ACADEMIA DE STUDII ECONOMICE DIN BUCURESTI

FACULTATEA DE CIBERNETICA, STATISTICA SI INFORMATICA ECONOMICA

SPECIALIZAREA INFORMATICA ECONOMICA



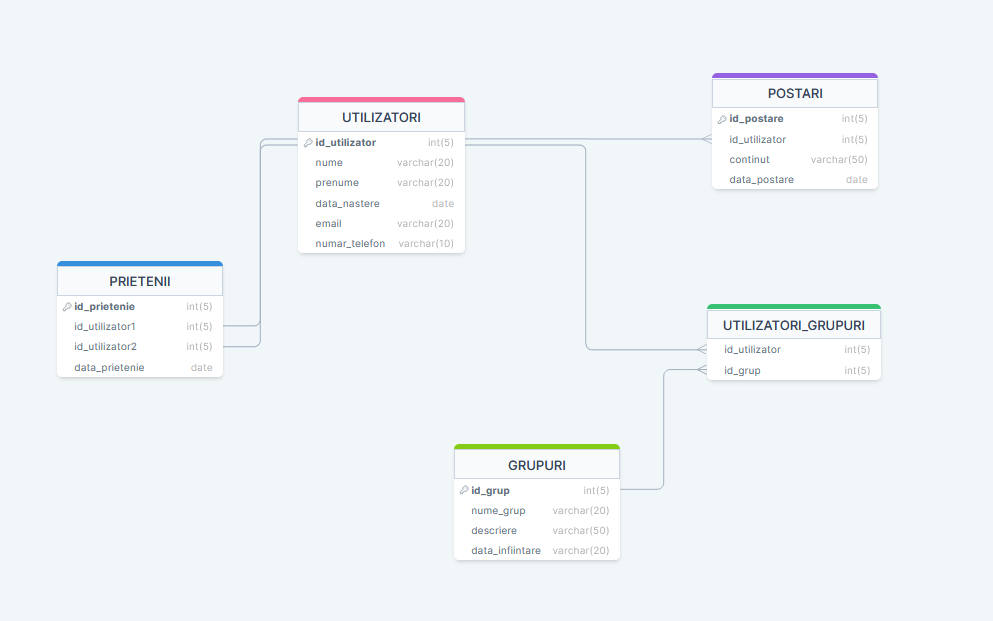
**PROIECT BAZE DE DATE**

**IMPLEMENTAREA UNEI BAZE DE DATE PENTRU O REȚEA DE SOCIALIZARE**

|  |  |
| --- | --- |
| PROFESOR COORDONATOR: | STUDENT: |
| DRD. ENE GABRIELA | MĂGUREANU ȘTEFAN |
|  |  |

Bucuresti

2024



***\*Observatie:*** Pentru schema de mai sus, *int = number* în Oracle SQL.

**Scurtă prezentare a proiectului**

Scopul acestui model de bază de date este acela de a stoca informațiile necesare pentru gestionarea unei rețele de socializare. Aceasta include detalii despre utilizatori, postări, prietenii și grupuri.

**Descrierea tabelei "Utilizatori":**

* **id\_utilizator (cheie primară):** Identificator unic pentru fiecare utilizator.
* **nume:** Numele utilizatorului.
* **prenume:** Prenumele utilizatorului.
* **data\_nastere:** Ziua de naștere a utilizatorului.
* **email:** Adresa de email a utilizatorului.
* **numar\_telefon:** Numărul de telefon al utilizatorului.

**Descrierea tabelei "Postări":**

* **id\_postare (cheie primară):** Identificator unic pentru fiecare postare.
* **id\_utilizator (cheie externă):** Legătura către tabela "Utilizatori" pentru identificarea autorului postării.
* **conținut:** Textul postării.
* **data\_postare:** Data postării.

**Descrierea tabelei "Prietenii":**

* **id\_prietenie (cheie primară):** Identificator unic pentru fiecare relație de prietenie.
* **id\_utilizator1 (cheie externă):** Legătura către tabela "Utilizatori" pentru primul utilizator în relație.
* **id\_utilizator2 (cheie externă):** Legătura către tabela "Utilizatori" pentru al doilea utilizator în relație.
* **data\_prietenie:** Data la care au devenit prieteni.

**Descrierea tabelei "Grupuri":**

* **id\_grup (cheie primară):** Identificator unic pentru fiecare grup.
* **nume\_grup:** Numele grupului.
* **descriere:** Descrierea grupului.
* **data\_infiintare:** Data când s-a creat grupul.

**Descrierea tabelei "UtilizatoriGrupuri":** (Această tabelă face posibilă existența relației many-to-many între **tabela** **utilizatori** și **tabela** **grupuri**)

* **id\_utilizator (cheie externă):** Legătura către tabela "Utilizatori" pentru identificarea utilizatorului în grup.
* **id\_grup (cheie externă):** Legătura către tabela "Grupuri" pentru identificarea grupului.

**CREEAREA TABELELOR**

CREATE TABLE Utilizatori (

id\_utilizator NUMBER CONSTRAINT pk\_utilizatori PRIMARY KEY,

nume VARCHAR2(50),

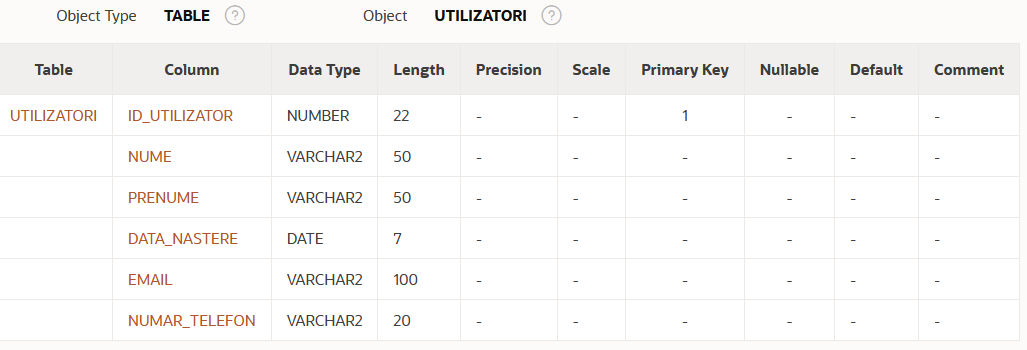
prenume VARCHAR2(50),

data\_nastere DATE,

email VARCHAR2(100),

numar\_telefon VARCHAR2(20)

);



CREATE TABLE Postari (

id\_postare NUMBER CONSTRAINT pk\_postari PRIMARY KEY,

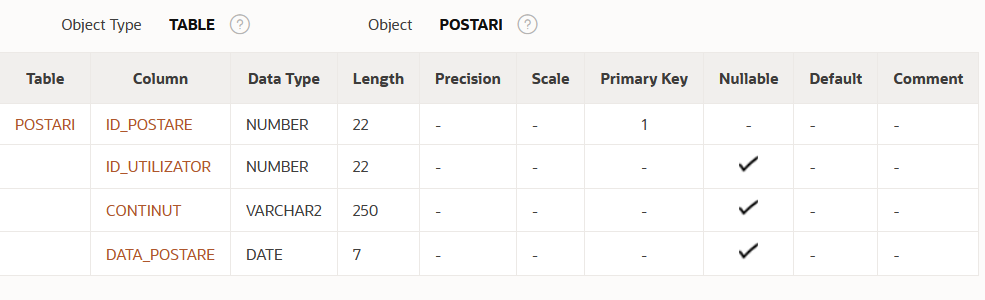
id\_utilizator NUMBER,

continut VARCHAR2(250),

data\_postare DATE,

CONSTRAINT fk\_postari\_utilizatori FOREIGN KEY (id\_utilizator) REFERENCES Utilizatori(id\_utilizator)

);



CREATE TABLE Prietenii (

id\_prietenie NUMBER CONSTRAINT pk\_prietenii PRIMARY KEY,

id\_utilizator1 NUMBER,

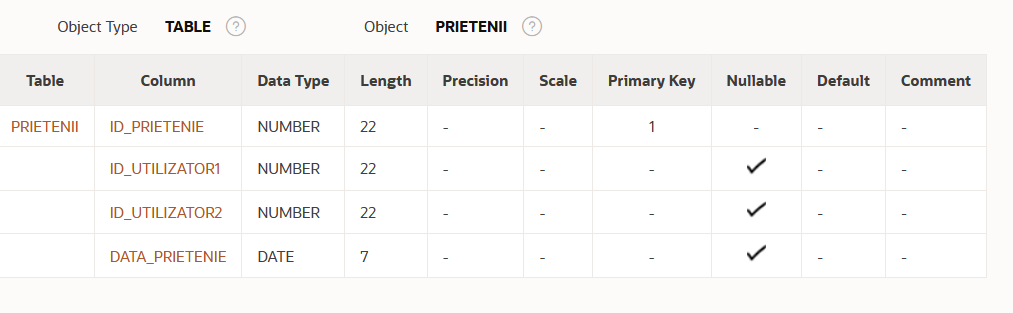
id\_utilizator2 NUMBER,

data\_prietenie DATE,

CONSTRAINT fk\_prietenii\_utilizatori1 FOREIGN KEY (id\_utilizator1) REFERENCES Utilizatori(id\_utilizator),

CONSTRAINT fk\_prietenii\_utilizatori2 FOREIGN KEY (id\_utilizator2) REFERENCES Utilizatori(id\_utilizator)

);



CREATE TABLE Grupuri (

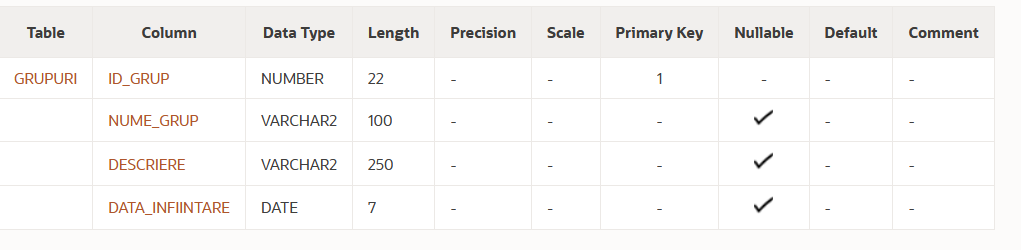
id\_grup NUMBER CONSTRAINT pk\_grupuri PRIMARY KEY,

nume\_grup VARCHAR2(100),

descriere VARCHAR2(250),

data\_infiintare DATE

);



CREATE TABLE UtilizatoriGrupuri (

id\_utilizator NUMBER,

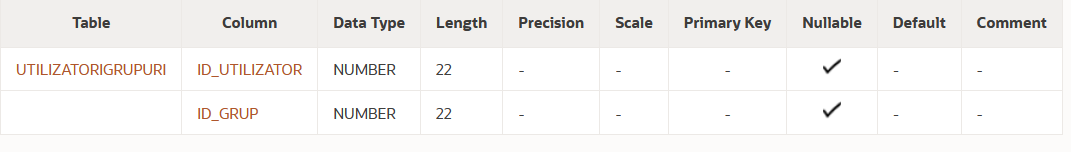
id\_grup NUMBER,

CONSTRAINT fk\_utilizatori\_grupuri\_utilizatori FOREIGN KEY (id\_utilizator) REFERENCES Utilizatori(id\_utilizator),

CONSTRAINT fk\_utilizatori\_grupuri\_grupuri FOREIGN KEY (id\_grup) REFERENCES Grupuri(id\_grup),

CONSTRAINT uq\_utilizatori\_grupuri UNIQUE (id\_utilizator, id\_grup) -- sa nu existe duplicate

);



**INSERAREA DATELOR IN TABELE**

SELECT \* FROM Utilizatori;

SELECT \* FROM Postari;

SELECT \* FROM Prietenii;

SELECT \* FROM Grupuri;

SELECT \* FROM UtilizatoriGrupuri;

**-- Utilizatori:**

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (100, 'Măgureanu', 'Ștefan', TO\_DATE('20-02-2003', 'DD-MM-YYYY'), 'stefan\_magu@mail.com', '0731226579');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (101, 'Miercan', 'Mihai', TO\_DATE('23-08-2003', 'DD-MM-YYYY'), 'miercan\_mihai@mail.com', '0725749203');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (102, 'Miclea', 'Mario', TO\_DATE('02-08-2003', 'DD-MM-YYYY'), 'miclea\_mario@mail.com', '0723754039');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (103, 'Marin', 'Radu', TO\_DATE('04-03-2003', 'DD-MM-YYYY'), 'marin\_radu@mail.com', '0742573943');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (104, 'Mărgelatu', 'Andrei', TO\_DATE('15-07-2003', 'DD-MM-YYYY'), 'margelatu\_andrei@mail.com', '0762839580');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (105, 'Marmureanu', 'Vlad', TO\_DATE('30-04-2003', 'DD-MM-YYYY'), 'marmureanu\_vlad@mail.com', '0725589237');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (106, 'Matei', 'Tudor', TO\_DATE('03-08-2003', 'DD-MM-YYYY'), 'matei\_tudor@mail.com', '0735564759');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (107, 'Leancă', 'Nely', TO\_DATE('27-05-2003', 'DD-MM-YYYY'), 'leanca\_nely@mail.com', '0728885620');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (108, 'Matei', 'Bianca', TO\_DATE('01-10-2003', 'DD-MM-YYYY'), 'matei\_bianca@mail.com', '0722647702');

INSERT INTO Utilizatori (id\_utilizator, nume, prenume, data\_nastere, email, numar\_telefon)

VALUES (109, 'Mărgărit', 'Andrada', TO\_DATE('09-11-2003', 'DD-MM-YYYY'), 'margarit\_andrada@mail.com', '0732991483');

**-- Postari:**

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (1, 100, 'Sunt Ștefan și sunt student la CSIE', TO\_DATE('18-12-2023', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (2, 101, 'Nu îmi place micro-economia', TO\_DATE('12-12-2023', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (3, 102, 'Abia aștept să apară GTA VI', TO\_DATE('14-12-2023', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (4, 103, 'Haideți la sală băieți', TO\_DATE('23-11-2023', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (5, 102, 'Anul ăsta o sa fie cel mai mișto revelion', TO\_DATE('17-11-2023', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (6, 109, 'Heei sunt Andrada și am fost plecată în Erasmus', TO\_DATE('02-01-2024', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (7, 107, 'La mulți ani tuturor!', TO\_DATE('01-01-2024', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (8, 107, 'Ce părere aveți despre noua rețea de socializare?', TO\_DATE('03-01-2024', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (9, 100, 'Abia aștept să vină vacanța!', TO\_DATE('23-11-2023', 'DD-MM-YYYY'));

INSERT INTO Postari (id\_postare, id\_utilizator, continut, data\_postare)

VALUES (10, 101, 'Cineva cu chef de vorbă?', TO\_DATE('28-12-2023', 'DD-MM-YYYY'));

**--Prietenii:**

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (1, 100, 101, TO\_DATE('18-12-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (2, 101, 102, TO\_DATE('14-12-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (3, 100, 104, TO\_DATE('08-10-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (4, 104, 103, TO\_DATE('18-12-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (5, 102, 103, TO\_DATE('01-12-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (6, 100, 109, TO\_DATE('23-12-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (7, 100, 108, TO\_DATE('25-11-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (8, 107, 109, TO\_DATE('01-01-2024', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (9, 104, 109, TO\_DATE('05-11-2023', 'DD-MM-YYYY'));

INSERT INTO Prietenii (id\_prietenie, id\_utilizator1, id\_utilizator2, data\_prietenie)

VALUES (10, 105, 106, TO\_DATE('13-11-2023', 'DD-MM-YYYY'));

**-- Grupuri:**

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (1, 'Programatori', 'Grup pentru discuții despre programare', TO\_DATE('01-11-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (2, 'Cursuri Online', 'Grup pentru cei interesati de cursuri online', TO\_DATE('02-11-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (3, 'Hobby-uri', 'Grup pentru pasionații de diverse hobby-uri', TO\_DATE('08-12-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (4, 'Călătorii', 'Grup pentru iubitorii de călătorii', TO\_DATE('10-12-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (5, 'Fitness', 'Grup pentru cei interesati de fitness și sport', TO\_DATE('03-12-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (6, 'Politică', 'Grup pentru discuții despre politică', TO\_DATE('02-11-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (7, 'Întrebări și răspunsuri', 'Grup pentru a pune întrebări sau a afla răspunsuri', TO\_DATE('01-12-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (8, 'Lecturații', 'Grup pentru pasionații de lectură', TO\_DATE('03-12-2024', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (9, 'Kanal D', 'Grupul oficial postului de televiziune Kanal D', TO\_DATE('09-11-2023', 'DD-MM-YYYY'));

INSERT INTO Grupuri (id\_grup, nume\_grup, descriere, data\_infiintare)

VALUES (10, 'ShopVL', 'Grup pentru anunturi despre vânzare-cumpărare diverse', TO\_DATE('23-12-2023', 'DD-MM-YYYY'));

**-- UtilizatoriGrupuri:**

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (100, 1);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (100, 2);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (104, 4);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (101, 1);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (102, 5);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (109, 4);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (109, 3);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (108, 1);

INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (107, 6);

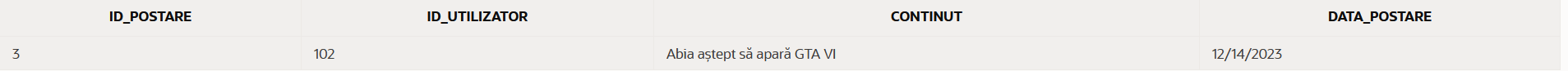
INSERT INTO UtilizatoriGrupuri (id\_utilizator, id\_grup)

VALUES (107, 1);

**Exemplu de actualizare al înregistrărilor**

* 1. Modificați conținutul postării cu id-ul 3.

Înainte de actualizare:

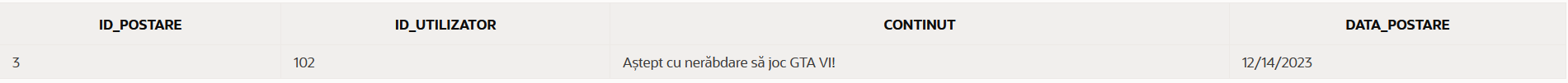
****

UPDATE Postari

SET continut = 'Aștept cu nerăbdare să joc GTA VI!'

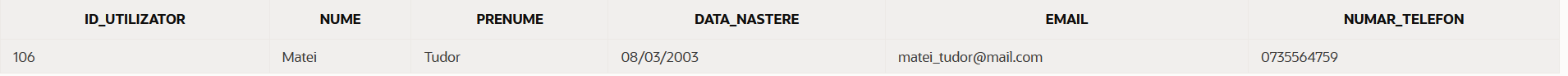
WHERE id\_postare = 3;

După actualizare:



* 1. Modificați numele,prenumele și adresa de mail a utilizatorului cu id-ul 106.

Înainte de actualizare:

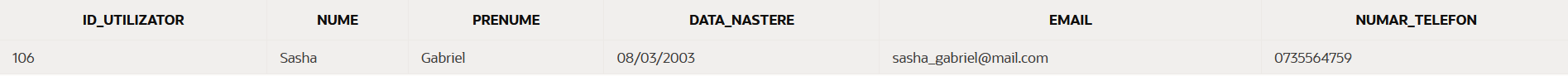


UPDATE Utilizatori

SET nume = 'Sasha', prenume ='Gabriel', email = 'sasha\_gabriel@mail.com'

WHERE id\_utilizator = 106;

După actualizare:



**Exemplu de stergere și recuperare a unei tabele**

Să se creeze o noua tabela denumită CopyUtilizatori(copie a tabelei utilizatori). Să se șteargă noua tabelă, apoi să se recupereze.

CREATE TABLE CopyUtilizatori AS

SELECT \* FROM Utilizatori;

DROP TABLE CopyUtilizatori;

FLASHBACK TABLE CopyUtilizatori TO BEFORE DROP;

**Exemplu de actualizare a structurii tabelelor si modificarea restrictiilor de integritate**

1. Să se adauge restrictia NOT NULL tuturor atributelor tabelei Utilizatori(cu exceptia PK)

ALTER TABLE Utilizatori MODIFY (nume VARCHAR2(50) CONSTRAINT nn\_utilizatori\_nume NOT NULL);

ALTER TABLE Utilizatori MODIFY (prenume VARCHAR2(50) CONSTRAINT nn\_utilizatori\_prenume NOT NULL);

ALTER TABLE Utilizatori MODIFY (data\_nastere DATE CONSTRAINT nn\_utilizatori\_data\_nastere NOT NULL);

ALTER TABLE Utilizatori MODIFY (email VARCHAR2(100) CONSTRAINT nn\_utilizatori\_email NOT NULL);

ALTER TABLE Utilizatori MODIFY (numar\_telefon VARCHAR2(20) CONSTRAINT nn\_utilizatori\_numar\_telefon NOT NULL);

2. Să se adauge în tabela Postări o nouă coloană denumită “nr\_like”, iar apoi să fie ștearsă.

ALTER TABLE Postari

ADD nr\_like NUMBER(5);

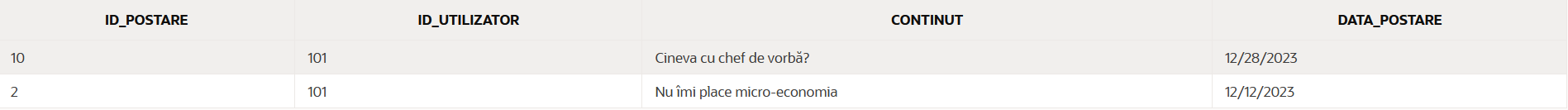
ALTER TABLE Postari

DROP COLUMN nr\_like;

**Exemple de interogări variate**

* 1. Să se afișeze toate postările făcute de utilizatorul cu id-ul 101.

SELECT \* FROM Postari WHERE id\_utilizator = 101;



* 1. Să se afișeze numele și prenumele tuturor utilizatorilor care au postat în grupul cu id-ul 1.

SELECT u.nume, u.prenume

FROM Utilizatori u

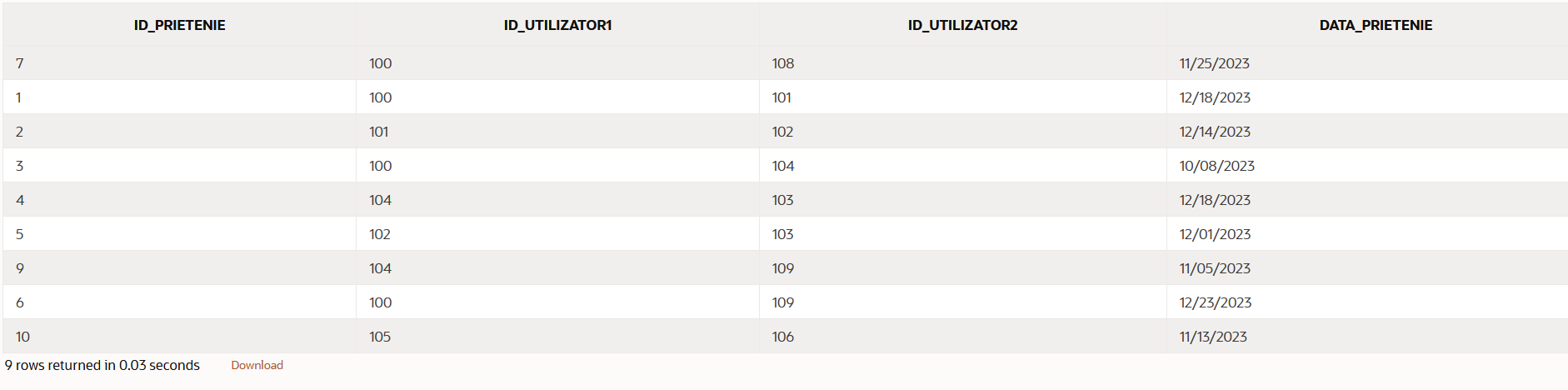
INNER JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator

WHERE ug.id\_grup = 1;



* 1. Să se afișeze toate prieteniile care au fost încheiate înainte de data de 01-01-2024.

SELECT \* FROM Prietenii WHERE data\_prietenie < TO\_DATE('01-01-2024','DD-MM-YYYY');



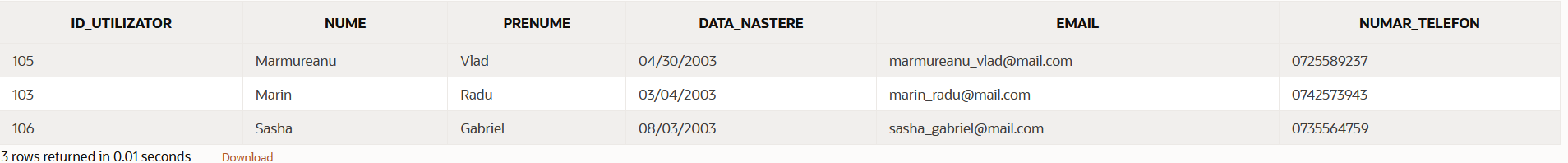
* 1. Să se afișeze utilizatorii care nu sunt membri ai niciunui grup.

SELECT u.\*

FROM Utilizatori u

LEFT JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator

WHERE ug.id\_grup IS NULL;

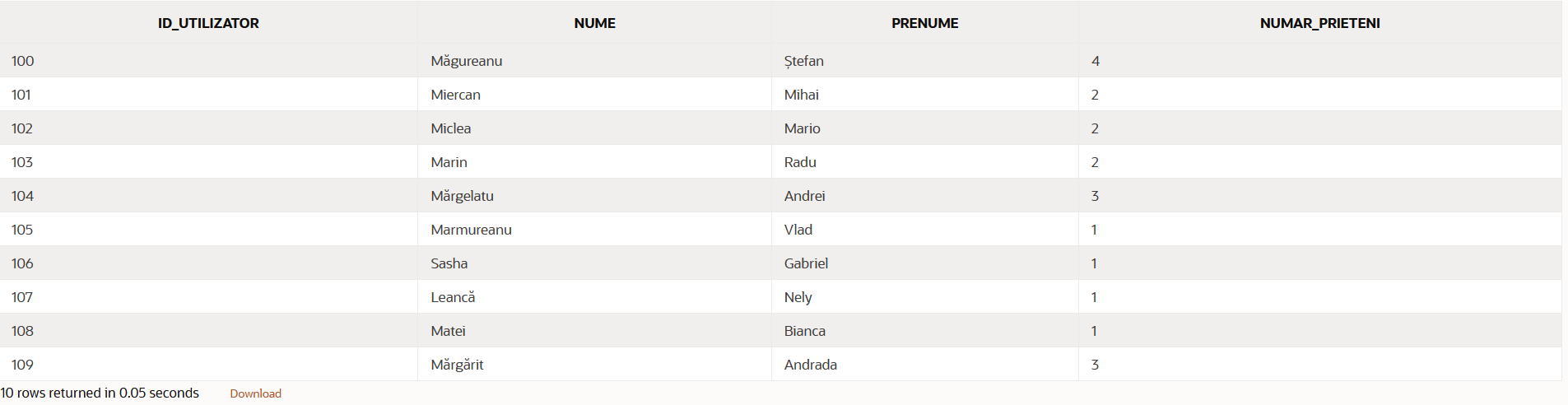


* 1. Să se afișeze numărul total de prieteni ai fiecărui utilizator:

SELECT u.id\_utilizator, u.nume,u.prenume, COUNT(p.id\_utilizator1) AS numar\_prieteni

FROM Utilizatori u, Prietenii p WHERE u.id\_utilizator=p.id\_utilizator1 OR u.id\_utilizator = p.id\_utilizator2

GROUP BY u.id\_utilizator, u.nume, u.prenume

ORDER BY u.id\_utilizator;

* 1. Să se afișeze numele grupurilor și numărul de membri pentru grupurile cu mai mult de 2 membri.

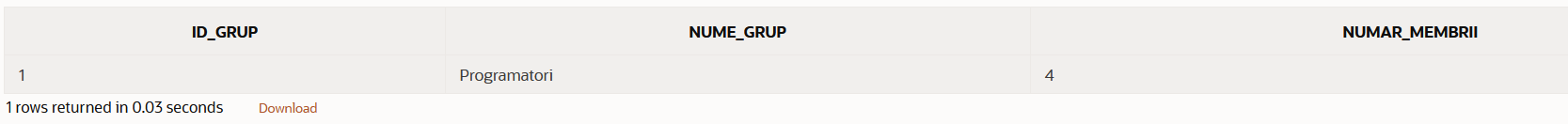
SELECT g.id\_grup,g.nume\_grup, COUNT(ug.id\_utilizator) AS numar\_membrii

FROM grupuri g INNER JOIN UtilizatoriGrupuri ug ON g.id\_grup = ug.id\_grup

GROUP BY g.id\_grup,g.nume\_grup

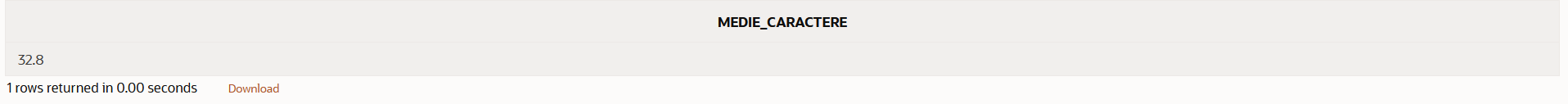
HAVING COUNT(ug.id\_utilizator)>2

ORDER BY g.id\_grup;

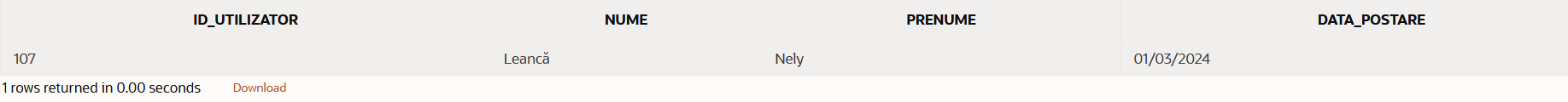


* 1. Afișați numărul mediu de caractere din continutul postărilor.

SELECT AVG(LENGTH(continut)) AS medie\_caractere FROM postari;



* 1. Afișează utilizatorul care a postat cel mai recent.

SELECT u.id\_utilizator,u.nume,u.prenume,p.data\_postare FROM utilizatori u, postari p WHERE u.id\_utilizator = p.id\_utilizator ORDER BY data\_postare DESC FETCH FIRST 1 ROW ONLY;

* 1. Afișează toate postările făcute de prietenii utilizatorului cu id-ul 100.

SELECT P.\*

FROM Postari P

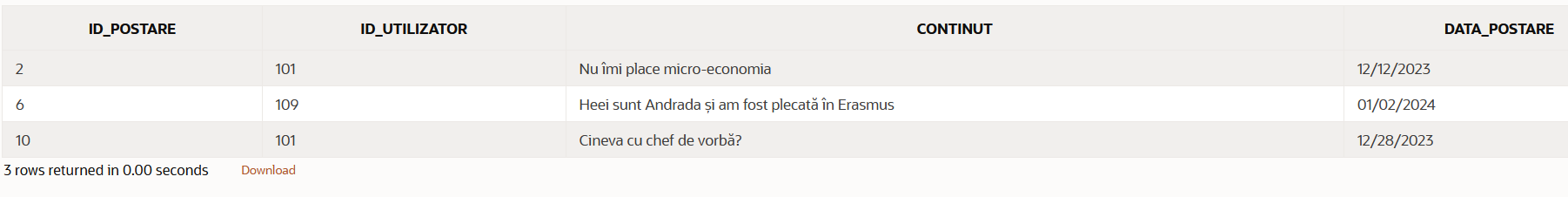
INNER JOIN Prietenii Pr ON P.id\_utilizator = Pr.id\_utilizator2 AND Pr.id\_utilizator1 = 100

UNION

SELECT P.\*

FROM Postari P

INNER JOIN Prietenii Pr ON P.id\_utilizator = Pr.id\_utilizator1 AND Pr.id\_utilizator2 = 100;



* 1. Afișează numele și prenumele utilizatorilor care au postat cel puțin o dată în oricare grup.

SELECT DISTINCT u.id\_utilizator, u.nume, u.prenume

FROM Utilizatori u

INNER JOIN Postari p ON u.id\_utilizator = p.id\_utilizator

INNER JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator;



* 1. Afișează numărul de postări pe zi în ultimele 7 zile.

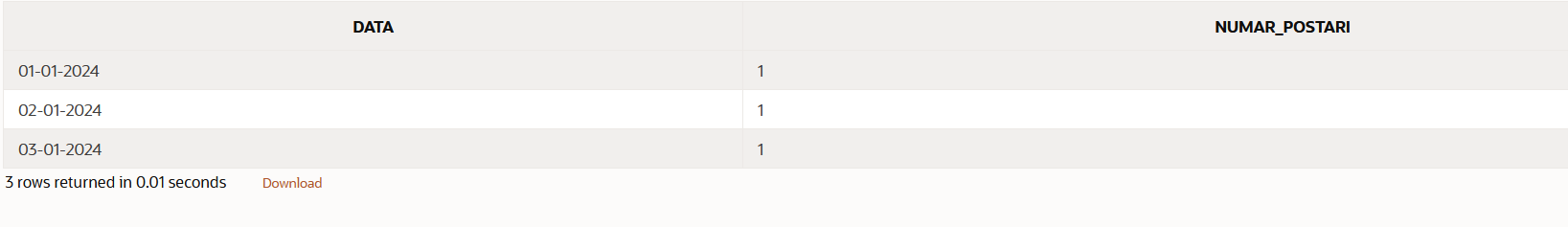
SELECT TO\_CHAR(data\_postare, 'DD-MM-YYYY') AS data, COUNT(\*) AS numar\_postari

FROM Postari

WHERE data\_postare >= SYSDATE - 7

GROUP BY TO\_CHAR(data\_postare, 'DD-MM-YYYY')

ORDER BY TO\_CHAR(data\_postare, 'DD-MM-YYYY');



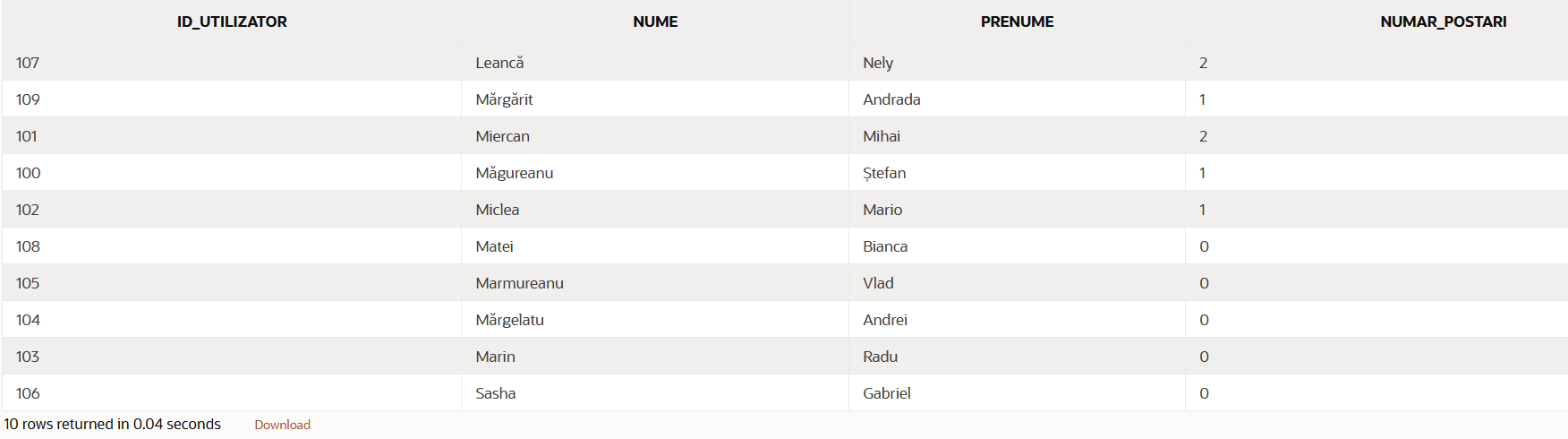
* 1. Afișează numărul total de postări pentru fiecare utilizator în ultimele 30 de zile.

SELECT u.id\_utilizator, u.nume, u.prenume, COUNT(p.id\_postare) AS numar\_postari

FROM Utilizatori u

LEFT JOIN Postari p ON u.id\_utilizator = p.id\_utilizator AND p.data\_postare >= SYSDATE - 30

GROUP BY u.id\_utilizator, u.nume, u.prenume;



* 1. Afișați utilizatorii care au postat în grupuri și au un nume de grup care conține cuvântul 'Online'.

SELECT u.\*

FROM Utilizatori u

INNER JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator

INNER JOIN Grupuri g ON ug.id\_grup = g.id\_grup

WHERE LOWER(g.nume\_grup) LIKE '%online%';



* 1. Afișează toate grupurile în care utilizatorul cu id-ul 107 este membru, dar nu și utilizatorul cu id-ul 106.

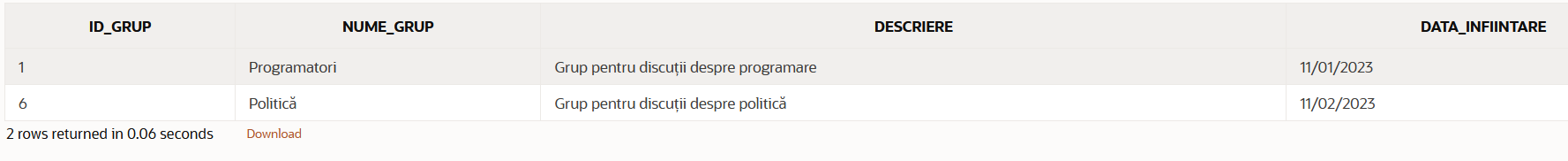
SELECT g.\*

FROM Grupuri g

JOIN UtilizatoriGrupuri ug1 ON g.id\_grup = Ug1.id\_grup AND Ug1.id\_utilizator = 107

LEFT JOIN UtilizatoriGrupuri ug2 ON g.id\_grup = ug2.id\_grup AND ug2.id\_utilizator = 106

WHERE ug2.id\_grup IS NULL;



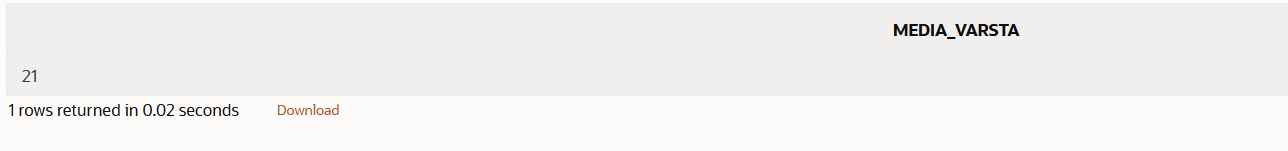
* 1. Afișează media de vârstă a utilizatorilor care au postat în ultima lună.

SELECT AVG(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM u.data\_nastere)) AS media\_varsta

FROM Utilizatori u

INNER JOIN Postari p ON u.id\_utilizator = p.id\_utilizator

WHERE p.data\_postare >= SYSDATE - 30;



* 1. Afișează numele de utilizatori care sunt în grupul "Fitness" și au făcut postări.

SELECT u.nume, u.prenume

FROM Utilizatori u

INNER JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator

INNER JOIN Grupuri g ON ug.id\_grup = g.id\_grup

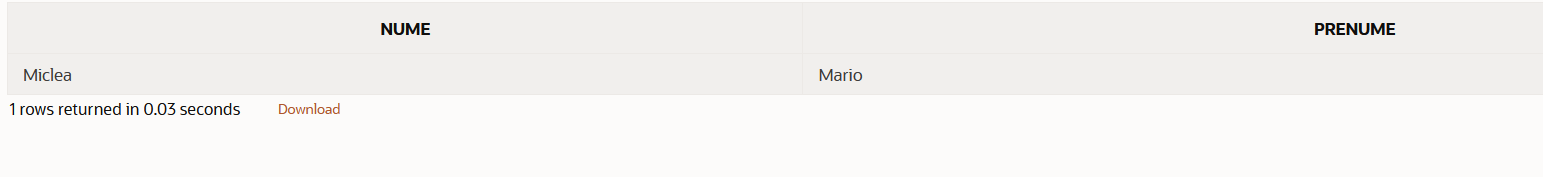
WHERE g.nume\_grup = 'Fitness'

INTERSECT

SELECT u.nume, u.prenume

FROM Utilizatori u

INNER JOIN Postari p ON u.id\_utilizator = p.id\_utilizator;



* 1. Afișează numele utilizatorilor care nu sunt membri în niciun grup.

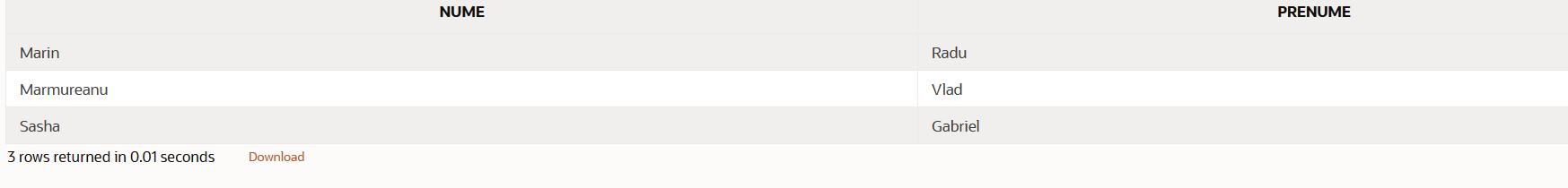
SELECT nume, prenume FROM Utilizatori

MINUS

SELECT u.nume, u.prenume

FROM Utilizatori u

INNER JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator;

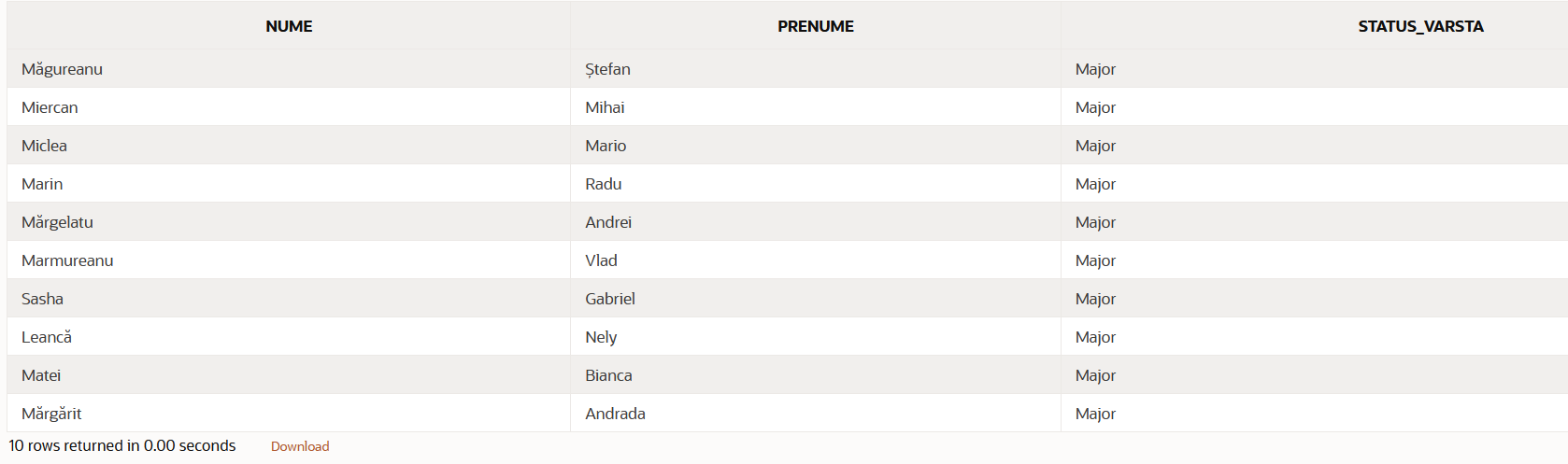


* 1. Afișează statutul utilizatorilor în funcție de vârstă(minor sau major).

SELECT nume, prenume,

CASE WHEN EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM data\_nastere) >= 18 THEN 'Major' ELSE 'Minor' END AS status\_varsta

FROM Utilizatori;



* 1. Afișează numărul total de prieteni pentru fiecare utilizator.

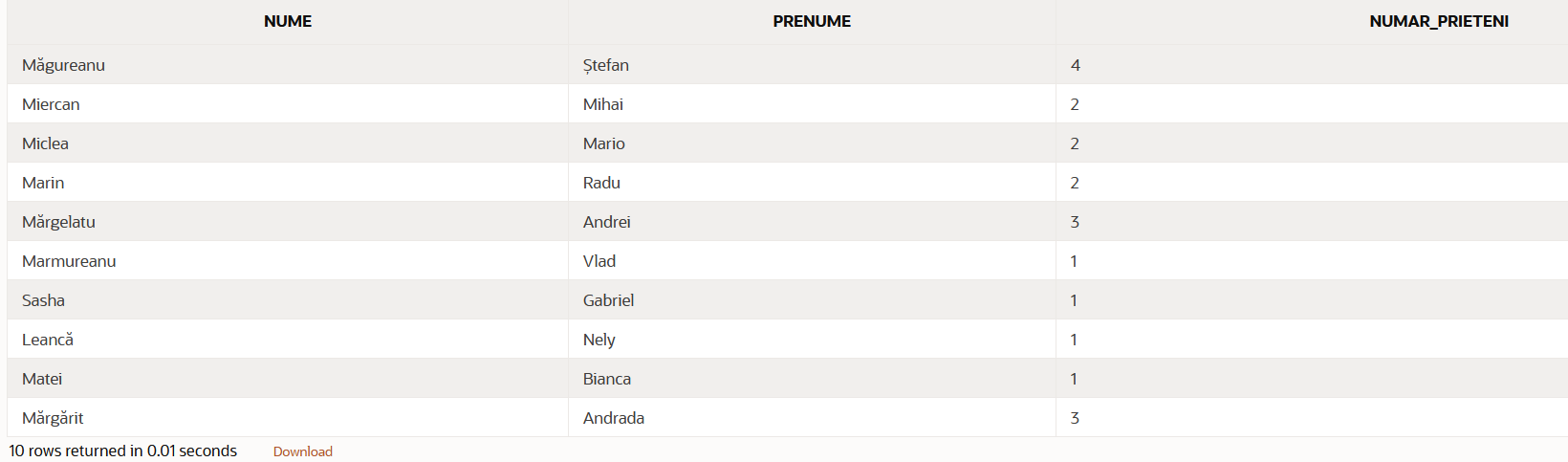
SELECT nume, prenume,

(SELECT COUNT(id\_prietenie)

FROM Prietenii

WHERE id\_utilizator1 = u.id\_utilizator OR id\_utilizator2 = u.id\_utilizator) AS numar\_prieteni

FROM Utilizatori u;



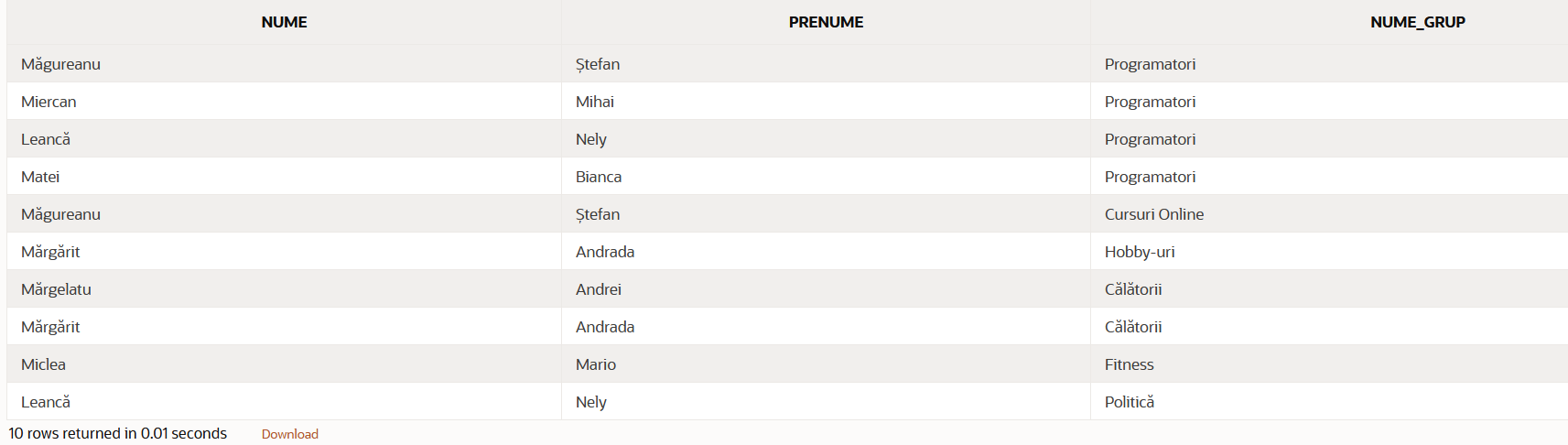
* 1. Afișează numele de utilizatori și numele grupurilor la care sunt membri.

SELECT u.nume, u.prenume, g.nume\_grup

FROM Utilizatori u

INNER JOIN UtilizatoriGrupuri ug ON u.id\_utilizator = ug.id\_utilizator

INNER JOIN Grupuri g ON ug.id\_grup = g.id\_grup;



* 1. Pentru fiecare postare, afisati categoria sa de continut(=0 -> fara continut, =1 ->continut minim, = 250 ->continut maxim, altfel -> continut mediu).

SELECT id\_postare, continut,

DECODE(

LENGTH(continut),

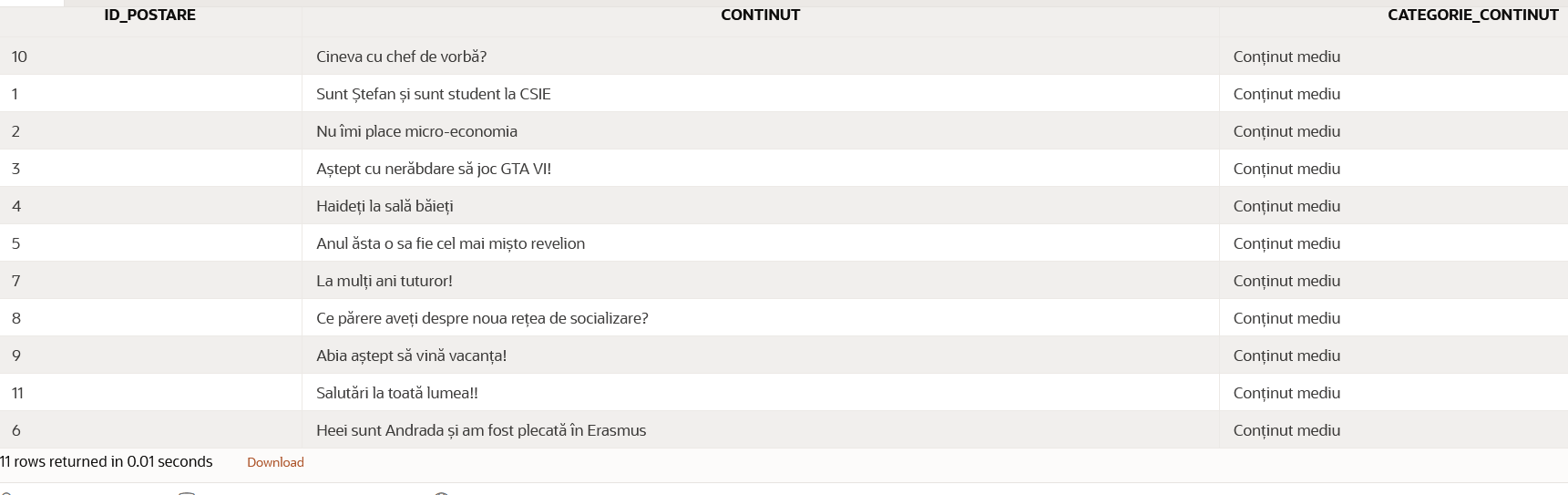
0, 'Fără conținut',

1, 'Conținut minim',

250, 'Conținut maxim',

'Conținut mediu'

) AS categorie\_continut

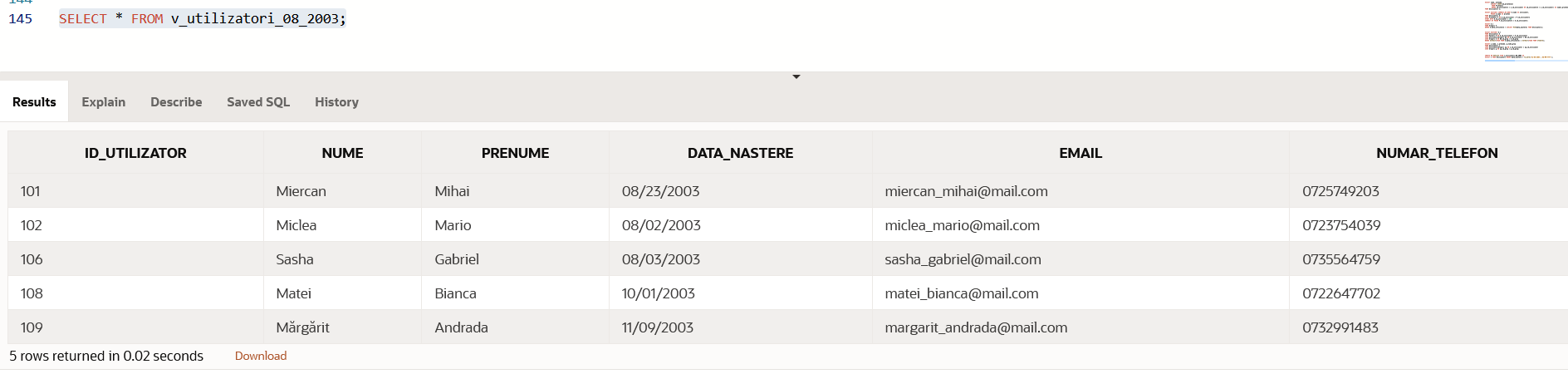
FROM Postari;

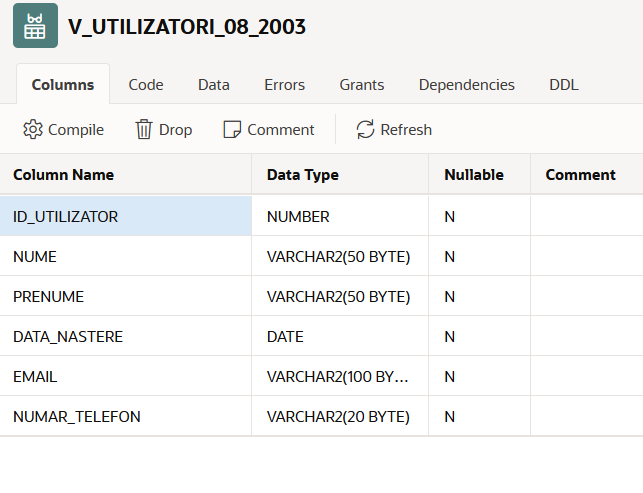
**Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, secvente**

* 1. Sa creeze o tabela virtuala cu toti utilizatorii nascuti dupa August 2003.

CREATE OR REPLACE VIEW v\_utilizatori\_08\_2003 AS

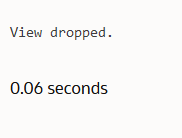
SELECT \* FROM Utilizatori WHERE data\_nastere > TO\_DATE('01-08-2003','DD-MM-YYYY');





* 1. Să se șteargă tabela virutala v\_utilizatori\_08\_2003.

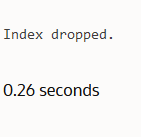
DROP VIEW v\_utilizatori\_08\_2003;



* 1. Să se creeze un index pe coloana "data\_nastere" din tabela "Utilizatori" pentru a optimiza căutările, apoi să se șteargă.

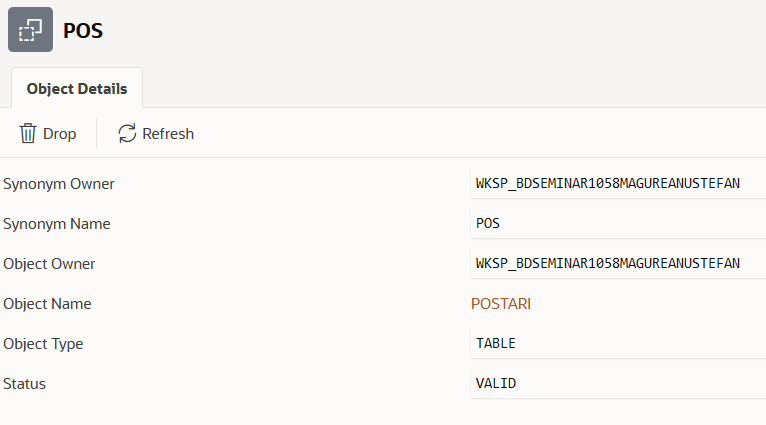
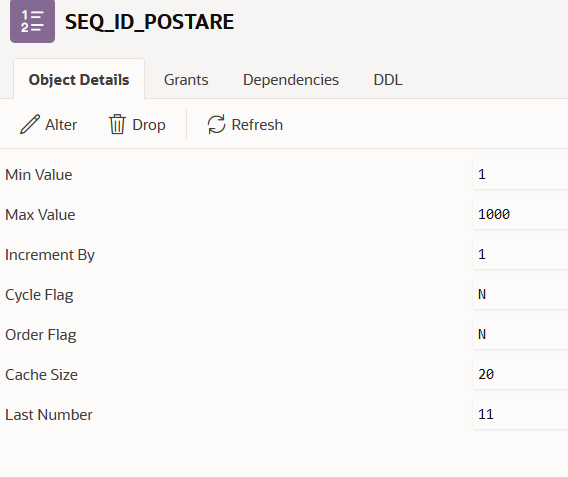
CREATE INDEX idx\_data\_nastere\_utilizatori ON Utilizatori(data\_nastere);

DROP INDEX idx\_data\_nastere\_utilizatori;



* 1. Să se creeze un sinonim pentru tabela "Postari" pentru a simplifica interogările.

CREATE SYNONYM pos FOR Postari;



* 1. Creează o secvență pentru a genera chei primare pentru tabela postări, coloana id\_postare, incepând de a ultimul id(10).

CREATE SEQUENCE seq\_id\_postare

START WITH 11 INCREMENT BY 1

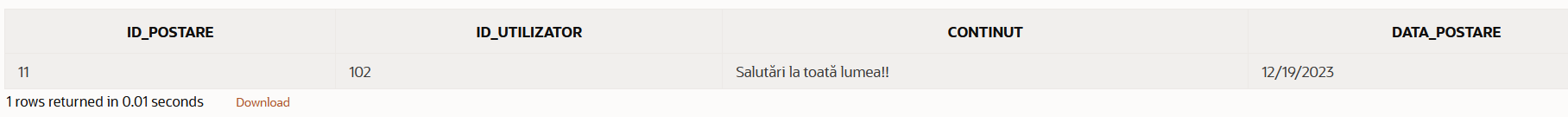
MAXVALUE 1000 NOCYCLE;

* 1. Să se afișeze valoarea curentă a secvenței.

SELECT seq\_id\_postare.CURRVAL FROM dual;

* 1. Să se adauge o inregistrare nouă în tabela postări folosind secventa definita anterior.

INSERT INTO Postari(id\_postare,id\_utilizator,continut,data\_postare)

VALUES (seq\_id\_postare.NEXTVAL,102,'Salutări la toată lumea!!',TO\_DATE('19-12-2023','DD-MM-YYYY'));