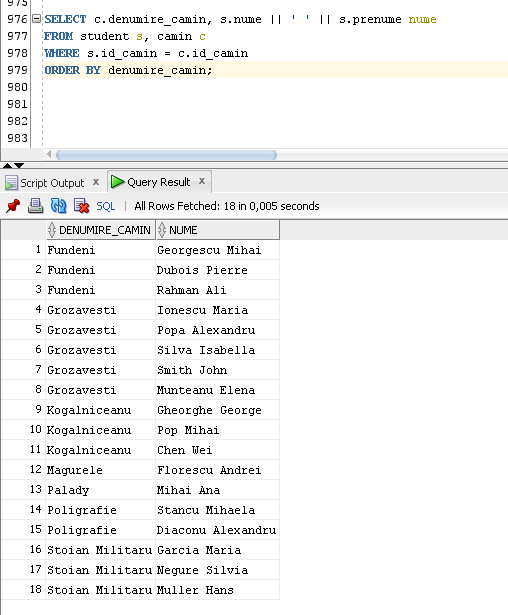
Verificare:

1. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat. Apelați subprogramul.

**CERINȚA:** Să se afișeze pentru fiecare cămin numele studenților.

|  |
| --- |
| CREATE OR REPLACE PROCEDURE afiseaza\_studenti\_camin IS  -- definim un cursor explicit pt a obtine caminele  CURSOR c\_camin IS  SELECT id\_camin, denumire\_camin  FROM camin;    -- definim un cursor parametrizat pt a selecta studentii in functie de id camin  CURSOR c\_student (p\_id\_camin camin.id\_camin%TYPE) IS  SELECT nume || ' ' || prenume AS nume\_complet  FROM student  WHERE id\_camin = p\_id\_camin;  BEGIN  -- obtinem fiecare camin  FOR camin\_rec IN c\_camin LOOP  DBMS\_OUTPUT.PUT\_LINE(camin\_rec.denumire\_camin);    -- obtinem studentii din caminului curent  FOR student\_rec IN c\_student(camin\_rec.id\_camin) LOOP  DBMS\_OUTPUT.PUT\_LINE('- ' || student\_rec.nume\_complet);  END LOOP;  DBMS\_OUTPUT.PUT\_LINE('');  END LOOP;  END;  /  -- apelare  BEGIN  afiseaza\_studenti\_camin;  END;  / |

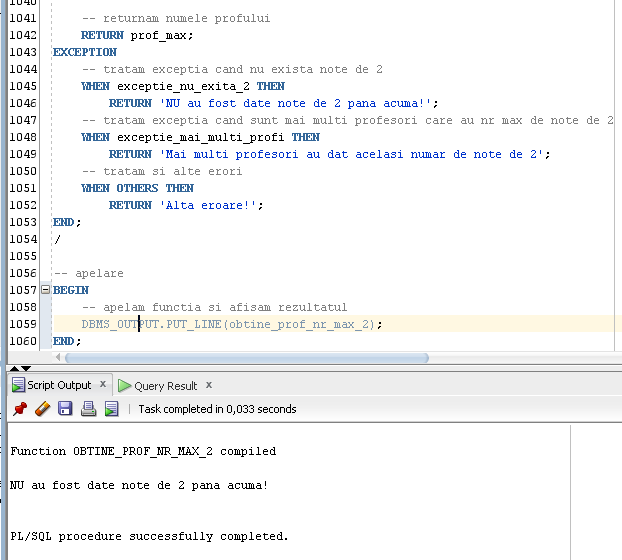
Output:

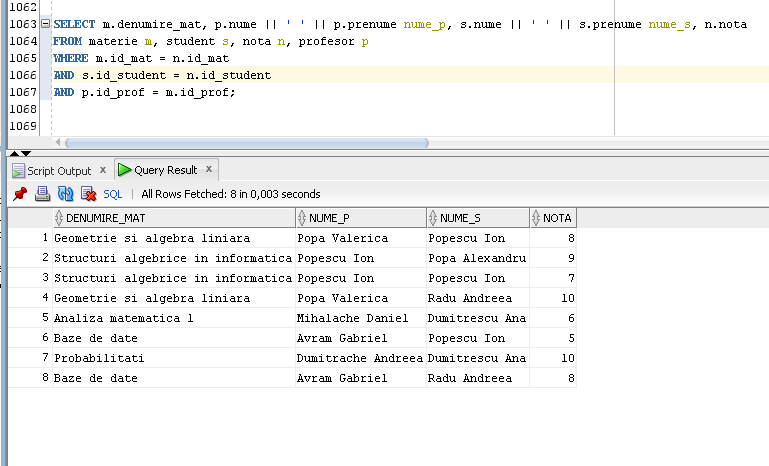
Verificare:

1. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele create. Tratați toate excepțiile care pot apărea, incluzând excepțiile predefinite NO\_DATA\_FOUND și TOO\_MANY\_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

**CERINȚA:** Să se afișeze profesorul care a dat cele mai multe note de 2.

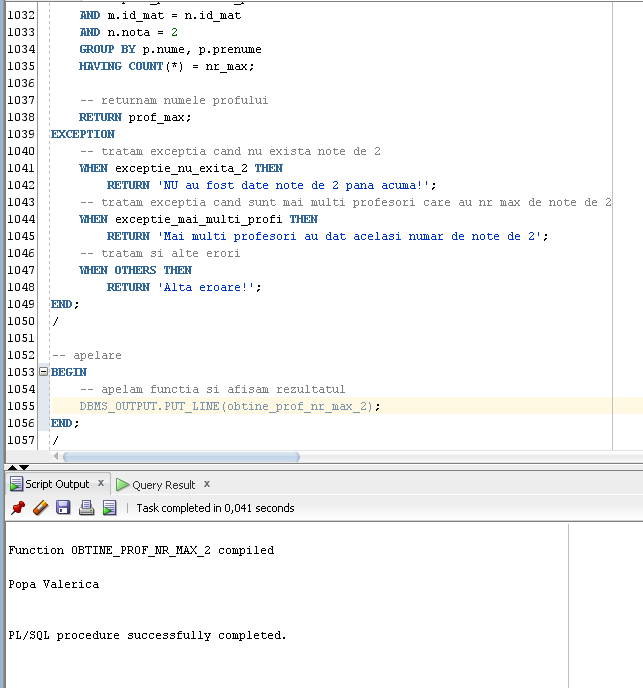
|  |
| --- |
| CREATE OR REPLACE FUNCTION obtine\_prof\_nr\_max\_2 RETURN VARCHAR2 IS  prof\_max VARCHAR2(50); -- variabila pt numele profului care a dat cele mai multe note de 2  nr\_max INTEGER := 0; -- variabila pt nr max de note de 2  nr\_profi INTEGER := 0; -- variabila pt nr de profi care au dat nr max de note de 2  BEGIN  -- gasim nr max de note de 2 date de profi  -- group by face ca fiecare prof sa fie considerat un grupulet separat  -- deci obtinem nr total de 2 uri pt fiecare prof  SELECT MAX(nr\_note)  INTO nr\_max  FROM (  SELECT COUNT(\*) nr\_note  FROM profesor p, materie m, nota n  WHERE p.id\_prof = m.id\_prof  AND m.id\_mat = n.id\_mat  AND n.nota = 2  GROUP BY p.nume, p.prenume  );    -- verificam daca nu exista note de 2  IF nr\_max IS NULL THEN  RAISE NO\_DATA\_FOUND;  END IF;    -- selectam profii care au dat acest nr max de note de 2  -- group by e ca mai sus si adaugam si  -- having pt a selecta doar profii care au nr max de note de 2 gasit anterior  SELECT COUNT(\*)  INTO nr\_profi  FROM (  SELECT p.nume, p.prenume  FROM profesor p, materie m, nota n  WHERE p.id\_prof = m.id\_prof  AND m.id\_mat = n.id\_mat  AND n.nota = 2  GROUP BY p.nume, p.prenume  HAVING COUNT(\*) = nr\_max  );    -- verificam daca sunt mai multi profesori care au acel nr max de 2 date  IF nr\_profi > 1 THEN  RAISE TOO\_MANY\_ROWS;  END IF;    -- selectam numele profului care a dat cele mai multe note 2  SELECT p.nume || ' ' || p.prenume  INTO prof\_max  FROM profesor p, materie m, nota n  WHERE p.id\_prof = m.id\_prof  AND m.id\_mat = n.id\_mat  AND n.nota = 2  GROUP BY p.nume, p.prenume  HAVING COUNT(\*) = nr\_max;    -- returnam numele profului  RETURN prof\_max;  EXCEPTION  -- tratam exceptia cand nu exista note de 2  WHEN NO\_DATA\_FOUND THEN  RETURN 'NU au fost date note de 2 pana acuma!';  -- tratam exceptia cand sunt mai multi profesori care au nr max de note de 2  WHEN TOO\_MANY\_ROWS THEN  RETURN 'Mai multi profesori au dat acelasi numar de note de 2';  -- tratam si alte erori  WHEN OTHERS THEN  RETURN 'Alta eroare!';  END;  /  -- apelare  BEGIN  -- apelam functia si afisam rezultatul  DBMS\_OUTPUT.PUT\_LINE(obtine\_prof\_nr\_max\_2);  END;  / |

Testăm excepția în care nu există profesori care au dat note de 2:

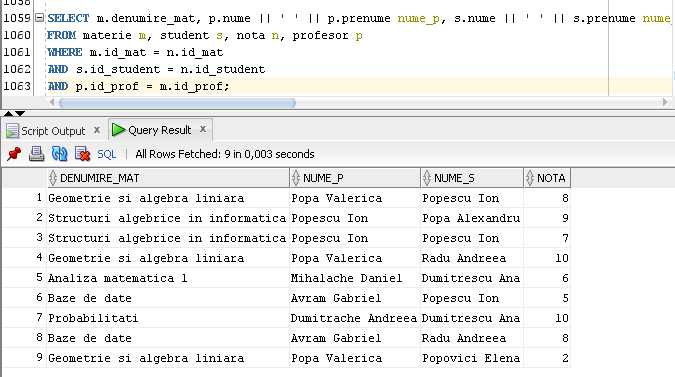
Verificăm faptul că într-adevăr nu au fost acordate note de 2:

Adăugăm o linie nouă în tabelul NOTA pentru a testa cazul în care un profesor a dat cele mai multe note de 2:

|  |
| --- |
| INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 2, 6, 2002); |

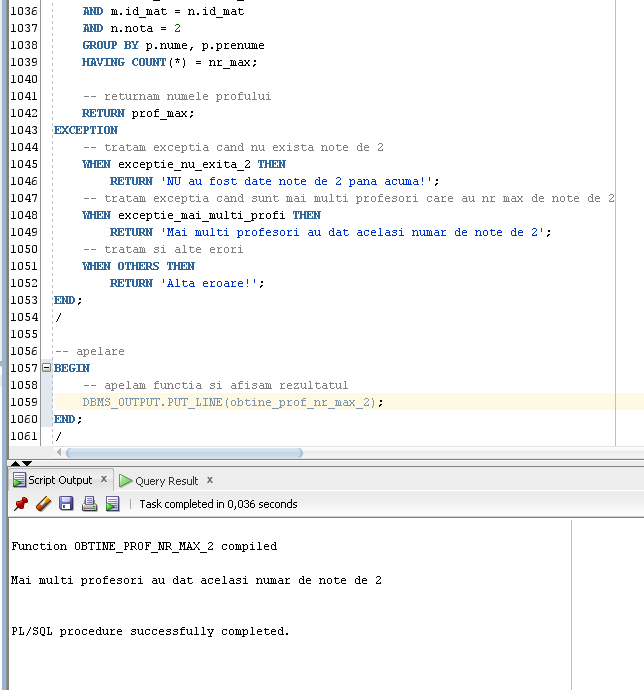
Noul output:

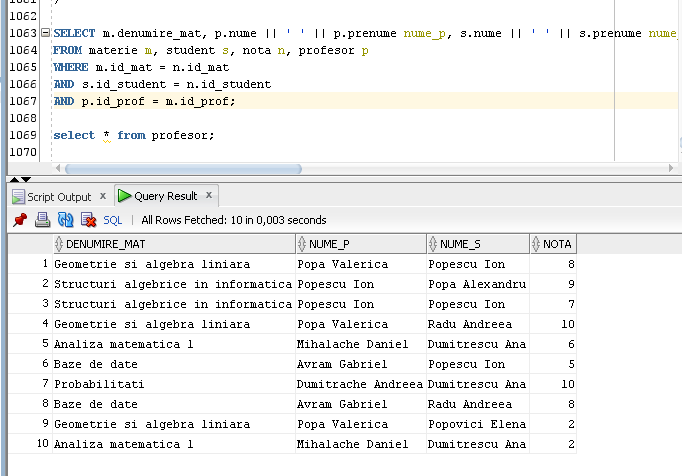
Verificăm (Popa Valerica este singura profesoară care a acordat note de 2):



Mai inserăm o notă de 2, dar la altă materie pentru a testa cazul în care mai mulți profesori au numărul cel mai mare de note de 2 date:

|  |
| --- |
| INSERT INTO Nota (id\_nota, nota, id\_student, id\_mat)  VALUES (seq\_nota\_id.NEXTVAL, 2, 9, 2012); |

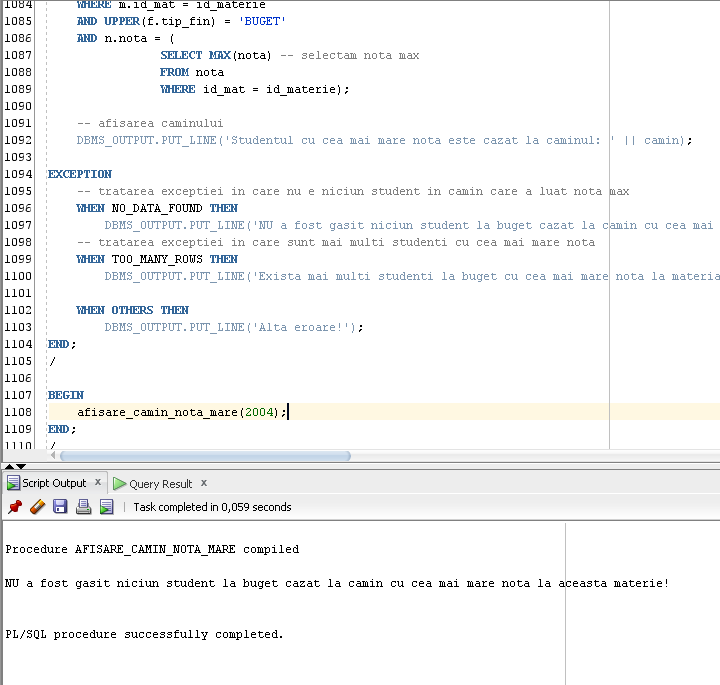
Noul output:

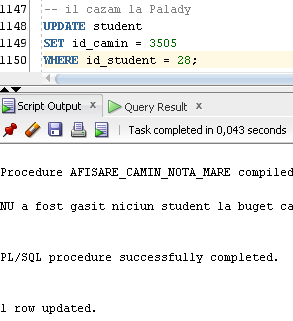
Verificăm (Atât Popa Valerica cât și Mihalache Daniel au dat câte o notă de 2):

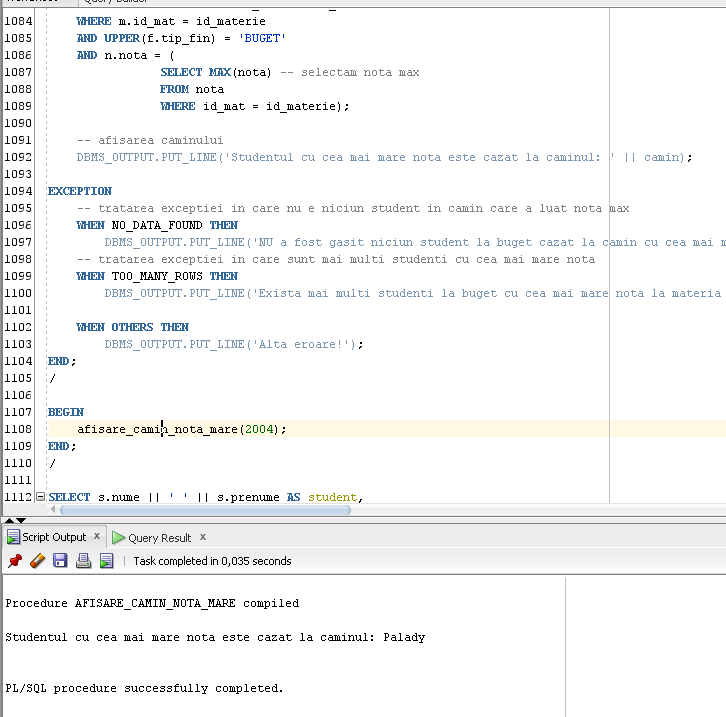
1. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip procedură care să aibă minim 2 parametri și să utilizeze într-o singură comandă SQL 5 dintre tabelele create. Definiți minim 2 excepții proprii, altele decât cele predefinite la nivel de sistem. Apelați subprogramul astfel încât să evidențiați toate cazurile definite și tratate.

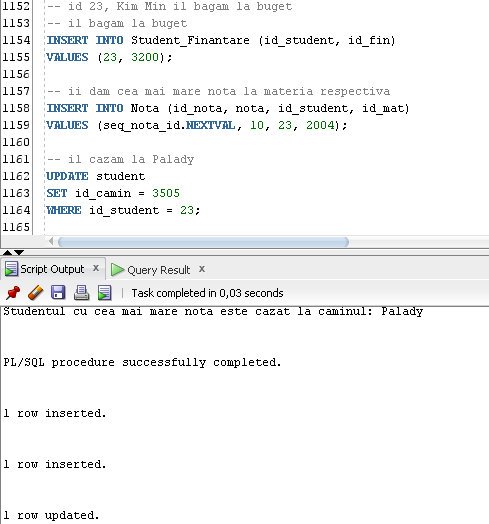
**CEIRNȚA:** Să se afișeze la ce cămin este cazat studentul de la buget cu cea mai mare notă la Structuri algebrice in informatica.

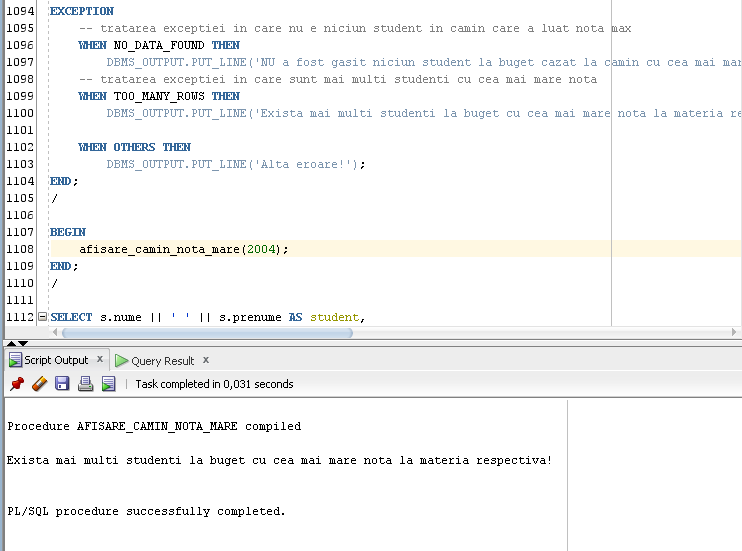
|  |
| --- |
| CREATE OR REPLACE PROCEDURE afisare\_camin\_nota\_mare(p\_id\_materie IN NUMBER, p\_tip\_fin VARCHAR2) IS  camin VARCHAR2(100); -- variabila pt denumirea caminului  nr\_studenti NUMBER; -- variabila pt nr de studenti gasiti    -- definim exceptii  exceptie\_nu\_exista\_studenti EXCEPTION;  exceptie\_mai\_multi\_studenti EXCEPTION;  BEGIN  SELECT COUNT(\*)  INTO nr\_studenti  FROM student s  JOIN nota n ON s.id\_student = n.id\_student  JOIN materie m ON n.id\_mat = m.id\_mat  JOIN camin c ON s.id\_camin = c.id\_camin  JOIN student\_finantare sf ON s.id\_student = sf.id\_student  JOIN finantare f ON sf.id\_fin = f.id\_fin  WHERE m.id\_mat = p\_id\_materie  AND UPPER(f.tip\_fin) = UPPER(p\_tip\_fin)  AND n.nota = (  SELECT MAX(nota) -- selectam nota max  FROM nota  WHERE id\_mat = p\_id\_materie);    IF nr\_studenti = 0 THEN  RAISE exceptie\_nu\_exista\_studenti;  ELSIF nr\_studenti > 1 THEN  RAISE exceptie\_mai\_multi\_studenti;  ELSE  SELECT c.denumire\_camin  INTO camin  FROM student s  JOIN nota n ON s.id\_student = n.id\_student  JOIN materie m ON n.id\_mat = m.id\_mat  JOIN camin c ON s.id\_camin = c.id\_camin  JOIN student\_finantare sf ON s.id\_student = sf.id\_student  JOIN finantare f ON sf.id\_fin = f.id\_fin  WHERE m.id\_mat = p\_id\_materie  AND UPPER(f.tip\_fin) = UPPER(p\_tip\_fin)  AND n.nota = (  SELECT MAX(nota) -- selectam nota max  FROM nota  WHERE id\_mat = p\_id\_materie);    -- afisarea caminului  DBMS\_OUTPUT.PUT\_LINE('Studentul cu cea mai mare nota este cazat la caminul: ' || camin);  END IF;    EXCEPTION  -- tratarea exceptiei in care nu e niciun student in camin care a luat nota max  WHEN exceptie\_nu\_exista\_studenti THEN  DBMS\_OUTPUT.PUT\_LINE('NU a fost gasit niciun student la buget cazat la camin cu cea mai mare nota la aceasta materie!');  -- tratarea exceptiei in care sunt mai multi studenti cu cea mai mare nota  WHEN exceptie\_mai\_multi\_studenti THEN  DBMS\_OUTPUT.PUT\_LINE('Exista mai multi studenti la buget cu cea mai mare nota la materia respectiva!');  WHEN OTHERS THEN  DBMS\_OUTPUT.PUT\_LINE('Alta eroare!');  END;  /  BEGIN  afisare\_camin\_nota\_mare(2004);  END;  / |

Testăm pentru materia cu id-ul 2004 Structuri algebrice in informatica, unde cea mai mare notă este obținută de studentul cu id-ul 28 care nu e cazat la niciun cămin și se află la Buget:

Acest student are dreptul la cămin așa că o să-l cazăm în Palady pentru a testa tot pentru materia cu id-ul 2004:

Testăm și se afișează căminul Palady (în tabela CAMIN există doar 2 studenți cazați la acest cămin, celălalt neavând note atribuite încă):

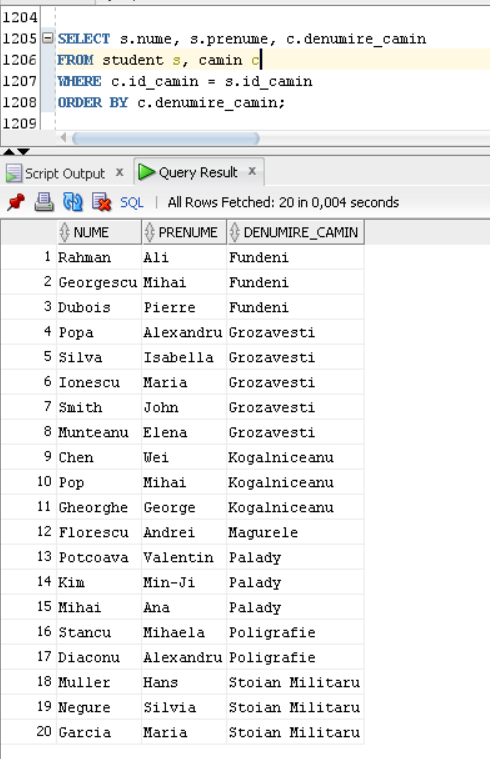
Mai adăugăm în NOTA un student cu nota maximă la această materie. Id-ul este 23, la Buget și îl introducem la materia cu id 2004, cu nota 10 și îl cazăm la căminul Palady:

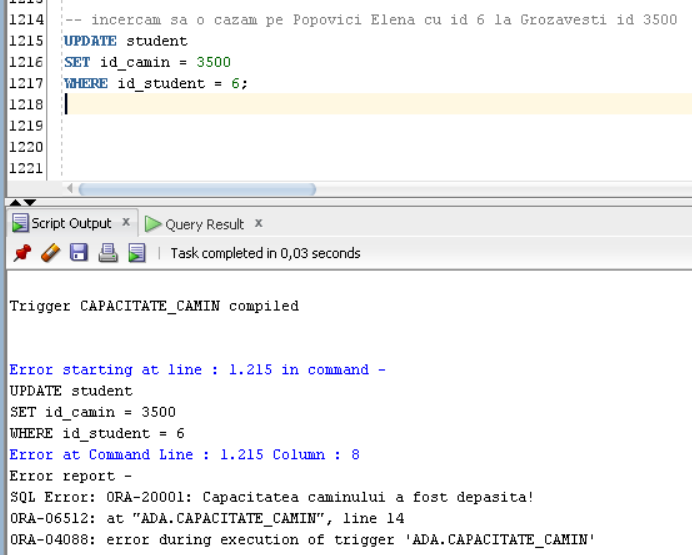
Testăm și ar trebui să se ridice excepția de exceptie\_mai\_multi\_studenti:

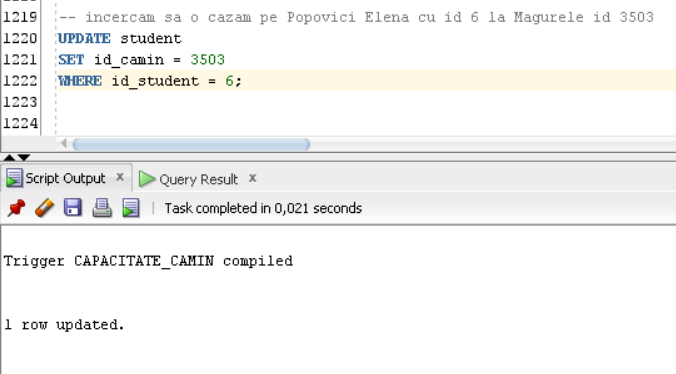
1. Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.

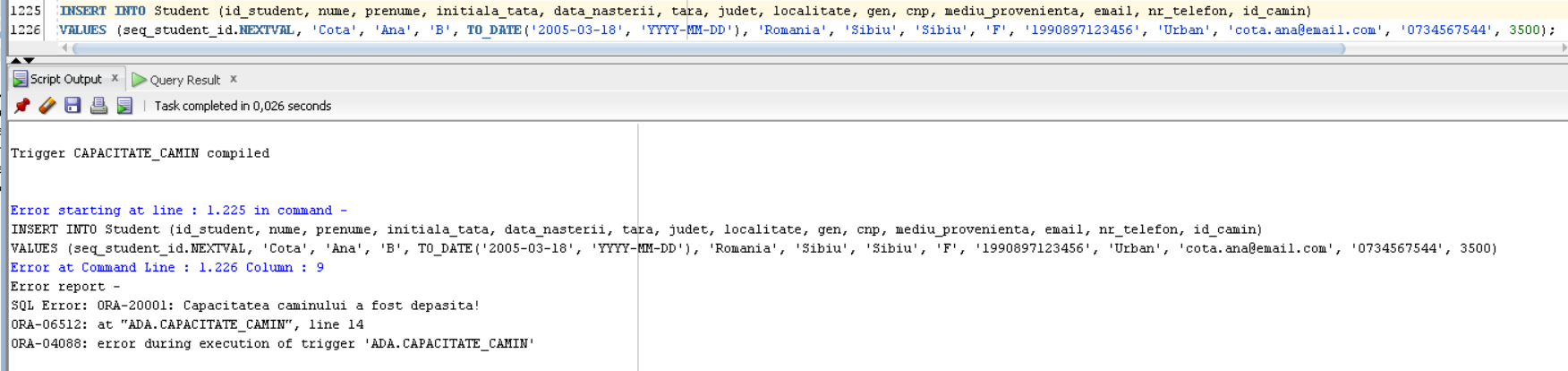
**CEIRNȚA:** Creați un trigger care nu va permite cazarea a mai mult de 6 studenți într-un singur cămin. (am ales 6 pentru că nu am mai mult de 5 studenți cazați într-un cămin, iar în căminul Grozăvești sunt cazați deja 5).

|  |
| --- |
| CREATE OR REPLACE TRIGGER capacitate\_camin  AFTER INSERT OR UPDATE ON student  DECLARE  maxim NUMBER;  BEGIN  -- verificam daca exista un camin care a depasit capacitatea  -- dupa comanda curenta  SELECT MAX(nr)  INTO maxim  FROM (SELECT ID\_CAMIN, COUNT(\*) AS nr  FROM student  WHERE id\_camin IS NOT NULL  GROUP BY id\_camin);    IF maxim > 6 THEN  RAISE\_APPLICATION\_ERROR(-20001, 'Capacitatea caminului a fost depasita!');  END IF;  END;  / |

Verificăm cam câți studenți sunt cazați la cămin și la ce cămin:

Încercăm să cazăm un al 6-lea student deja existent în baza de date, la căminul Grozăvești (id\_camin 3500). Testăm cu Popovici Elena (id\_student 6):

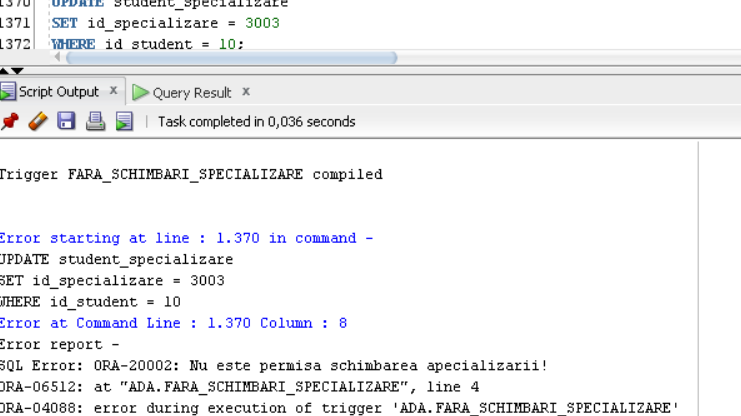
O să încerc să cazez același student la alt cămin, Magurele, unde mai există doar un singur student cazat în momentul de față:

Încercăm și o inserare:

1. Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

**CEIRNȚA:** Creați un trigger care nu permite schimbarea specializării. AICI AM RAMAS

|  |
| --- |
| CREATE OR REPLACE TRIGGER fara\_schimbari\_specializare  BEFORE UPDATE ON student\_specializare -- declansare inainte de orice update a tabelei  FOR EACH ROW -- se aplica pt incercarea inserarii fiecarei linii noi  BEGIN  IF :OLD.id\_specializare != :NEW.id\_specializare THEN  -- daca se incearca modificarea specializarii se ridica exceptia  RAISE\_APPLICATION\_ERROR(-20002, 'Nu este permisa schimbarea apecializarii!');  END IF;  END;  / |

Testăm pentru studentul Gheorghe George cu id-ul 10, pentru care am încercat schimbarea specializării, de la Matematica informatica, cu id 3004, la Matematica aplicata, cu id 3003:

1. Definiți un trigger de tip LDD. Declanșați trigger-ul.

**CEIRNȚA:** Creați un trigger LDD care afișează numele tabelului, cât și comanda LDD aplicată asupra acestuia.

|  |
| --- |
| CREATE OR REPLACE TRIGGER afisare\_nume\_tabel\_comanda  AFTER CREATE OR DROP OR ALTER ON SCHEMA -- declansare dupa comenziile specificate la nivel de schema  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Comanda: ' || SYS.SYSEVENT || ' pe tabela: ' || SYS.DICTIONARY\_OBJ\_NAME);  END;  /  -- testare  CREATE TABLE tabel\_test (  id\_test NUMBER  );  DROP TABLE tabel\_test; |

