ACADEMIA DE STUDII ECONOMICE DIN BUCURESTI

FACULTATEA DE CIBERNETICA, STATISTICA SI INFORMATICA ECONOMICA

SPECIALIZAREA INFORMATICA ECONOMICA

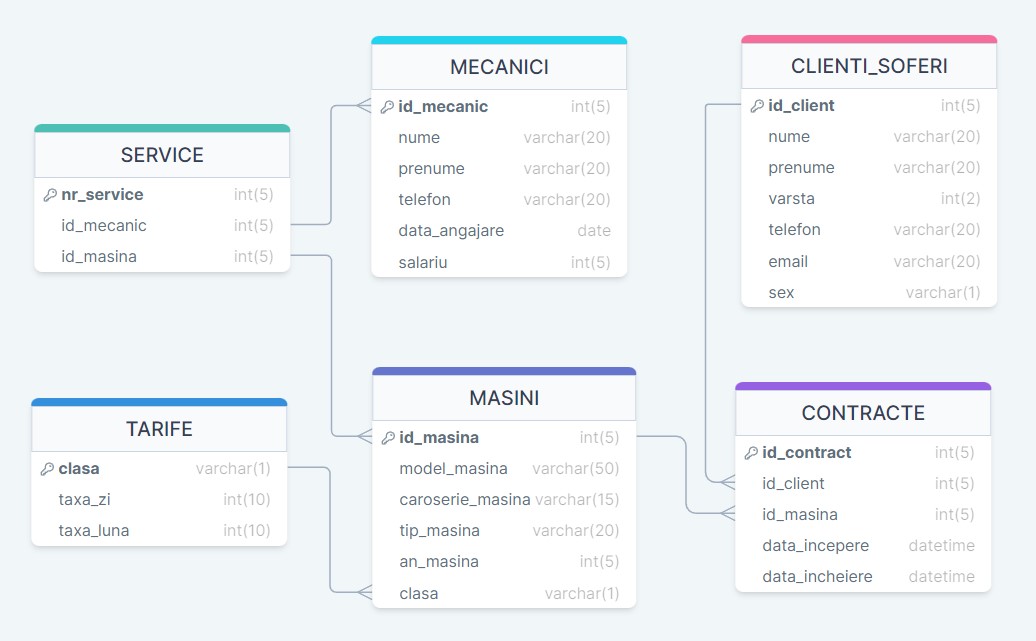


**PROIECT BAZE DE DATE**

**IMPLEMENTAREA UNEI BAZE DE DATE PENTRU O FIRMA DE ÎNCHIRIERI AUTO**

Bucuresti

2023



**Figura 1. *Schema initiala BD pentru activitatea de evidenta a comenzilor si a resurselor umane***

**PREZENTARE SCURTA A PROIECTULUI**

Baza de date este concepută pentru gestionarea unui serviciu de închiriere auto. Scopul principal al proiectului este să ofere o platformă pentru înregistrarea contractelor de închiriere între clienți și firmă, evidențierea informațiilor despre mașinile disponibile pentru închiriere, precum și gestionarea activităților de service auto.

**Descrierea tabelelor:**

1. **TARIFE:**
   * clasa (cheie primară): specifică clasa mașinilor (P - premium, B - business, E - economic)
   * taxa\_zi: taxa de închiriere/zi pentru clasa respectivă
   * taxa\_luna: taxa de închiriere/luna pentru clasa respectivă
2. **MASINI:**
   * id\_masina (cheie primară): identificator unic pentru fiecare mașină
   * model\_masina: modelul mașinii
   * caroserie\_masina: tipul caroseriei mașinii
   * tip\_masina: tipul de combustibil al mașinii
   * an\_masina: anul fabricației mașinii
   * clasa (cheie străină): face referire la clasa din tabelul TARIFE
3. **CLIENTI\_SOFERI:**
   * id\_client (cheie primară): identificator unic pentru fiecare client
   * nume, prenume: numele și prenumele clientului
   * varsta: vârsta clientului
   * telefon, email, sex: informații de contact și sexul clientului
4. **CONTRACTE:**
   * id\_contract (cheie primară): identificator unic pentru fiecare contract
   * id\_client (cheie străină): face referire la id\_client din tabelul CLIENTI\_SOFERI
   * id\_masina (cheie străină): face referire la id\_masina din tabelul MASINI
   * data\_incepere, data\_incheiere: perioada de închiriere
5. **MECANICI:**
   * id\_mecanic (cheie primară): identificator unic pentru fiecare mecanic
   * nume, prenume: numele și prenumele mecanicului
   * telefon: numărul de telefon al mecanicului
   * data\_angajare: data angajării mecanicului
   * salariu: salariul mecanicului
6. **SERVICE:**
   * nr\_service (cheie primară): identificator unic pentru fiecare serviciu
   * id\_mecanic (cheie străină): face referire la id\_mecanic din tabelul MECANICI
   * id\_masina (cheie străină): face referire la id\_masina din tabelul MASINI

**CREAREA TABELELOR**

CREATE TABLE TARIFE(

clasa CHAR(1) CONSTRAINT pk\_tarife PRIMARY KEY,

taxa\_zi NUMBER(8,2),

taxa\_luna NUMBER(8,2)

);

CREATE TABLE MASINI(

id\_masina NUMBER(5) CONSTRAINT pk\_masini PRIMARY KEY,

model\_masina VARCHAR2(30),

caroserie\_masina VARCHAR2(15),

tip\_masina VARCHAR2(15),

an\_masina NUMBER(4),

clasa CHAR(1),

CONSTRAINT fk\_masini\_tarife FOREIGN KEY (clasa) REFERENCES TARIFE(clasa)

);

CREATE TABLE CLIENTI\_SOFERI(

id\_client NUMBER(5) CONSTRAINT pk\_clienti PRIMARY KEY,

nume VARCHAR2(20),

prenume VARCHAR2(20),

varsta NUMBER(2) CONSTRAINT ck\_clienti\_varsta CHECK (varsta>=18),

telefon VARCHAR2(20),

email VARCHAR2(30),

sex CHAR(1)

);

CREATE TABLE CONTRACTE(

id\_contract NUMBER(5) CONSTRAINT pk\_contracte PRIMARY KEY,

id\_client NUMBER(5),

id\_masina NUMBER(5),

data\_incepere DATE DEFAULT SYSDATE,

data\_incheiere DATE NOT NULL,

CONSTRAINT fk\_contracte\_clienti FOREIGN KEY (id\_client) REFERENCES CLIENTI\_SOFERI(id\_client),

CONSTRAINT fk\_contracte\_masini FOREIGN KEY(id\_masina) REFERENCES MASINI(id\_masina),

CONSTRAINT ck\_contracte\_data CHECK (data\_incheiere > data\_incepere)

);

CREATE TABLE MECANICI(

id\_mecanic NUMBER(5) CONSTRAINT pk\_mecanici PRIMARY KEY,

nume VARCHAR2(20),

prenume VARCHAR2(20),

telefon VARCHAR2(20),

data\_angajare DATE DEFAULT SYSDATE,

salariu NUMBER(5,2)

);

CREATE TABLE SERVICE(

nr\_service NUMBER(5) CONSTRAINT pk\_service PRIMARY KEY,

id\_mecanic NUMBER(5) CONSTRAINT fk\_service\_mecanici REFERENCES MECANICI(id\_mecanic),

id\_masina NUMBER(5) CONSTRAINT fk\_service\_masini REFERENCES MASINI(id\_masina)

);

**INSERAREA DATELOR IN TABELE**

-- TARIFE /\*P = premium class | B = business class | E = economic class \*/

* INSERT INTO TARIFE (clasa, taxa\_zi, taxa\_luna)

VALUES ('P', 26, 800.0);

* INSERT INTO TARIFE (clasa, taxa\_zi, taxa\_luna)

VALUES ('B', 16.5, 511.5);

* INSERT INTO TARIFE (clasa, taxa\_zi, taxa\_luna)

VALUES ('E', 10, 310.0);

-- MASINI

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (1, 'Toyota Camry', 'Sedan', 'Benzina', 2022, 'B');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (2, 'BMW X5', 'SUV', 'Hibrid', 2022, 'P');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (3, 'Smart Fortwo', 'Micro', 'Electric', 2023, 'E');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (4, 'Renault Clio', 'Hatchback', 'Benzina', 2022, 'E');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (5, 'Audi A3', 'Sedan', 'Diesel', 2022, 'P');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (6, 'Ford Fiesta', 'Hatchback', 'Benzina', 2015, 'E');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (7, 'Mercedes-Benz GLC', 'SUV', 'Hibrid', 2023, 'P');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (8, 'Volkswagen Golf', 'Hatchback', 'Diesel', 2018, 'E');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (9, 'Nissan Qashqai', 'Crossover', 'Diesel', 2022, 'B');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (10, 'Toyota Prius', 'Hatchback', 'Electric', 2020, 'P');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (11, 'Mazda MX-5', 'Roadster', 'Benzina', 2020, 'B');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (12, 'Dacia Spring', 'Crossover', 'Electric', 2021, 'E');

* INSERT INTO MASINI (id\_masina, model\_masina, caroserie\_masina, tip\_masina, an\_masina, clasa)

VALUES (13, 'Hyundai Kona', 'SUV', 'Electric', 2021, 'B');

-- CLIENTI

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (1, 'Ionescu', 'Mircea', 30, '0736472819', 'mircea.ionescu@gmail.com', 'M');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (2, 'Sandulescu', 'Alice', 25, '0782819374', 'alice.sandulescu@gmail.com', 'F');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (3, 'Popescu', 'Ion', 40, '0725123456', 'ion.popescu@gmail.com', 'M');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (4, 'Vasilescu', 'Elena', 35, '0766789012', 'elena.vasilescu@gmail.com', 'F');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (5, 'Georgescu', 'Cristina', 28, '0723112233', 'cristina.georgescu@gmail.com', 'F');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (6, 'Dumitru', 'Alexandru', 32, '0733445566', 'alex.dumitru@gmail.com', 'M');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (7, 'Stanescu', 'Mihaela', 29, '0767890123', 'mihaela.stanescu@gmail.com', 'F');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (8, 'Radulescu', 'Marian', 38, '0723556677', 'marian.radulescu@gmail.com', 'M');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (9, 'Iordache', 'Andreea', 26, '0788445566', 'andreea.iordache@gmail.com', 'F');

* INSERT INTO CLIENTI\_SOFERI (id\_client, nume, prenume, varsta, telefon, email, sex)

VALUES (10, 'Florescu', 'Gabriel', 33, '0766123456', 'gabriel.florescu@gmail.com', 'M');

-- CONTRACTE

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (1, 1, 1, TO\_DATE('01-10-2023', 'DD-MM-YYYY'), TO\_DATE('10-10-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (2, 2, 3, TO\_DATE('15-10-2023', 'DD-MM-YYYY'), TO\_DATE('01-11-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (3, 10, 5, TO\_DATE('05-11-2023', 'DD-MM-YYYY'), TO\_DATE('15-11-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (4, 6, 10, TO\_DATE('20-11-2023', 'DD-MM-YYYY'), TO\_DATE('25-11-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (5, 4, 13, TO\_DATE('01-12-2023', 'DD-MM-YYYY'), TO\_DATE('10-12-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (6, 3, 8, TO\_DATE('15-12-2023', 'DD-MM-YYYY'), TO\_DATE('20-12-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (7, 8, 7, TO\_DATE('16-12-2023', 'DD-MM-YYYY'), TO\_DATE('18-12-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (8, 9, 4, TO\_DATE('03-10-2023', 'DD-MM-YYYY'), TO\_DATE('09-10-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (9, 7, 12, TO\_DATE('10-10-2023', 'DD-MM-YYYY'), TO\_DATE('21-11-2023', 'DD-MM-YYYY'));

* INSERT INTO CONTRACTE (id\_contract, id\_client, id\_masina, data\_incepere, data\_incheiere)

VALUES (10, 5, 6, TO\_DATE('23-10-2023', 'DD-MM-YYYY'), TO\_DATE('28-10-2023', 'DD-MM-YYYY'));

--MECANICI

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (1, 'Iliescu', 'David', '0736192877', TO\_DATE('30-09-2020', 'DD-MM-YYYY'), 5000.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (2, 'Traian', 'Emma', '0782915647', TO\_DATE('28-09-2021', 'DD-MM-YYYY'), 5500.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (3, 'Popescu', 'Ion', '0754321987', TO\_DATE('15-03-2022', 'DD-MM-YYYY'), 5200.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (4, 'Dragomir', 'Elena', '0723456123', TO\_DATE('10-08-2022', 'DD-MM-YYYY'), 4800.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (5, 'Munteanu', 'Gabriel', '0765432198', TO\_DATE('05-12-2022', 'DD-MM-YYYY'), 5100.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (6, 'Florescu', 'Ana-Maria', '0776543219', TO\_DATE('20-02-2023', 'DD-MM-YYYY'), 4900.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (7, 'Radulescu', 'Adrian', '0743219876', TO\_DATE('15-05-2023', 'DD-MM-YYYY'), 5300.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (8, 'Iordache', 'Georgiana', '0798765432', TO\_DATE('10-09-2023', 'DD-MM-YYYY'), 5000.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (9, 'Stefanescu', 'Valentin', '0787654321', TO\_DATE('02-01-2024', 'DD-MM-YYYY'), 5400.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (10, 'Barbu', 'Elena', '0712345678', TO\_DATE('02-01-2024', 'DD-MM-YYYY'), 4800.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (11, 'Dumitrescu', 'Alexandru', '0765432198', TO\_DATE('10-11-2023', 'DD-MM-YYYY'), 5200.00);

* INSERT INTO MECANICI (id\_mecanic, nume, prenume, telefon, data\_angajare, salariu)

VALUES (12, 'Balan', 'Cristina', '0723456123', TO\_DATE('10-11-2023', 'DD-MM-YYYY'), 5000.00);

--SERVICE

* INSERT INTO SERVICE (nr\_service, id\_mecanic, id\_masina)

VALUES (1, 1, 2);

* INSERT INTO SERVICE (nr\_service, id\_mecanic, id\_masina)

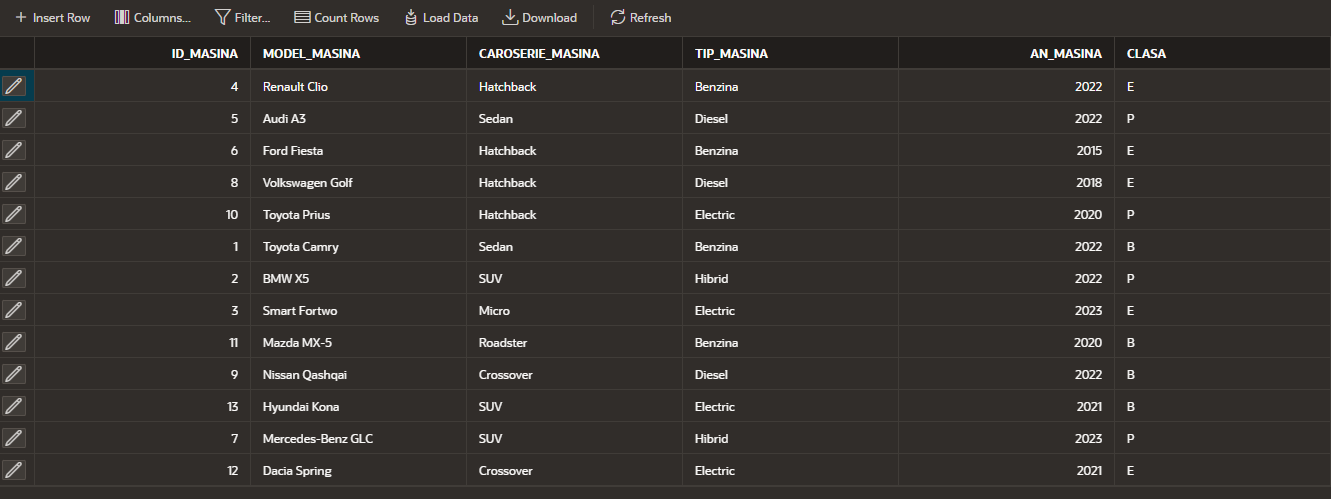
VALUES (2, 5, 11);

* INSERT INTO SERVICE (nr\_service, id\_mecanic, id\_masina)

VALUES (3, 8, 9);

**UTILIZAREA COMENZILOR CREATE,ALTER,DROP,INSERT,UPDATE,DELETE**

**Cerinta 1:** Doresc schimbarea id\_masina NUMBER in nr\_masina VARCHAR2



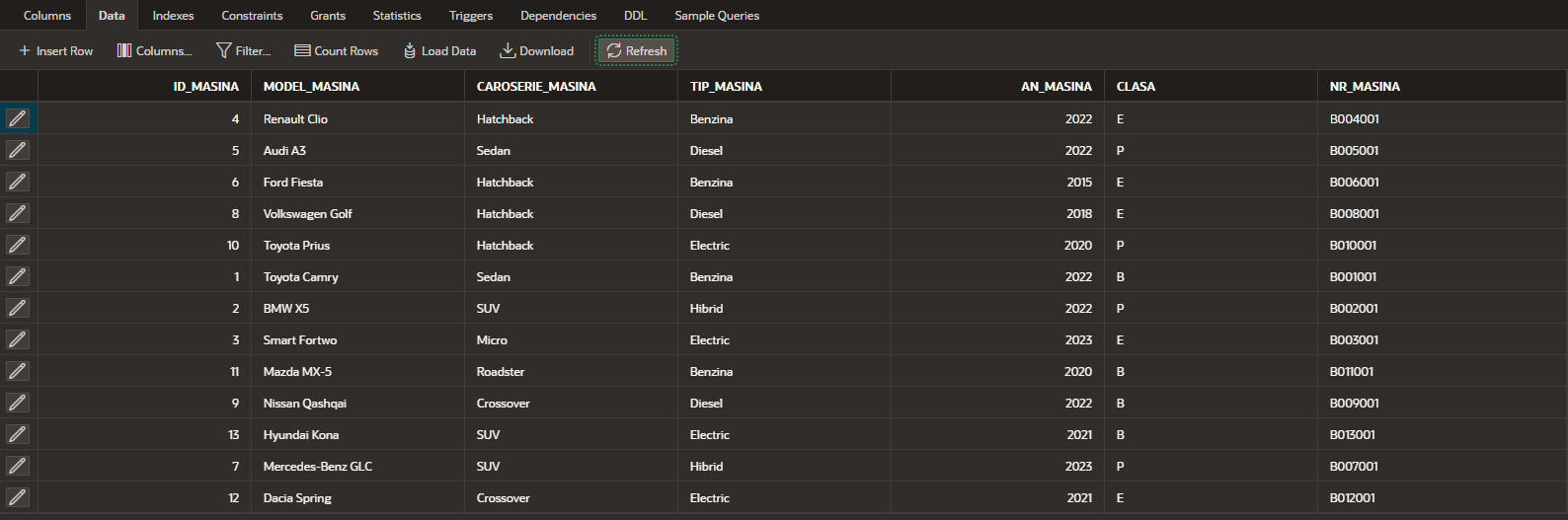
**Etapa 1: Adaug coloana nr\_masina**

ALTER TABLE MASINI ADD (nr\_masina VARCHAR2(7));

**Etapa 2: Adaug inregistrarile (prin comanda update)**

UPDATE MASINI SET nr\_masina =:new\_nr\_masina

where id\_masina=:existing\_nr\_masina;



**Etapa3: Sterg foreign key-urile associate cu coloana id\_masina + primary key**

ALTER TABLE CONTRACTE DROP CONSTRAINT fk\_contracte\_masini;

ALTER TABLE SERVICE DROP CONSTRAINT fk\_service\_masini;

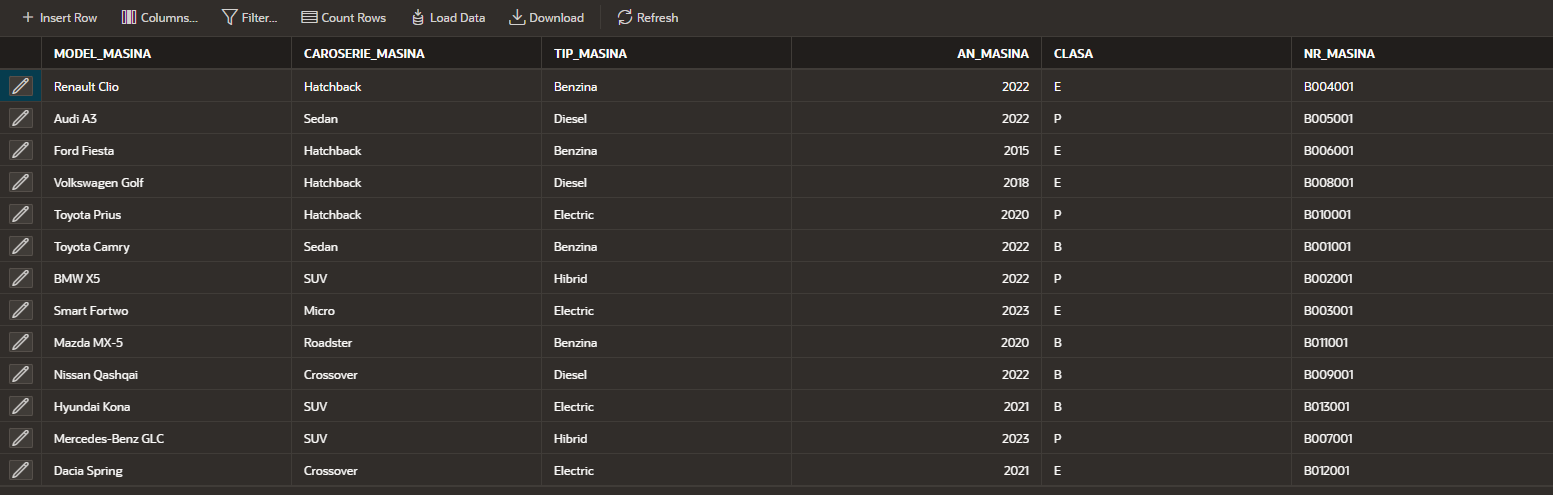
ALTER TABLE MASINI DROP CONSTRAINT pk\_masini;

**Etapa4: Sterg coloana id\_masina**

ALTER TABLE MASINI DROP COLUMN id\_masina;

**Etapa5: Definesc nr\_masina ca noua primary key**

ALTER TABLE MASINI ADD CONSTRAINT pk\_masini PRIMARY KEY(nr\_masina);

****

****

**Repet etapa 1,2,4 pentru tabelele ce au legatura cu masini + Etapa 5 (definite ca foreign key)**

ALTER TABLE SERVICE ADD (nr\_masina VARCHAR2(7));

UPDATE SERVICE SET nr\_masina =:new\_nr\_masina

where id\_masina=:existing\_nr\_masina;

ALTER TABLE SERVICE DROP COLUMN id\_masina;

ALTER TABLE SERVICE ADD CONSTRAINT fk\_service\_masini FOREIGN KEY(nr\_masina) REFERENCES MASINI(nr\_masina);

ALTER TABLE CONTRACTE ADD (nr\_masina VARCHAR2(7));

UPDATE CONTRACTE SET nr\_masina =:new\_nr\_masina

where id\_masina=:existing\_nr\_masina;

ALTER TABLE CONTRACTE DROP COLUMN id\_masina;

ALTER TABLE CONTRACTE ADD CONSTRAINT fk\_contracte\_masini FOREIGN KEY(nr\_masina) REFERENCES MASINI(nr\_masina);

**Cerinta 2:** Doresc adaugarea unei noi tabele, apoi stergerea ei si recuperarea sa

CREATE TABLE PIESE(

id\_piesa NUMBER(6) CONSTRAINT pk\_testare PRIMARY KEY,

denumire VARCHAR2(20),

pret NUMBER(8,2)

);

DROP TABLE PIESE CASCADE CONSTRAINTS;

FLASHBACK TABLE PIESE TO BEFORE DROP;

**Cerinta 3:** Doresc adaugarea datelor in tabela PIESE si crearea unei legaturi cu tabela SERVICE

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (1, 'Ulei motor', 150.99);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (2, 'Filtru de aer', 112.5);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (3, 'Plăcuțe de frână', 280.0);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (4, 'Baterie auto', 720.0);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (5, 'Lampă far', 340.5);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (6, 'Filtru de ulei', 115.0);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (7, 'Bujie', 88.75);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (8, 'Disc de frână', 265.0);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (9, 'Amortizor', 55.0);

INSERT INTO PIESE (id\_piesa, denumire, pret) VALUES (10, 'Alternator', 700.0);

CREATE TABLE PIESE\_SERVICE(

id\_piesa NUMBER(6),

nr\_service NUMBER(5),

cantitate NUMBER(5),

pret NUMBER(8),

CONSTRAINT fk\_ps\_piese FOREIGN KEY (id\_piesa) REFERENCES PIESE(id\_piesa),

CONSTRAINT fk\_ps\_service FOREIGN KEY (nr\_service) REFERENCES SERVICE(nr\_service)

);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (1, 1, 2, 301.98);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (2, 1, 1, 112.5);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (3, 2, 1, 280.0);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (4, 2, 1, 720.0);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (5, 3, 2, 681.0);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (6, 3, 3, 345.0);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (7, 1, 2, 177.5);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

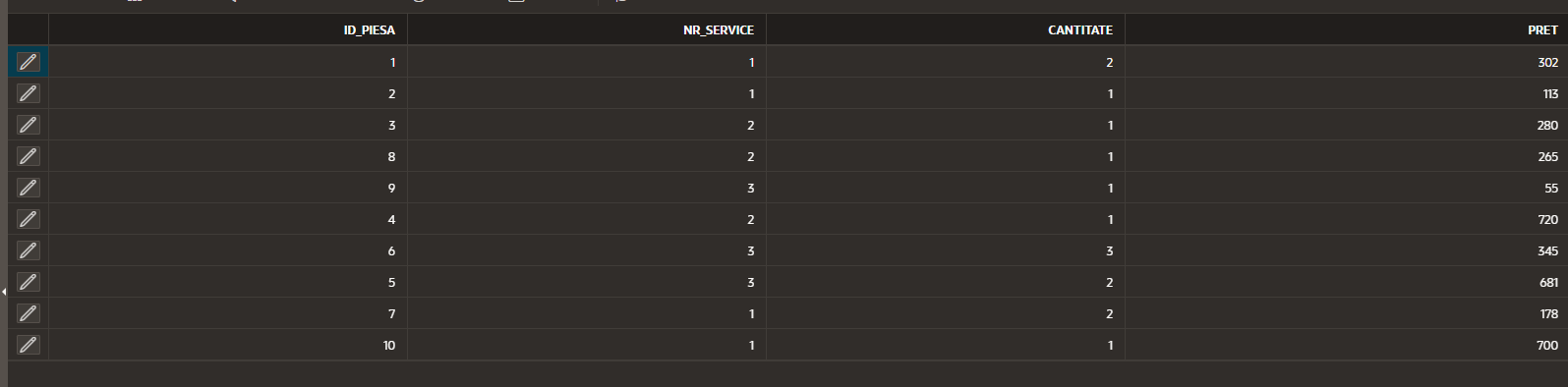
VALUES (8, 2, 1, 265.0);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (9, 3, 1, 55.0);

INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (10, 1, 1, 700.0);



INSERT INTO PIESE\_SERVICE (id\_piesa, nr\_service, cantitate, pret)

VALUES (6, 2, 1, 115.0);

UPDATE PIESE\_SERVICE SET cantitate=2,pret=230.0 WHERE id\_piesa=6 AND nr\_service=2;

DELETE FROM PIESE\_SERVICE WHERE id\_piesa=6 AND nr\_service=2;

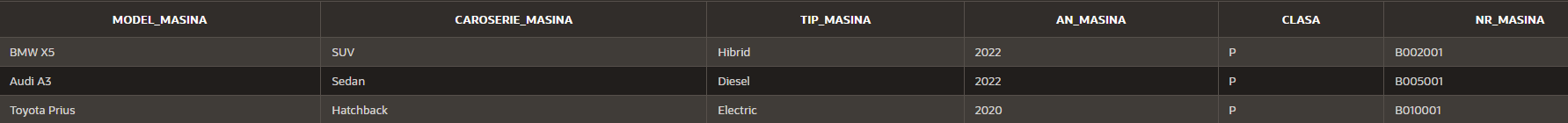


**Figura 2. *Schema FINALA BD pentru activitatea de evidenta a comenzilor si a resurselor umane***

**EXEMPLE DE INTEROGARI**

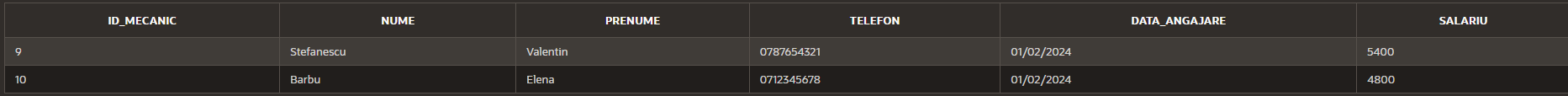
1/ Sa se selecteze toate masinile care au anul mai mic de 2023 si care sunt in clasa premium.

SELECT \* FROM masini WHERE an\_masina<2023 AND lower(clasa)='p' ORDER BY NR\_MASINA;



2/ Sa se selecteze toti mecanicii care au fost angajati in 2024

SELECT \* FROM MECANICI WHERE EXTRACT(YEAR FROM data\_angajare)=2024 ORDER BY id\_mecanic;

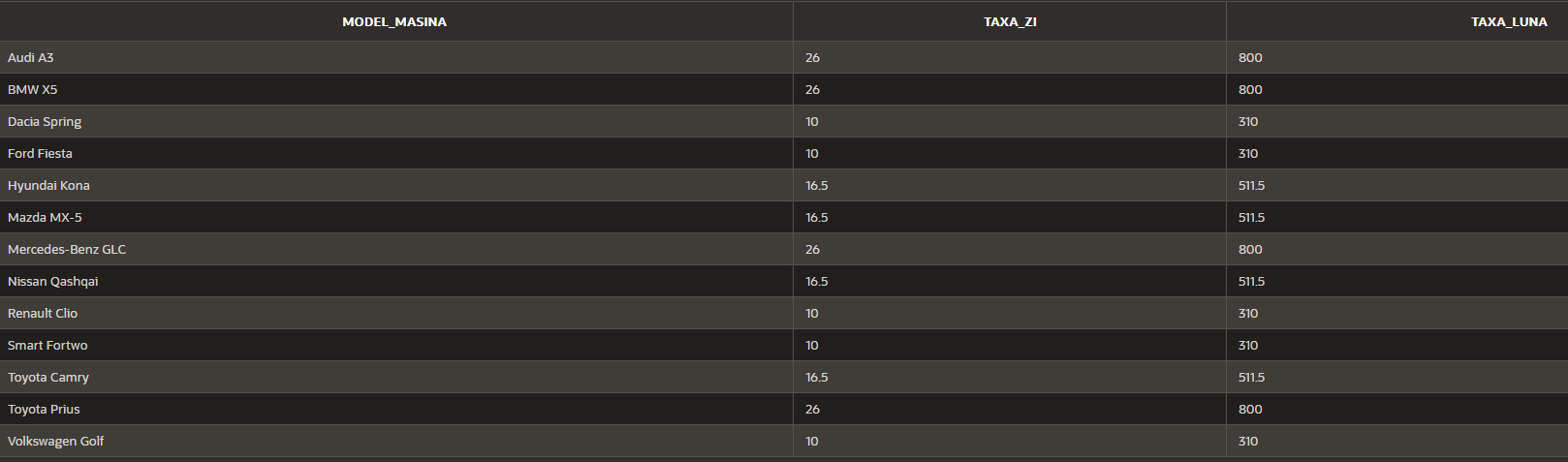


3/ Sa se afiseze denumirile masinilor si taxele aferente lor

SELECT m.model\_masina, t.taxa\_zi,t.taxa\_luna FROM MASINI m, TARIFE t

WHERE t.clasa=m.clasa

ORDER BY model\_masina;



4/ Sa se afiseze salariul si numele tuturor mecanicilor care au lucrat in service si care au salariul mai mare decat oricare dintre mecanicii care nu au lucrat in service

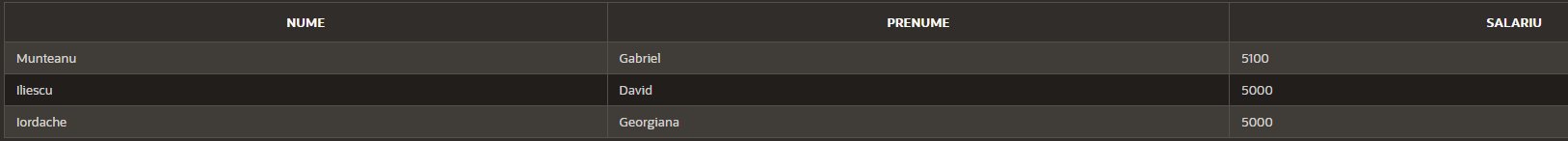
SELECT m.nume,m.prenume,m.salariu FROM mecanici m, service s

WHERE m.id\_mecanic=s.id\_mecanic

AND m.salariu>ANY

(SELECT m2.salariu FROM mecanici m2

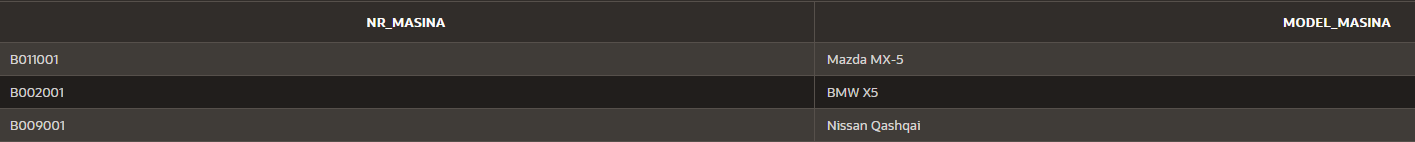
WHERE m2.id\_mecanic NOT IN(SELECT s2.id\_mecanic FROM service s2));



5/ Sa se afiseze masinile care se afla in service

SELECT m.nr\_masina,m.model\_masina FROM masini m

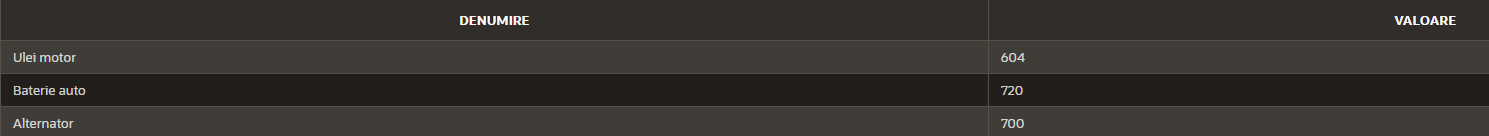
INNER JOIN service s ON m.nr\_masina=s.nr\_masina;



6/ Sa se calculeze valoarea fiecarei piese (valoare = cantitate \* pret) cuprinsa intre 500 si 1000 si sa se afiseze denumirea piesei si valoarea comenzii

SELECT p.denumire,ps.cantitate \* ps.pret AS valoare FROM piese p, piese\_service ps

WHERE p.id\_piesa=ps.id\_piesa AND ps.cantitate \* ps.pret BETWEEN 500 AND 1000;



7/ Sa se afiseze mecanicii care au acelasi salariu cu mecanicul ce are numele Iliescu si prenumele David

SELECT \* FROM mecanici

WHERE salariu=ANY(SELECT salariu FROM mecanici WHERE lower(nume)='iliescu' AND lower(prenume)='david')

ORDER BY data\_angajare;



8/ Sa se afiseze toate piesele utilizate in service, cat si pretul si cantitatea

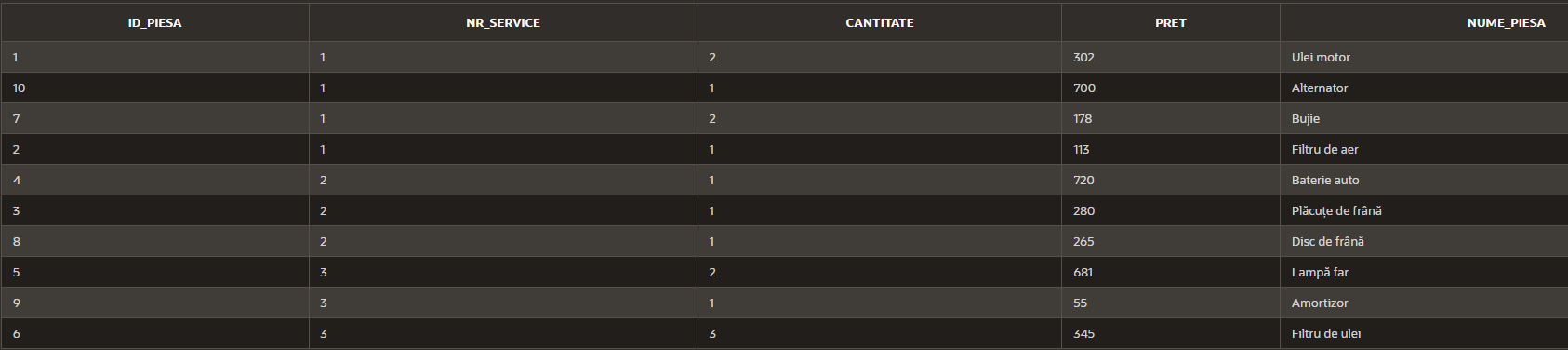
SELECT ps.\*, p.denumire AS nume\_piesa

FROM PIESE\_SERVICE ps

JOIN PIESE p ON ps.id\_piesa = p.id\_piesa

JOIN SERVICE s ON ps.nr\_service = s.nr\_service

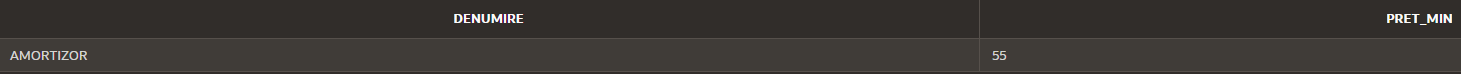
ORDER BY ps.nr\_service;



9/ Sa se afiseze cu litere mari piesele care au pretul cel mai mic

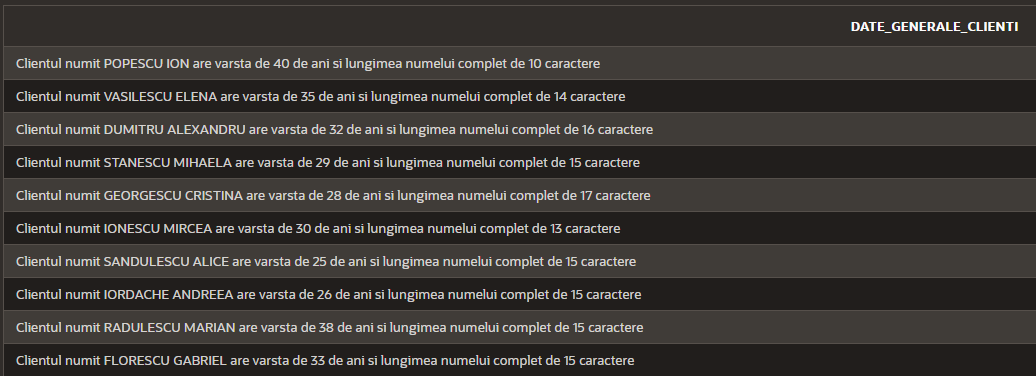
SELECT UPPER(denumire) AS DENUMIRE,pret\_min FROM PIESE

WHERE pret\_min=(SELECT MIN(pret\_min) FROM PIESE);



10/ Sa se afiseze date generale ale clientilor (numele clientilor concatenată cu varsta şi lungimea prenumelui)

SELECT 'Clientul numit ' || UPPER(nume) ||' '|| UPPER(prenume) || ' are varsta de ' ||varsta|| ' de ani si lungimea numelui complet de ' ||(LENGTH(nume)+LENGTH(prenume))|| ' caractere' AS DATE\_GENERALE\_CLIENTI FROM CLIENTI\_SOFERI;



11/ Sa se afiseze nr\_service,data\_service,numarul de luni intre data curenta si data\_service si data corespunzatoare dupa o luna de la service

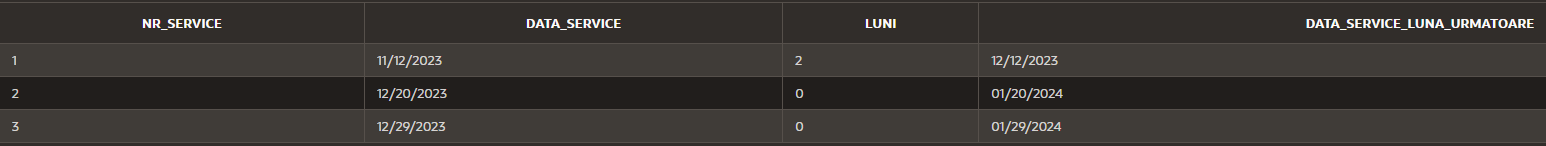
SELECT nr\_service,data\_service,

round(MONTHS\_BETWEEN(SYSDATE,data\_service)) AS LUNI,

ADD\_MONTHS(data\_service,1) AS DATA\_SERVICE\_LUNA\_URMATOARE

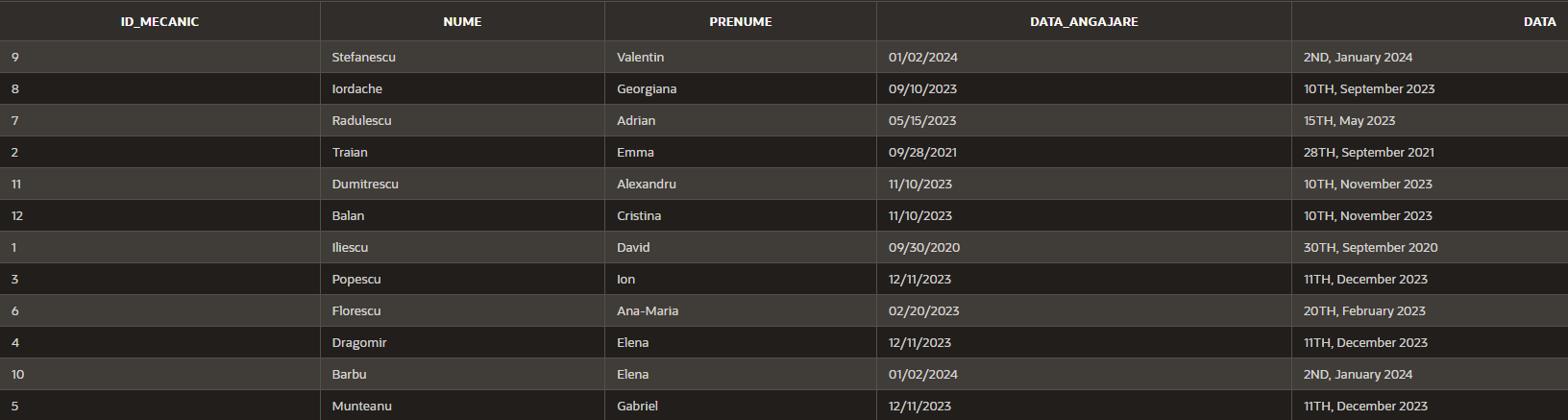
FROM SERVICE

ORDER BY DATA\_SERVICE;



12/ Să se afişeze id\_mecanic,nume,prenume si data angajarii in formatul initial si in format 'fmDDth, Month YYYY'

SELECT id\_mecanic,nume,prenume,data\_angajare,TO\_CHAR(data\_angajare,'fmDDth, Month YYYY')AS data FROM MECANICI;

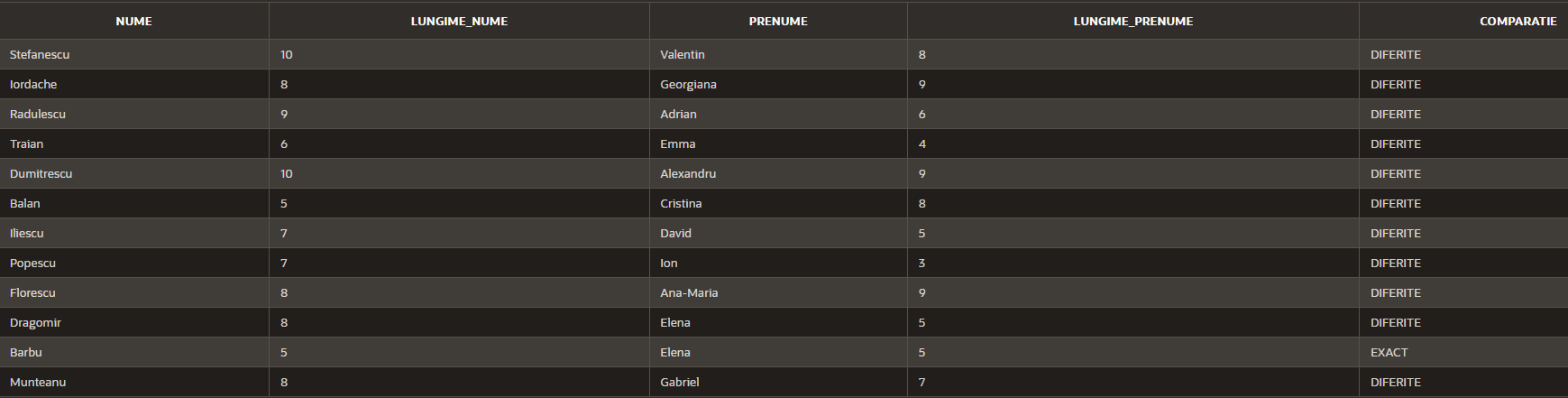


13/ Sa se afiseze, din mecanici, lungimea numelui, lungimea prenumelui, daca acestea sunt egale sa se returneze 'exact', iar daca nu sunt egale sa se returneze ’diferite’

SELECT nume,LENGTH(nume) AS lungime\_nume,

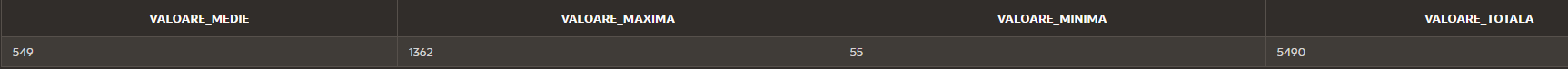
prenume,LENGTH(prenume) AS lungime\_prenume,

NVL2((NULLIF(LENGTH(nume),LENGTH(prenume))),'DIFERITE','EXACT') AS comparatie FROM mecanici;



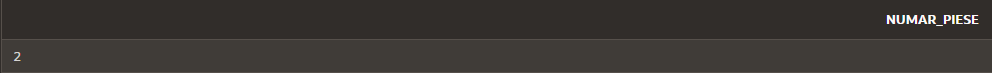
14/ Să se afişeze valoarea maximă, valoarea medie, valoarea minimă şi valoarea totală a pieselor comandate

SELECT AVG(cantitate\*pret) AS VALOARE\_MEDIE, MAX(cantitate\*pret) AS VALOARE\_MAXIMA, MIN(cantitate\*pret) AS VALOARE\_MINIMA, sum(cantitate\*pret) AS VALOARE\_TOTALA FROM piese\_service;



15/ Sa se afiseze numarul de piese care au pretul minim < 100

SELECT COUNT(\*) AS NUMAR\_PIESE FROM PIESE WHERE pret\_min<100



16/ Sa se afiseze id\_piesa, numele pieselor si valoarea totala pentru cele a caror valoare este mai mica decat media.

SELECT p.id\_piesa, p.denumire, sum(ps.pret\*ps.cantitate) AS VALOARE\_TOTALA

FROM piese p, piese\_service ps WHERE p.id\_piesa=ps.id\_piesa

GROUP BY p.id\_piesa,p.denumire

HAVING sum(ps.pret\*ps.cantitate) < ANY(SELECT AVG(a.pret\*a.cantitate) FROM piese\_service a)

ORDER BY VALOARE\_TOTALA desc;



17/ Sa se aplice o reducere clientilor care au inchiriat masini in luna:

* octombrie: 5%,
* noiembrie: 10%,
* decembrie: 15%

SELECT c.id\_client,CONCAT(c.nume,c.prenume) AS nume\_client, co.nr\_masina,t.taxa\_zi AS taxa\_initiala,

(CASE WHEN EXTRACT(MONTH FROM co.data\_incepere)=10 THEN ROUND(t.taxa\_zi\*0.95,2)

WHEN EXTRACT(MONTH FROM co.data\_incepere)=11 THEN ROUND(t.taxa\_zi\*0.9,2)

WHEN EXTRACT(MONTH FROM co.data\_incepere)=12 THEN ROUND(t.taxa\_zi\*0.85,2)

ELSE t.taxa\_zi END) AS taxa\_finala

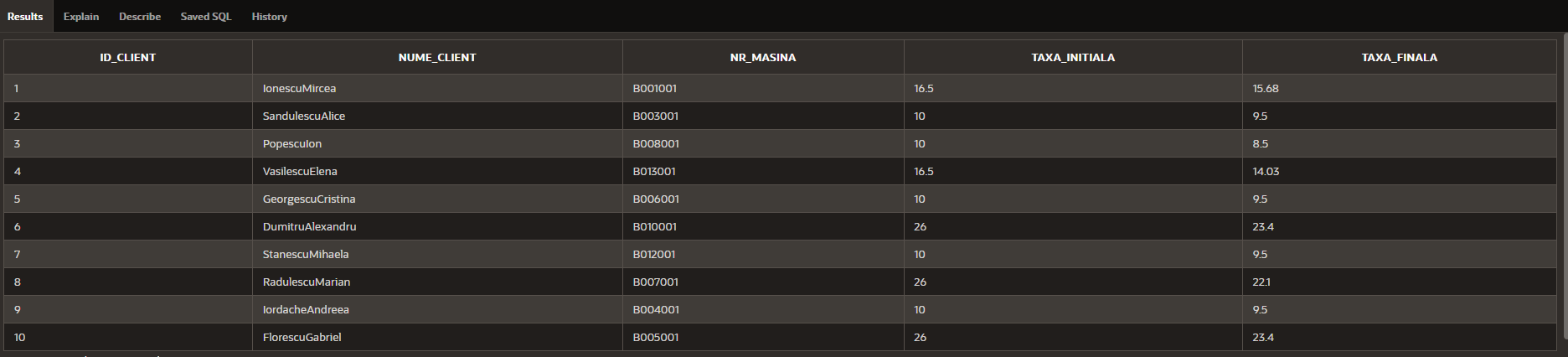
FROM clienti\_soferi c,contracte co,masini m,tarife t

WHERE c.id\_client = co.id\_client AND

co.nr\_masina = m.nr\_masina AND

m.clasa = t.clasa

ORDER BY id\_client;



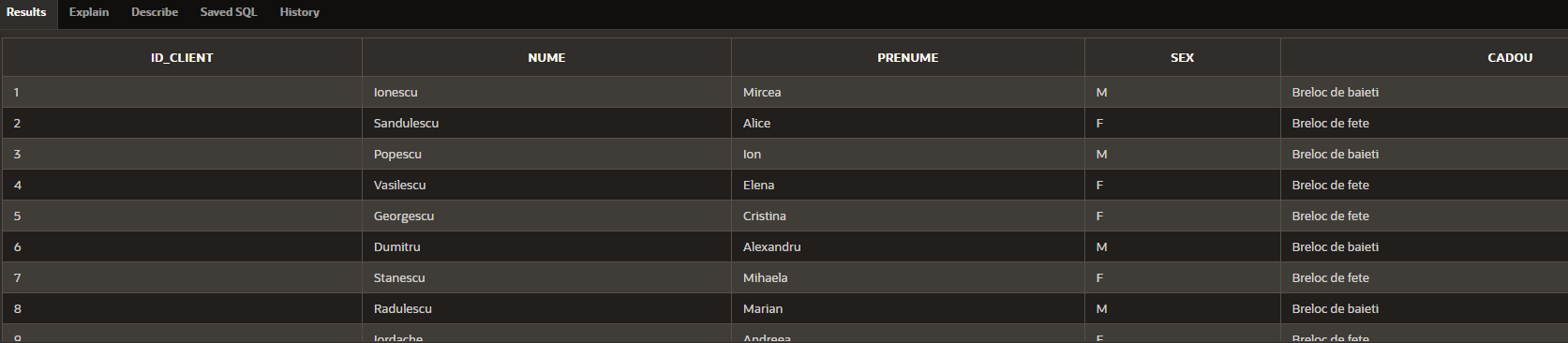
18/ Sa se repartizeze cadourile oferite clientilor la prima inchiriere pe sexe

* M: breloc de baieti
* F: breloc de fete
* NULL: odorizant de masina

SELECT id\_client,nume,prenume,sex,DECODE(lower(sex),'m','Breloc de baieti','f','Breloc de fete','Odorizant de masina') AS cadou

FROM clienti\_soferi

ORDER BY id\_client;



19/ Sa se afiseze date despre service, numele mecanicului si piesele utilizate (folosind functia UNION)

SELECT s.nr\_service, m.nume AS mecanic, p.denumire AS piesa, ps.cantitate

FROM SERVICE s

JOIN MECANICI m ON s.id\_mecanic = m.id\_mecanic

JOIN PIESE\_SERVICE ps ON s.nr\_service = ps.nr\_service

JOIN PIESE p ON ps.id\_piesa = p.id\_piesa

UNION

SELECT s.nr\_service, NULL, NULL, NULL

FROM SERVICE s

WHERE s.nr\_service NOT IN (SELECT nr\_service FROM PIESE\_SERVICE)

ORDER BY nr\_service;



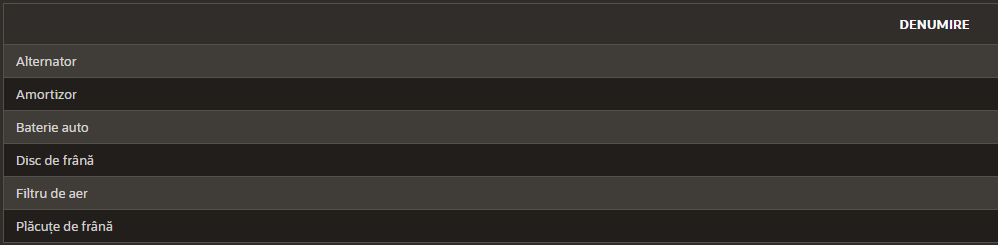
20/ Sa se afiseze toate piesele care au fost utilizate în service cu cantitati <=1 (folosind MINUS)

SELECT denumire FROM PIESE

MINUS

SELECT p.denumire FROM PIESE p, PIESE\_SERVICE ps

WHERE p.id\_piesa = ps.id\_piesa AND ps.cantitate > 1;



21/ Sa se afiseze numele comune dintre mecanici si clienti

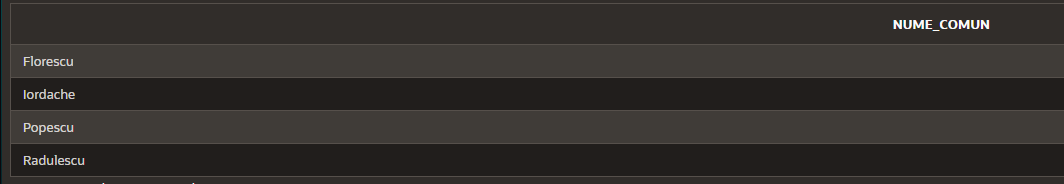
SELECT nume AS nume\_comun

FROM MECANICI

INTERSECT

SELECT nume

FROM CLIENTI\_SOFERI;

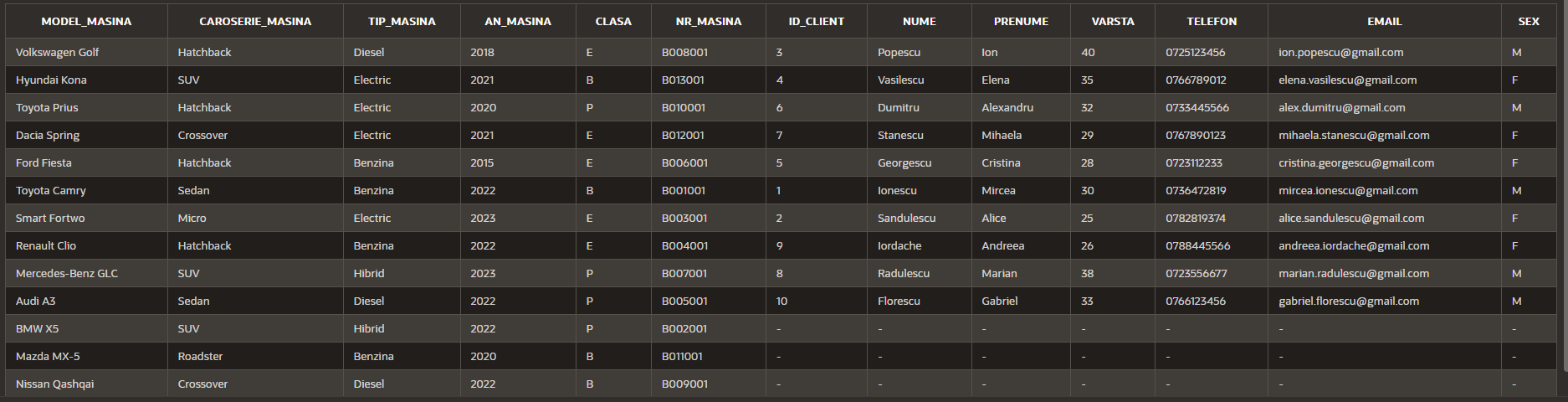


22/ Sa se afiseze toate masinile si clientii care le folosesc, indiferent daca masinile sunt utilizate sau nu

SELECT m.\*,c.\* FROM masini m

LEFT JOIN contracte co ON co.nr\_masina=m.nr\_masina

LEFT JOIN clienti\_soferi c ON c.id\_client=co.id\_client



23/ Sa se creeze o tabela view care afiseaza toti clientii care au inchiriat masini in luna decembrie (doar acces de citire)

CREATE OR REPLACE VIEW INCHIRIERI\_DECEMBRIE AS

SELECT cs.id\_client, cs.nume AS client\_nume, cs.prenume AS client\_prenume, m.nr\_masina, m.model\_masina, c.data\_incepere, c.data\_incheiere

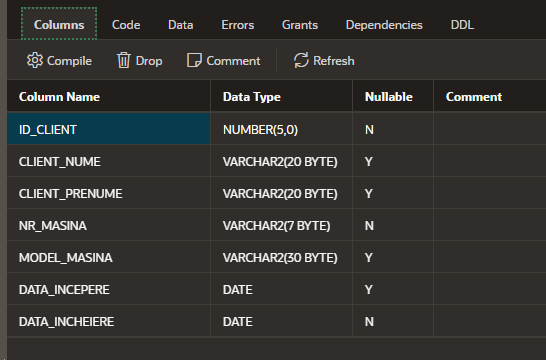
FROM CONTRACTE c

JOIN CLIENTI\_SOFERI cs ON c.id\_client = cs.id\_client

JOIN MASINI m ON c.nr\_masina = m.nr\_masina

WHERE EXTRACT(MONTH FROM c.data\_incepere) = 12

WITH READ ONLY;

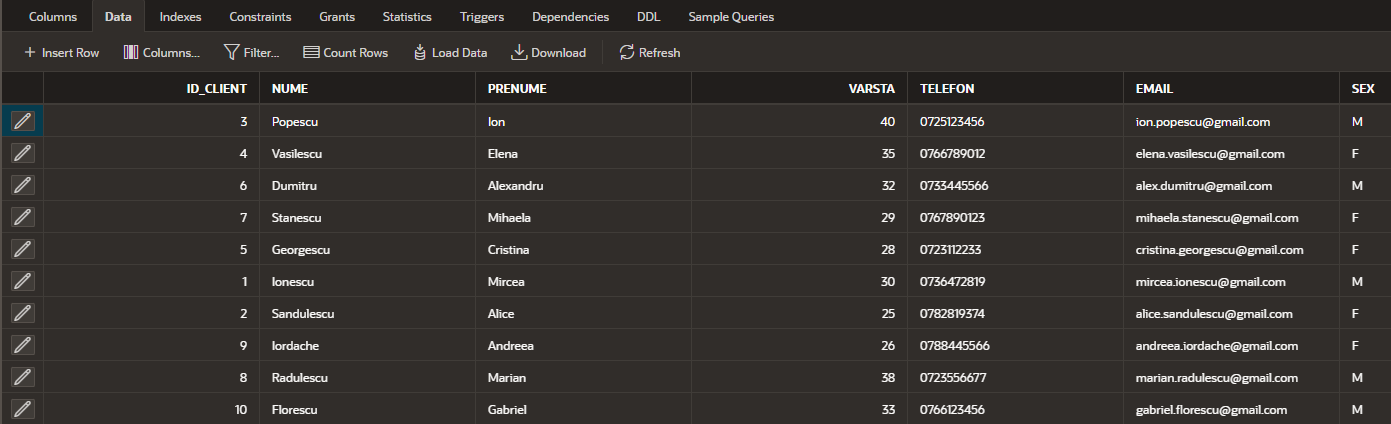


SELECT \* FROM INCHIRIERI\_DECEMBRIE



24/ Sa se creeze o noua tabela(copie a tabelei clienti). Apoi, sa se adauge un client nou utilizand secvente (pentru unicitatea cheii primare). Sa se afiseze in dual valoarea ce urmeaza sa fie implementata a secventei si sa se stearga atat tabela cat si secventa.

CREATE TABLE clienti\_soferi\_test AS SELECT \* FROM clienti\_soferi;



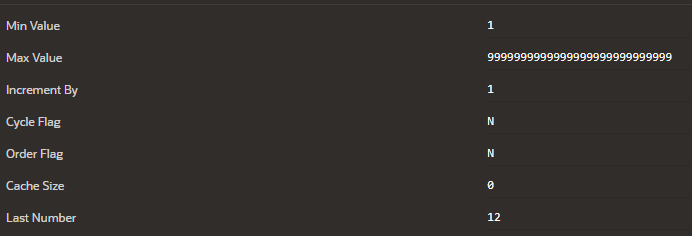
SELECT COUNT(\*) FROM clienti\_soferi\_test

CREATE SEQUENCE clienti\_soferi\_id\_seq

INCREMENT BY 1

START WITH 11

NOMAXVALUE NOCACHE NOCYCLE;

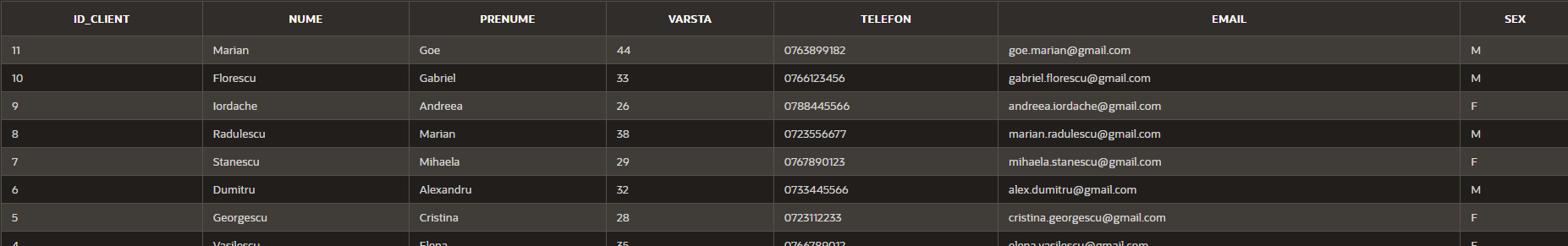


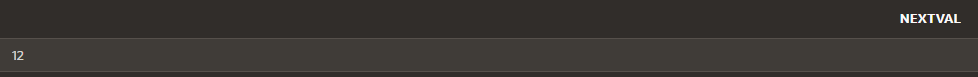
INSERT INTO clienti\_soferi\_test

VALUES(clienti\_soferi\_id\_seq.NEXTVAL,'Marian','Goe',44,'0763899182','goe.marian@gmail.com','M');

SELECT \* FROM clienti\_soferi\_test

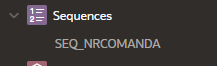
ORDER BY ID\_CLIENT DESC;



SELECT clienti\_soferi\_id\_seq.NEXTVAL FROM DUAL;

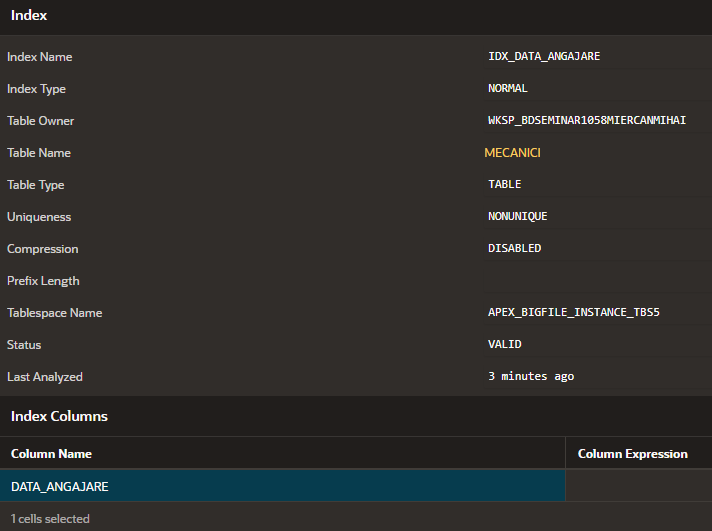
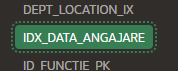
DROP SEQUENCE clienti\_soferi\_id\_seq;

DROP TABLE clienti\_soferi\_test;



25/ Sa se creeze index pentru data angajarii a mecanicilor si sa se afiseze. In final, sa se stearga indexul

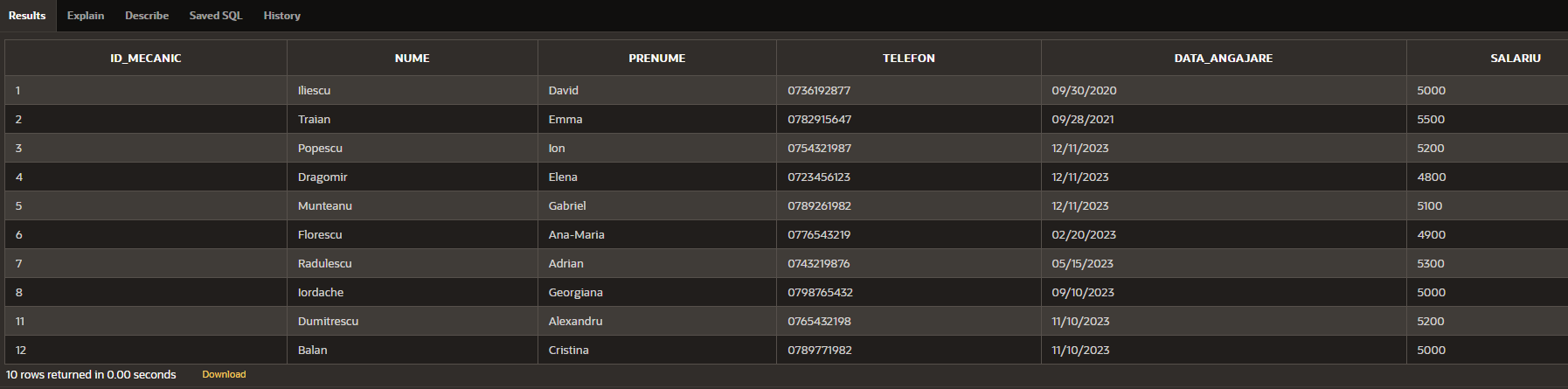
CREATE INDEX idx\_data\_angajare ON MECANICI(data\_angajare);



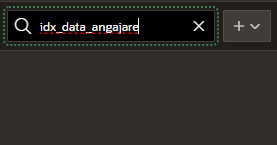
SELECT \* FROM MECANICI

WHERE data\_angajare < TO\_DATE('22-12-2023', 'DD-MM-YYYY')

ORDER BY ID\_MECANIC;



DROP INDEX idx\_data\_angajare;



26/ Sa se creeze un sinonim pentru tabela clienti\_soferi. Sa se afiseze datele din tabel utilizand sinonimul

CREATE SYNONYM cs FOR clienti\_soferi;

SELECT \* FROM cs;

