

**UNIVERSITATEA DIN BUCUREŞTI**  
**FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ**

**Specializarea: Informatică**

**PROIECT – SISTEME DE GESTIUNE A BAZELOR DE DATE**

**Baza de date pentru gestionarea unei platforme de streaming  
de muzică**

**Realizat de:**

Dorneanu Diana-Delia

Grupa: 241

Anul II

An universitar: 2025 – 2026

Bucureşti

## Contents

1. Prezentarea bazei de date(utilitatea ei) .....	3
2. Diagrama entitate-relație(ERD) .....	5
3. Diagrama conceptuală .....	6
4. Implementarea in Oracle a diagramei conceptuale .....	7
5. Popularea bazei de date.....	10
6. Subprogram stocat independent care să utilizeze toate cele 3 tipuri de colecții studiate .....	19
7. Subprogram stocat <b>independent</b> care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat, <b>dependent de celălalt cursor</b> .....	22
8. Subprogram stocat <b>independent de tip funcție</b> care să utilizeze <b>într-o singură comandă SQL</b> 3 dintre tabelele create. .....	24
9. <b>Subprogram stocat independent de tip procedură</b> care să aibă minim 2 parametri și să utilizeze <b>într-o singură comandă SQL</b> 5 dintre tabelele create .....	26
10. Definiți un <i>trigger</i> de tip LMD la nivel de comandă. Declanșați <i>trigger</i> -ul. ....	29
11. Definiți un <i>trigger</i> de tip LMD la nivel de linie. Declanșați <i>trigger</i> -ul. ....	34
12. Definiți un <i>trigger</i> de tip LDD. Declanșați <i>trigger</i> -ul. ....	44
13. Cerintă optională.....	45

# 1. Prezentarea bazei de date(utilitatea ei)

Baza de date modelează funcționalitatea unei platforme de streaming muzical similară cu Spotify, având ca scop gestionarea utilizatorilor, artiștilor, melodiilor, albumelor, abonamentelor, genurilor muzicale și a caselor de discuri, precum și a interacțiunilor dintre utilizatori și conținutul disponibil în platformă. Este implementată utilizând Oracle Database 21c Express Edition (Release 21.0.0.0.0 – Production), instalat și rulat local pe sistemul de operare Windows 11, fără utilizarea unei mașini virtuale.

## 1. UTILIZATORI

Entitatea stochează informațiile utilizatorilor platformei.

Fiecare utilizator poate avea un singur tip de abonament activ și poate interacționa cu melodiile și artiștii din sistem. Pentru a asculta o melodie, nu este necesară urmărirea artistului.

## 2. ABONAMENTE

Reține tipurile de abonamente disponibile în platformă. „Perioada” semnifică nr de luni cumpărate pentru acel abonament. Prelungirea acestuia se observă în ISTORIC\_ABONAMENTE.

## 3. ARTIȘTI

Reprezintă persoanele sau trupele care creează muzică.

## 4. CASE\_DISCURI

Conține informații despre casele de discuri care reprezintă artiștii.

## 5. ALBUME

Stochează informații despre albumele lansate de artiști. Vom considera că există un singur artist principal, fără colaborări.

## 6. MELODII

Conține detaliile melodiilor disponibile în platformă. O melodie aparține neapărat de un album.

## 7. GENURI\_MUZICALE

Reprezintă clasificările muzicale ale melodiilor.

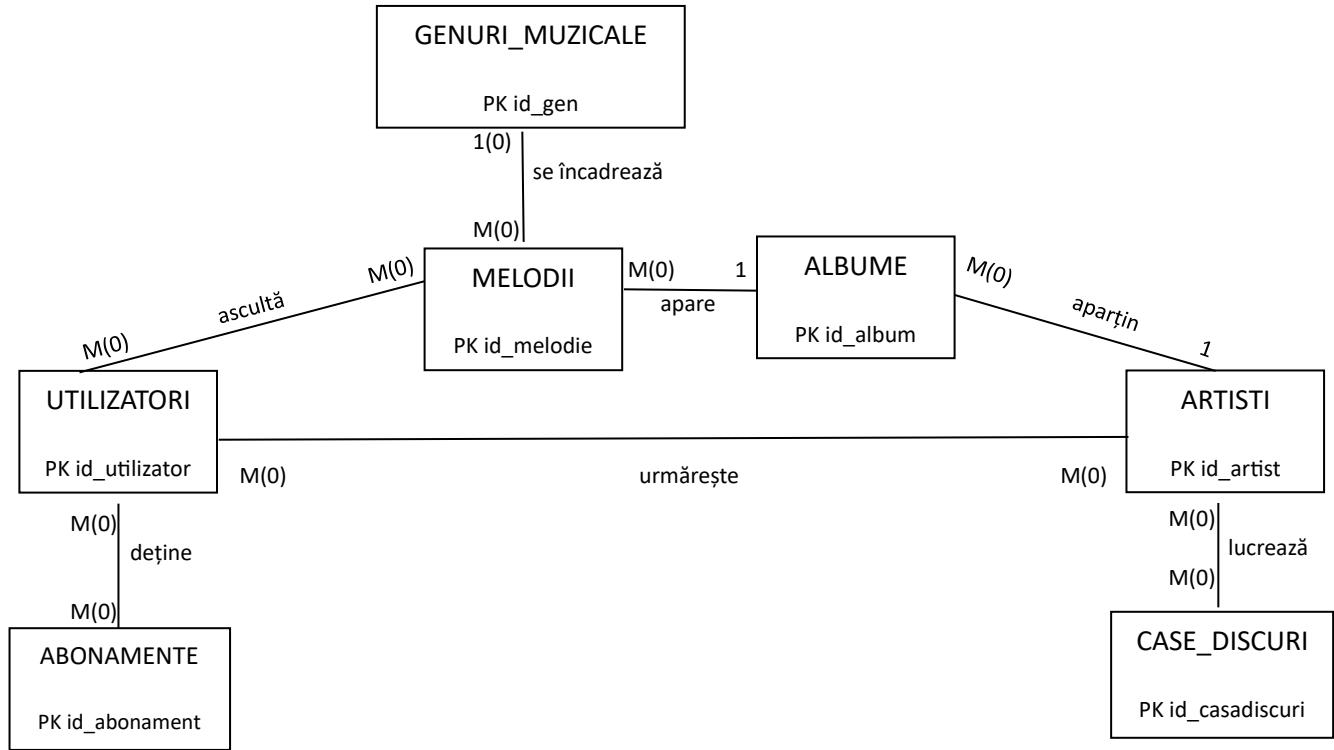
## RELATII

- **ABONAMENTE (M/0) — UTILIZATORI (N/0)**, prin tabela asociativă **ISTORIC\_ABONAMENTE**  
Fiecare utilizator poate avea mai multe abonamente sau niciunul, dar doar unul activ.

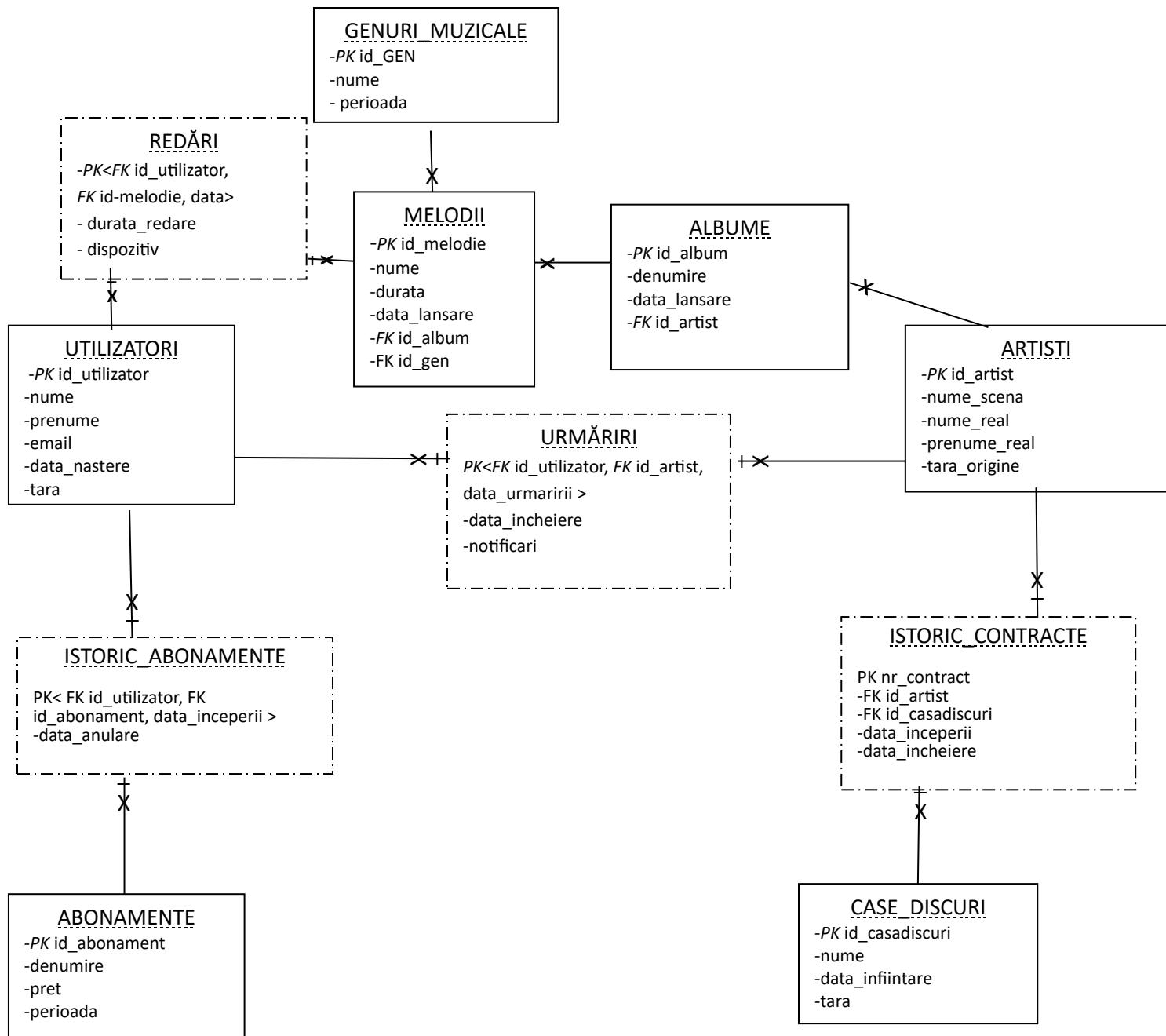
Un tip de abonament poate fi atribuit mai multor utilizatori sau niciunui.

- **UTILIZATORI (M/0) — (N/0) MELODII**, prin tabela asociativă **REDĂRI**  
Un utilizator poate asculta mai multe melodii sau niciuna.  
  
O melodie poate fi ascultată de mai mulți utilizatori sau de niciunul.
- **UTILIZATORI (M/0) — (N/0) ARTISTI**, prin tabela asociativă **URMĂRIRI**  
Un utilizator poate urmări mai mulți artiști sau niciunul.  
  
Un artist poate fi urmărit de mai mulți utilizatori sau de nimeni.
- **CASE\_DISCURSI (M/0) — ARTIȘTI (M/0)**, prin tabela asociativă **ISTORIC\_CONTRACTE**  
Un artist poate fi asociat mai multor case de discuri, sau niciuneia.  
  
O casă de discuri poate reprezenta mai mulți artiști sau niciunul.
- **ARTIȘTI (1/1) — ALBUME (M/0)**  
Un artist poate lansa mai multe albume sau niciunul.  
  
Un album este creat neapărat de un singur artist.
- **ALBUME (1/1) — MELODII (M/0)**  
Un album conține mai multe melodii sau niciuna.  
  
O melodie trebuie să apară într-un singur album.
- **GENURI\_MUZICALE (1/0) — MELODII (N/0)**  
O melodie poate avea un gen muzical sau niciunul.  
Un gen muzical poate corespunde mai multor melodii sau niciuneia.

## 2. Diagrama entitate-relație(ERD)



### 3. Diagrama conceptuală



## 4. Implementarea in Oracle a diagramei conceptuale

```
CREATE TABLE UTILIZATORI (
    ID_UTILIZATOR NUMBER PRIMARY KEY,
    NUME          VARCHAR2(50) NOT NULL,
    PRENUME       VARCHAR2(50) NOT NULL,
    EMAIL         VARCHAR2(50) NOT NULL UNIQUE,
    DATA_NASTERE DATE,
    TARA          VARCHAR2(50)
);

CREATE TABLE ABONAMENTE (
    ID_ABONAMENT NUMBER PRIMARY KEY,
    DENUMIRE      VARCHAR2(50) NOT NULL,
    PRET          NUMBER(6,2) NOT NULL CHECK (PRET >= 0),
    PERIOADA      NUMBER NOT NULL CHECK (PERIOADA > 0)
);

CREATE TABLE ARTISTI (
    ID_ARTIST     NUMBER PRIMARY KEY,
    NUME_SCENA    VARCHAR2(50) NOT NULL,
    NUME_REAL     VARCHAR2(50),
    PRENUME_REAL  VARCHAR2(50),
    TARA_ORIGINE VARCHAR2(50)
);

CREATE TABLE CASE_DISCURI (
    ID_CASADISCURI NUMBER PRIMARY KEY,
    NUME           VARCHAR2(50) NOT NULL,
    DATA_INFIINTARE DATE,
    TARA           VARCHAR2(50)
);

CREATE TABLE GENURI_MUZICALE (
    ID_GEN        NUMBER PRIMARY KEY,
    NUME          VARCHAR2(30) NOT NULL,
    PERIOADA      VARCHAR2(30)
);

CREATE TABLE ALBUME (
    ID_ALBUM      NUMBER PRIMARY KEY,
    DENUMIRE      VARCHAR2(25) NOT NULL,
    DATA_LANSARE  DATE NOT NULL,
    ID_ARTIST    NUMBER NOT NULL,
    CONSTRAINT FK_ALBUM_ARTIST
        FOREIGN KEY (ID_ARTIST)
        REFERENCES ARTISTI(ID_ARTIST)
);

CREATE TABLE MELODII (
    ID_MELODIE    NUMBER PRIMARY KEY,
    NUME          VARCHAR2(20) NOT NULL,
    DURATA        NUMBER CHECK (DURATA > 0),
    DATA_LANSARE  DATE,
    ID_ALBUM      NUMBER NOT NULL,
    ID_GEN        NUMBER,
```

```

CONSTRAINT FK_MELODIE_ALBUM
    FOREIGN KEY (ID_ALBUM)
    REFERENCES ALBUME(ID_ALBUM),
CONSTRAINT FK_MELODIE_GEN
    FOREIGN KEY (ID_GEN)
    REFERENCES GENURI_MUZICALE(ID_GEN)
);

CREATE TABLE ISTORIC_ABONAMENTE (
    ID_UTILIZATOR NUMBER,
    ID_ABONAMENT NUMBER,
    DATA_INCEPERII DATE,
    DATA_ANULARE DATE,
    CONSTRAINT PK_ISTORIC_ABONAMENTE
        PRIMARY KEY (ID_UTILIZATOR, ID_ABONAMENT, DATA_INCEPERII),
    CONSTRAINT FK_IA_UTILIZATOR
        FOREIGN KEY (ID_UTILIZATOR)
        REFERENCES UTILIZATORI(ID_UTILIZATOR),
    CONSTRAINT FK_IA_ABONAMENT
        FOREIGN KEY (ID_ABONAMENT)
        REFERENCES ABONAMENTE(ID_ABONAMENT),
    CONSTRAINT CK_DATE_ABONAMENT
        CHECK (DATA_ANULARE IS NULL OR DATA_ANULARE > DATA_INCEPERII)
);
;

CREATE TABLE REDARI (
    ID_UTILIZATOR NUMBER,
    ID_MELODIE NUMBER,
    DATA_REDARE DATE,
    DURATA_REDARE NUMBER CHECK (DURATA_REDARE > 0),
    DISPOZITIV VARCHAR2(25),
    CONSTRAINT PK_REDARI
        PRIMARY KEY (ID_UTILIZATOR, ID_MELODIE, DATA_REDARE),
    CONSTRAINT FK_RED_UTILIZATOR
        FOREIGN KEY (ID_UTILIZATOR)
        REFERENCES UTILIZATORI(ID_UTILIZATOR),
    CONSTRAINT FK_RED_MELODIE
        FOREIGN KEY (ID_MELODIE)
        REFERENCES MELODII(ID_MELODIE)
);
;

CREATE TABLE URMARIRI (
    ID_UTILIZATOR NUMBER,
    ID_ARTIST NUMBER,
    DATA_URMARIRII DATE,
    DATA_INCHEIERE DATE,
    NOTIFICARI CHAR(1) CHECK (NOTIFICARI IN ('Y', 'N')),
    CONSTRAINT PK_URMARIRI
        PRIMARY KEY (ID_UTILIZATOR, ID_ARTIST, DATA_URMARIRII),
    CONSTRAINT FK_URM_UTILIZATOR
        FOREIGN KEY (ID_UTILIZATOR)
        REFERENCES UTILIZATORI(ID_UTILIZATOR),
    CONSTRAINT FK_URM_ARTIST
        FOREIGN KEY (ID_ARTIST)
        REFERENCES ARTISTI(ID_ARTIST),
    CONSTRAINT CK_DATE_URMARIRE
        CHECK (DATA_INCHEIERE IS NULL OR DATA_INCHEIERE > DATA_URMARIRII)
);
;
```

```

);
CREATE TABLE ISTORIC_CONTRACTE (
    NR_CONTRACT      NUMBER PRIMARY KEY,
    ID_ARTIST        NUMBER NOT NULL,
    ID_CASADISCURI  NUMBER NOT NULL,
    DATA_INCEPERII   DATE NOT NULL,
    DATA_INCHEIERE  DATE,
    CONSTRAINT FK_IC_ARTIST
        FOREIGN KEY (ID_ARTIST)
        REFERENCES ARTISTI(ID_ARTIST),
    CONSTRAINT FK_IC_CASADISCURI
        FOREIGN KEY (ID_CASADISCURI)
        REFERENCES CASE_DISCURI(ID_CASADISCURI),
    CONSTRAINT CK_DATE_CONTRACT
        CHECK (DATA_INCHEIERE IS NULL OR DATA_INCHEIERE > DATA_INCEPERII)
);

```

The screenshot shows the Database Explorer interface with the 'tables' node expanded. A new table named 'ISTORIC\_CONTRACTE' is listed under the 'ABONAMENTE' category. The table structure is identical to the one shown in the code block above. The status bar at the bottom indicates the command was completed in 113 ms.

The screenshot shows the Database Explorer interface with the 'tables' node expanded. A query is being run in the console:

```

CREATE TABLE ISTORIC_CONTRACTE (
);
SELECT TABLE_NAME
FROM USER_TABLES
ORDER BY TABLE_NAME;

```

The output tab shows the results of the query:

TABLE_NAME
1 ABONAMENTE
2 ALBUME
3 ARTISTI
4 CASE_DISCURI
5 GENURI_MUZICALE
6 ISTORIC_ABONAMENTE
7 ISTORIC_CONTRACTE
8 MELODII
9 REDARI
10 URMARIRI
11 UTILIZATORI

The status bar at the bottom indicates the command was completed in 529 ms.

OBSERVATIE: Am ales să pun opțiunea de *on delete cascade* pe următoarele tabele, deoarece:

- **UTILIZATORI**

Utilizatorul trebuie să aibă opțiunea de a-si șterge contul, ca în cazul în care toate datele asociate acestuia sunt eliminate automat din sistem.

- **ALBUME / MELODII**

Albumele și melodiile pot fi eliminate din platformă, de exemplu din motive de copyright, iar ștergerea în cascădă asigură eliminarea tuturor datelor dependente.

Celelalte tabele conțin date istorice sau contractuale care nu trebuie șterse automat, pentru a păstra coherența și istoricul aplicației.

```
ALTER TABLE MELODII DROP CONSTRAINT FK_MELODIE_ALBUM;
ALTER TABLE MELODII
ADD CONSTRAINT FK_MELODIE_ALBUM
    FOREIGN KEY (ID_ALBUM)
        REFERENCES ALBUME(ID_ALBUM)
    ON DELETE CASCADE;

ALTER TABLE ISTORIC_ABONAMENTE DROP CONSTRAINT FK_IA_UTILIZATOR;
ALTER TABLE ISTORIC_ABONAMENTE
ADD CONSTRAINT FK_IA_UTILIZATOR
    FOREIGN KEY (ID_UTILIZATOR)
        REFERENCES UTILIZATORI(ID_UTILIZATOR)
    ON DELETE CASCADE;

ALTER TABLE REDARI DROP CONSTRAINT FK_RED_UTILIZATOR;
ALTER TABLE REDARI
ADD CONSTRAINT FK_RED_UTILIZATOR
    FOREIGN KEY (ID_UTILIZATOR)
        REFERENCES UTILIZATORI(ID_UTILIZATOR)
    ON DELETE CASCADE;

ALTER TABLE REDARI DROP CONSTRAINT FK_RED_MELODIE;
ALTER TABLE REDARI
ADD CONSTRAINT FK_RED_MELODIE
    FOREIGN KEY (ID_MELODIE)
        REFERENCES MELODII(ID_MELODIE)
    ON DELETE CASCADE;

ALTER TABLE URMARIRI DROP CONSTRAINT FK_URM_UTILIZATOR;
ALTER TABLE URMARIRI
ADD CONSTRAINT FK_URM_UTILIZATOR
    FOREIGN KEY (ID_UTILIZATOR)
        REFERENCES UTILIZATORI(ID_UTILIZATOR)
    ON DELETE CASCADE;
```

The screenshot shows a MySQL command-line interface window. At the top, there's a toolbar with various icons. Below it is a text area containing SQL code. The code includes creating a user ('CREATE USER'), setting a password ('SET PASSWORD'), and granting privileges ('GRANT'). Below the code, there's a table definition for 'utilizatori' with columns like ID\_UTILIZATOR, NUME, PRENUME, EMAIL, DATA\_NASTERE, and TARA. A 'SELECT \*' query is run against this table, returning a single row where ID\_UTILIZATOR is 21. The bottom part of the window shows the results of the query in a table format with columns labeled ID\_UTILIZATOR, NUME, PRENUME, DATA\_NASTERE, and TARA.

```
> CREATE USER 'streaming_user'@'localhost' IDENTIFIED BY 'password';
Query OK, 1 row affected (0.00 sec)

> GRANT ALL PRIVILEGES ON *.* TO 'streaming_user'@'localhost';
Query OK, 0 rows affected (0.00 sec)

> 
260 ✓  INSERT INTO UTILIZATORI
261   VALUES ( ID_UTILIZATOR_SEQ_UTILIZATORI.NEXTVAL, 'TEST', 'USER', 'test.user@email.com', DATE '2000-01-01', 'ROMANIA');
262   •
263 ✓  INSERT INTO REDARI
264   VALUES ( ID_UTILIZATOR 21, ID_MELODIE 1, DATA_REDARE SYSDATE, DURATA_REDARE 100, DISPOZITIV 'MOBIL');

265 ✓  DELETE FROM UTILIZATORI
266   WHERE ID_UTILIZATOR = 21;
267
268 ✓  SELECT * FROM UTILIZATORI
269   WHERE ID_UTILIZATOR=21;
270

Output [STREAMING_USER.UTILIZATORI]
ID_UTILIZATOR    NUME    PRENUME    DATA_NASTERE    TARA
21      TEST      USER  2000-01-01  ROMANIA
```

## 5. Popularea bazei de date

```
CREATE SEQUENCE SEQ_UTILIZATORI START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_ABONAMENTE START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_ARTISTI START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_CASE_DISCURI START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_GENURI START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_ALBUME START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_MELODII START WITH 1 INCREMENT BY 1 NOCYCLE;
CREATE SEQUENCE SEQ_CONTRACTE START WITH 1 INCREMENT BY 1 NOCYCLE;
```

The screenshot shows the Oracle SQL Developer interface. In the top navigation bar, there's a 'Version control' dropdown. Below it, the 'Database' sidebar shows the schema structure with nodes like 'Database', 'Tables', 'Sequences', and 'Packages'. The main area is a 'Console' tab titled 'cod.sql' which contains the sequence creation DDL statements. Each statement has a green checkmark next to it, indicating successful execution. The 'Services' sidebar at the bottom shows the database connection and session status. The bottom status bar indicates the session is connected to 'localhost'.

- *Utilizatori*

```
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'WHITE', 'DANIELLE',
  'white.danielle@email.com', DATE '2005-12-10', 'UK');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'CORTEZ', 'ALBERTO',
  'cortez.alberto@email.com', DATE '2008-10-12', 'SPANIA');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'POPESCU', 'ANA',
  'ana.popescu@email.com', DATE '1998-05-10', 'ROMANIA');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'IONESCU', 'MARIA',
  'maria.ionescu@email.com', DATE '1995-11-22', 'ROMANIA');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'SMITH', 'JOHN',
  'john.smith@email.com', DATE '1990-03-15', 'USA');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'DUPONT', 'LUC',
  'luc.dupont@email.com', DATE '1988-07-01', 'FRANTA');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'GARCIA', 'CARLA',
  'carla.garcia@email.com', DATE '2000-09-18', 'SPANIA');
INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'MONEA', 'OANA',
  'oana.monea@email.com', DATE '2002-05-18', 'ROMANIA');
```

```

Database Explorer          Version control
+ [ ] @laborator 2 of 188  console [@localhost] cod.sql
  + [ ] @localhost 2 of 39
    > [ ] PUBLIC
    > [ ] STREAMING_USER
      > [ ] tables 11
        > [ ] ABONAMENTE
        > [ ] ALBUME
      > [ ] UTILIZATORI
      > [ ] console 31 ms
      > [ ] ABONAMENTE

Services
[ ] + [ ] - [ ] x
Tx [ ] Database
  > [ ] @localhost
    > [ ] cod.sql
    < UTILIZATORI
    > [ ] console 31 ms
    < ABONAMENTE

Database Console: > @localhost [cod.sql] console[@localhost]
  
```

```

 1 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'WHITE', PRENUME 'DANIELLE', EMAIL 'white.danielle@email.com', DATA_NASTERE '1961-01-01');
 2 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'CORTEZ', PRENUME 'ALBERTO', EMAIL 'cortez.alberto@email.com', DATA_NASTERE '1998-01-01');
 3 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'POPESCU', PRENUME 'ANA', EMAIL 'ana.popescu@email.com', DATA_NASTERE '1998-01-01');
 4 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'IONESCU', PRENUME 'MARIA', EMAIL 'maria.ionescu@email.com', DATA_NASTERE '1990-03-01');
 5 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'SMITH', PRENUME 'JOHN', EMAIL 'john.smith@email.com', DATA_NASTERE '1990-03-01');
 6 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'DUPONT', PRENUME 'LUC', EMAIL 'luc.dupont@email.com', DATA_NASTERE '1988-01-01');
 7 ✓ INSERT INTO UTILIZATORI VALUES (ID_UTILIZATORI SEQ_UTILIZATORI.NEXTVAL, NUME 'GARCIA', PRENUME 'CARLA', EMAIL 'carla.garcia@email.com', DATA_NASTERE '2000-01-01');

  
```

Output: STREAMING\_USER.UTILIZATORI

```

[2026-01-06 16:02:31] STREAMING_USER> INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'DUPONT','LUC', 'luc.dupont@email.com')
[2026-01-06 16:02:31] 1 row affected in 4 ms
[2026-01-06 16:02:31] STREAMING_USER> INSERT INTO UTILIZATORI VALUES (SEQ_UTILIZATORI.NEXTVAL, 'GARCIA','CARLA', 'carla.garcia@email.com')
[2026-01-06 16:02:31] 1 row affected in 4 ms
  
```

## • ABONAMENTE

```

Database Explorer          Version control
+ [ ] @laborator 2 of 188  console [@localhost] ABONAMENTE [@localhost] cod.sql
  + [ ] @localhost 2 of 39
    > [ ] PUBLIC
    > [ ] STREAMING_USER
      > [ ] tables 11
        > [ ] ABONAMENTE
      > [ ] UTILIZATORI
      > [ ] console 63 ms
      > [ ] ABONAMENTE 404 ms

Services
[ ] + [ ] - [ ] x
Tx [ ] Database
  > [ ] @localhost
    > [ ] cod.sql
    < UTILIZATORI
    > [ ] console 63 ms
    < ABONAMENTE

Database Console: > @localhost [cod.sql] console[@localhost]
  
```

```

 1 ✓ INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'DUO', 27, 1);
 2 ✓ INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'PUPIL', 0, 12);
 3 ✓ INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'STUDENT', 10, 1);
 4 ✓ INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'STANDARD', 20, 1);
 5 ✓ INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'FAMILY', 36, 1);
 6 ✓ INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'ANUAL', 200, 12);

  
```

Output: STREAMING\_USER.UTILIZATORI

```

[2026-01-06 14:44:28] STREAMING_USER> INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'STANDARD', 20, 1)
[2026-01-06 14:44:28] 1 row affected in 6 ms
[2026-01-06 14:44:28] STREAMING_USER> INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'FAMILY', 36, 1)
[2026-01-06 14:44:28] 1 row affected in 8 ms
[2026-01-06 14:44:28] STREAMING_USER> INSERT INTO ABONAMENTE VALUES (SEQ_ABONAMENTE.NEXTVAL, 'ANUAL', 200, 12)
[2026-01-06 14:44:28] 1 row affected in 9 ms
  
```

## • Artisti

```

Database Explorer          Version control
+ [ ] @laborator 2 of 188  cod.sql
  + [ ] @localhost 2 of 39
    > [ ] PUBLIC
    > [ ] STREAMING_USER
      > [ ] tables 11
        > [ ] ARTISTI
      > [ ] UTILIZATORI
      > [ ] console 63 ms
      > [ ] ARTISTI 404 ms

Services
[ ] + [ ] - [ ] x
Tx [ ] Database
  > [ ] @localhost
    > [ ] cod.sql
    < UTILIZATORI
    > [ ] console 63 ms
    < ARTISTI

Database Console: > @localhost [cod.sql] console[@localhost]
  
```

```

 1 ✓ INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'NOVA', 'IONESCU', 'ANDREI', 'ROMANIA');
 2 ✓ INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'ELEM', 'MARINESCU', 'ELENA', 'ROMANIA');
 3 ✓ INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'SKYLINE', 'JOHN', 'CARTER', 'USA');
 4 ✓ INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'MARTIN-PIERRE', 'MARTIN', 'PIERRE', 'FRANTA');
 5 ✓ INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'SOL', 'CARLOS', 'GOMEZ', 'SPANIA');
 6 ✓ INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'AJ', 'AVERY', 'JEAN', 'UK');

  
```

```

Database Explorer          console [@localhost] x cod.sql
+  -  ↻  ⌂  🌐  DDL >  ↻  ⌂  🌐  Tx: Manual  ✓  ⌂  ...  ↻  🔍  ⚙  -  ✎  ×
  >  @laborator  2 of 188
  >  @localhost  2 of 39
  >  PUBLIC
  >  STREAMING_USER
    >  tables  11
      >  ABONAMENTE
      >  ALBUME
Services
  +  -  ⌂  🌐  x
  TNS  Database
    >  @localhost
      cod.sql
      UTILIZATORI
      >  ...
Output  STREAMING_USER.UTILIZATORI
[2026-01-06 16:05:07] STREAMING_USER> INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'NOVA', 'IONESCU', 'ANDREI', 'ROMANIA');
[2026-01-06 16:05:07] STREAMING_USER> INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'ELEM', 'MARINESCU', 'ELENA', 'ROMANIA');
[2026-01-06 16:05:07] STREAMING_USER> INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'SKYLINE', 'JOHN', 'CARTER', 'USA');
[2026-01-06 16:05:07] STREAMING_USER> INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'MARTIN-PIERRE', 'MARTIN', 'PIERRE', 'FRANTA');
[2026-01-06 16:05:07] STREAMING_USER> INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'SOL', 'CARLOS', 'GOMEZ', 'SPANIA');
[2026-01-06 16:05:07] STREAMING_USER> INSERT INTO ARTISTI VALUES (SEQ_ARTISTI.NEXTVAL, 'AJ', 'AVERY', 'JEAN', 'UK');

```

- *Case\_Discuri*

```

INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'UNIVERSAL-MUSIC', DATE '1999-01-01', 'USA');
INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'SONY-MUSIC', DATE '2015-03-15', 'USA');
INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'WARNER-MUSIC', DATE '2009-09-01', 'USA');
INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'RO-MUSIC', DATE '2006-05-20', 'ROMANIA');
INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'ROYAL-MUSIC', DATE '2004-03-14', 'UK');
insert into CASE_DISCURI values (SEQ_CASE_DISCURI.NEXTVAL, 'ANONYMOUS', DATE '2020-02-20', 'UK');

```

```

Database Explorer          console [@localhost] x cod.sql
+  -  ↻  ⌂  🌐  DDL >  ↻  ⌂  🌐  Tx: Manual  ✓  ⌂  ...  ↻  🔍  ⚙  -  ✎  ×
  >  @laborator  2 of 188
  >  @localhost  2 of 39
  >  PUBLIC
  >  STREAMING_USER
    >  tables  11
      >  ABONAMENTE
      >  ALBUME
Services
  +  -  ⌂  🌐  x
  TNS  Database
    >  @localhost
      cod.sql
      UTILIZATORI
      >  ...
      >  console 38 ms
      >  ABONAMENTE
Output  STREAMING_USER.UTILIZATORI
[2026-01-06 16:06:11] STREAMING_USER> INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'RO-MUSIC', DATE '2006-05-20', 'R');
[2026-01-06 16:06:11] STREAMING_USER> INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'ROYAL-MUSIC', DATE '2004-03-14', 'R');
[2026-01-06 16:06:11] STREAMING_USER> INSERT INTO CASE_DISCURI VALUES (SEQ_CASE_DISCURI.NEXTVAL, 'ROYAL-MUSIC', DATE '2004-03-14', 'UK');

```

- *Genuri\_muzicale*

```

INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'POP', '2000');
INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'ROCK', '1970');
INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'ROCK-PUNK', '2005');
INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'JAZZ', '1950');
INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'ELECTRONIC', '1990');
INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'HIP-HOP', '2010');
INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'WHITE-GIRL', '2015');

```

laborator

Database Explorer

Object Explorer

SQL Editor

Output

```
1 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'POP', PERIOADA '2000');
2 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'ROCK', PERIOADA '1970');
3 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'ROCK-PUCK', PERIOADA '2005');
4 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'JAZZ', PERIOADA '1950');
5 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'ELECTRONIC', PERIOADA '1990');
6 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'HIP-HOP', PERIOADA '2010');
7 ✓ INSERT INTO GENURI_MUZICALE VALUES (ID_GEN SEQ.GENURI.NEXTVAL, NUME 'WHITE-GIRL', PERIOADA '2015');
```

Services

Output

```
[2026-01-06 16:10:03] STREAMING_USER> INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'HIP-HOP', '2010')
[2026-01-06 16:10:03] 1 row affected in 4 ms
[2026-01-06 16:10:03] STREAMING_USER> INSERT INTO GENURI_MUZICALE VALUES (SEQ_GENURI.NEXTVAL, 'WHITE-GIRL', '2015')
[2026-01-06 16:10:03] 1 row affected in 5 ms
```

- *Albume*

```
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'PRIMUL_PAS', DATE '2018-06-01', 1);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'ECLIPSA', DATE '2019-09-15', 2);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'HORIZON', DATE '2017-04-10', 3);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'OMBRE', DATE '2020-02-20', 4);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'CALOR', DATE '2021-08-05', 5);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'FIRST-LIGHT', DATE '2022-03-01', 6);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'REBORN', DATE '2016-09-01', 3);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'BLUE-NOTES', DATE '2019-11-20', 4);
INSERT INTO ALBUME VALUES (SEQ_ALBUME.NEXTVAL, 'DIGITAL-SOUL', DATE '2021-04-15', 2);
```

The screenshot shows the Visual Studio Code interface with the following details:

- Top Bar:** Shows tabs for "laborator" and "Version control".
- Database Explorer:** Shows connections to "[@localhost] 2 of 188" and "[@localhost] 2 of 39". A context menu is open over the "[@localhost] 2 of 39" connection, with options like "PUBLIC", "STREAMING\_USER", and "ABONAMENTE".
- Code Editor:** Displays a SQL script for inserting data into the "ALBUM" table. The script includes 10 INSERT statements with values ranging from 'PRTMUL\_PAS' to 'DIGITAL-SOUL'. The "STREAMING\_USER" connection is highlighted.
- Output Panel:** Shows the results of the last three INSERT statements:
  - [2026-01-06 16:34:40] STREAMING\_USER> INSERT INTO ALBUM VALUES (SEQ\_ALBUM.NEXTVAL, 'REBORN', DATE '2016-09-01', 3)
  - [2026-01-06 16:34:40] 1 row affected in 5 ms
  - [2026-01-06 16:34:40] STREAMING\_USER> INSERT INTO ALBUM VALUES (SEQ\_ALBUM.NEXTVAL, 'BLUE-NOTES', DATE '2019-11-20', 4)
  - [2026-01-06 16:34:40] 1 row affected in 4 ms
  - [2026-01-06 16:34:40] STREAMING\_USER> INSERT INTO ALBUM VALUES (SEQ\_ALBUM.NEXTVAL, 'DIGITAL-SOUL', DATE '2021-04-15', 2)
  - [2026-01-06 16:34:40] 1 row affected in 4 ms
- Bottom Left:** Services panel with Database, localhost, and console tabs.
- Bottom Right:** Status bar showing "STREAMING\_USER" and "console 417 ms".

#### • Melodii

```
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'START', 210, DATE '2018-06-01', 1, 1);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'SPERANTA', 205, DATE '2018-
```

```

06-01', 1, 1);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'UMBRA', 220, DATE '2019-09-
15', 2, 4);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'FREQ', 210, DATE '2019-09-
15', 2, 4);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'REFLEX', 200, DATE '2019-
09-15', 2, 4);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'SKY-HIGH', 240, DATE '2017-
04-10', 3, 2);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'RUN', 230, DATE '2017-04-
10', 3, 2);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'BREAK-FREE', 225, DATE
'2017-04-10', 3, 2);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'EDGE', 235, DATE '2017-04-
10', 3, 2);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'OMBRE', 255, DATE '2020-02-
20', 4, 3);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'MIDNIGHT', 270, DATE '2020-
02-20', 4, 3);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'BLUE-WALK', 250, DATE
'2020-02-20', 4, 3);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'REAL-TALK', 205, DATE
'2021-08-05', 5, 5);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'URBAN', 215, DATE '2021-08-
05', 5, 5);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'DAWN', 205, DATE '2022-03-
01', 6, 1);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'MORNING', 210, DATE '2022-
03-01', 6, 1);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'CLEAR-SKY', 200, DATE
'2022-03-01', 6, 1);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'BLUE-NOTE', 275, DATE
'2019-11-20', 8, 3);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'SMOOTH-WALK', 265, DATE
'2019-11-20', 8, 3);
INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'LATE-NIGHT', 280, DATE
'2019-11-20', 8, 3);

```

The screenshot shows the DataGrip IDE interface with the following components:

- Database Explorer:** Shows the database structure with tables like ABONAMENTE, ALUME, ARTISTI, CASE\_DISCURSI, GENURLMIZICALE, ISTORIC\_ABONAMENTE, ISTORIC\_CONTRACT, MELODI, REDARI, UMRARIRI, UTILATORI, and sequences.
- SQL Editor:** Contains the SQL code for inserting data into the MELODII table.
- Log Tab:** Displays the execution log with the following entries:

```

[2020-01-06 10:30:37] 1 row affected in 4 ms
[2020-01-06 10:38:57] STREAMING_USER> INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'BLUE-NOTE', 275, DATE '2019-11-20', 8, 3)
[2020-01-06 10:38:57] 1 row affected in 4 ms
[2020-01-06 10:38:57] STREAMING_USER> INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'SMOOTH-WALK', 265, DATE '2019-11-20', 8, 3)
[2020-01-06 10:38:57] 1 row affected in 3 ms
[2020-01-06 10:38:57] STREAMING_USER> INSERT INTO MELODII VALUES (SEQ_MELODII.NEXTVAL, 'LATE-NIGHT', 280, DATE '2019-11-20', 8, 3)
[2020-01-06 10:38:57] 1 row affected in 5 ms

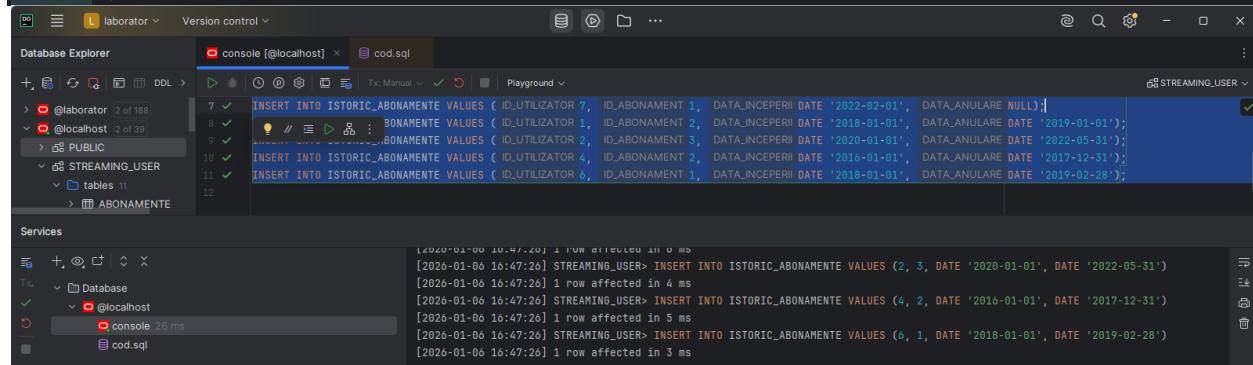
```

- *Istoric\_Abonamente*

```

INSERT INTO ISTORIC_ABONAMENTE VALUES (1, 3, DATE '2021-01-01', NULL);
INSERT INTO ISTORIC_ABONAMENTE VALUES (2, 2, DATE '2022-06-01', NULL);
INSERT INTO ISTORIC_ABONAMENTE VALUES (3, 4, DATE '2018-01-01', DATE '2020-01-01');
INSERT INTO ISTORIC_ABONAMENTE VALUES (3, 5, DATE '2020-01-02', NULL);
INSERT INTO ISTORIC_ABONAMENTE VALUES (4, 1, DATE '2015-01-01', DATE '2015-12-31');
);
INSERT INTO ISTORIC_ABONAMENTE VALUES (6, 2, DATE '2019-03-01', NULL);
INSERT INTO ISTORIC_ABONAMENTE VALUES (7, 1, DATE '2022-02-01', NULL);
INSERT INTO ISTORIC_ABONAMENTE VALUES (1, 2, DATE '2018-01-01', DATE '2019-01-01');
INSERT INTO ISTORIC_ABONAMENTE VALUES (2, 3, DATE '2020-01-01', DATE '2022-05-31');
INSERT INTO ISTORIC_ABONAMENTE VALUES (4, 2, DATE '2016-01-01', DATE '2017-12-31');
INSERT INTO ISTORIC_ABONAMENTE VALUES (6, 1, DATE '2018-01-01', DATE '2019-02-28');

```



- *Urmăriți*

```

INSERT INTO URMARIRI VALUES (1, 1, DATE '2019-05-01', NULL, 'Y');
INSERT INTO URMARIRI VALUES (1, 3, DATE '2020-06-01', NULL, 'N');
INSERT INTO URMARIRI VALUES (2, 2, DATE '2019-03-10', DATE '2023-10-11', 'Y');
INSERT INTO URMARIRI VALUES (2, 4, DATE '2021-01-01', NULL, 'Y');
INSERT INTO URMARIRI VALUES (3, 5, DATE '2016-07-01', NULL, 'N');
INSERT INTO URMARIRI VALUES (3, 1, DATE '2018-09-09', NULL, 'Y');
INSERT INTO URMARIRI VALUES (4, 2, DATE '2020-02-20', NULL, 'Y');
INSERT INTO URMARIRI VALUES (5, 3, DATE '2022-01-10', DATE '2025-02-09', 'Y');
INSERT INTO URMARIRI VALUES (6, 4, DATE '2021-05-05', NULL, 'N');
INSERT INTO URMARIRI VALUES (7, 6, DATE '2022-08-01', NULL, 'Y');
INSERT INTO URMARIRI VALUES (1, 4, DATE '2021-07-01', NULL, 'Y');
INSERT INTO URMARIRI VALUES (2, 5, DATE '2022-03-01', NULL, 'N');
INSERT INTO URMARIRI VALUES (3, 2, DATE '2017-04-01', DATE '2019-01-01', 'Y');

```

```

12 ✓ INSERT INTO URMARIRI VALUES ( ID_UTILIZATOR 2 , ID_ARTIST 5 , DATA_URMARIRI DATE '2022-03-01' , DATA_INCHEIERE NULL , NOTIFICARI 'N' );
13 ✓ INSERT INTO URMARIRI VALUES ( ID_UTILIZATOR 3 , ID_ARTIST 2 , DATA_URMARIRI DATE '2017-04-01' , DATA_INCHEIERE DATE '2019-01-01' , NOTIFICARI 'Y' );

```

Execution log:

```

[2020-01-01 10:47:24] 1 row affected in 3 ms
[2020-01-01 10:47:24] STREAMING_USER> INSERT INTO URMARIRI VALUES (1, 4, DATE '2021-07-01', NULL, 'Y')
[2020-01-01 10:47:24] 1 row affected in 3 ms
[2020-01-01 10:49:24] STREAMING_USER> INSERT INTO URMARIRI VALUES (2, 5, DATE '2022-03-01', NULL, 'N')
[2020-01-01 10:49:24] 1 row affected in 5 ms
[2020-01-01 10:49:24] STREAMING_USER> INSERT INTO URMARIRI VALUES (3, 2, DATE '2017-04-01', DATE '2019-01-01', 'Y')
[2020-01-01 10:49:24] 1 row affected in 4 ms

```

- Istoric\_Contracte*

```

INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 1, 4, DATE '2017-01-01', DATE '2019-01-01');
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 1, 1, DATE '2019-01-02', NULL);
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 2, 4, DATE '2018-05-01', DATE '2020-05-01');
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 2, 2, DATE '2020-05-02', NULL);
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 3, 3, DATE '2016-01-01', NULL);
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 4, 5, DATE '2019-02-01', NULL);
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 5, 2, DATE '2021-06-01', NULL);
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 6, 1, DATE '2022-03-01', NULL);
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 1, 3, DATE '2014-01-01', DATE '2016-12-31');
INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 3, 1, DATE '2010-01-01', DATE '2015-12-31');

```

```

1 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 1 , ID_CASADISCURI 4 , DATA_INCEPERI DATE '2017-01-01' , DATA_INCHEIERE DATE '2017-01-01' );
2 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 1 , ID_CASADISCURI 1 , DATA_INCEPERI DATE '2019-01-02' , DATA_INCHEIERE NULL );
3 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 2 , ID_CASADISCURI 4 , DATA_INCEPERI DATE '2018-05-01' , DATA_INCHEIERE DATE '2018-05-01' );
4 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 2 , ID_CASADISCURI 2 , DATA_INCEPERI DATE '2020-05-02' , DATA_INCHEIERE NULL );
5 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 3 , ID_CASADISCURI 3 , DATA_INCEPERI DATE '2016-01-01' , DATA_INCHEIERE NULL );
6 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 4 , ID_CASADISCURI 5 , DATA_INCEPERI DATE '2019-02-01' , DATA_INCHEIERE NULL );
7 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 5 , ID_CASADISCURI 2 , DATA_INCEPERI DATE '2021-06-01' , DATA_INCHEIERE NULL );
8 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 6 , ID_CASADISCURI 1 , DATA_INCEPERI DATE '2022-03-01' , DATA_INCHEIERE NULL );
9 ✓ INSERT INTO ISTORIC_CONTRACTE VALUES ( NR_CONTRACT SEQ_CONTRACTE.NEXTVAL, ID_ARTIST 1 , ID_CASADISCURI 3 , DATA_INCEPERI DATE '2014-01-01' , DATA_INCHEIERE DATE '2014-01-01' );

```

Execution log:

```

[2020-01-01 10:50:31] STREAMING_USER> INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 3, 1, DATE '2010-01-01', DATE '2010-01-01')
[2020-01-01 10:50:31] 1 row affected in 4 ms
[2020-01-01 10:50:31] STREAMING_USER> INSERT INTO ISTORIC_CONTRACTE VALUES (SEQ_CONTRACTE.NEXTVAL, 4, 2, DATE '2015-01-01', DATE '2015-01-01')
[2020-01-01 10:50:31] 1 row affected in 4 ms

```

- Redări

```

INSERT INTO REDARI VALUES (1, 1, DATE '2019-06-01', 180, 'MOBIL');
INSERT INTO REDARI VALUES (1, 5, DATE '2020-06-02', 200, 'DESKTOP');
INSERT INTO REDARI VALUES (1, 9, DATE '2021-04-20', 190, 'TABLETA');
INSERT INTO REDARI VALUES (2, 3, DATE '2020-10-01', 200, 'MOBIL');
INSERT INTO REDARI VALUES (3, 7, DATE '2021-03-01', 225, 'TV');
INSERT INTO REDARI VALUES (3, 1, DATE '2022-01-15', 210, 'MOBIL');
INSERT INTO REDARI VALUES (4, 11, DATE '2022-09-01', 205, 'MOBIL');
INSERT INTO REDARI VALUES (4, 15, DATE '2022-10-01', 205, 'DESKTOP');
INSERT INTO REDARI VALUES (5, 3, DATE '2022-11-05', 220, 'TV');
INSERT INTO REDARI VALUES (6, 3, DATE '2021-12-01', 215, 'TABLETA');
INSERT INTO REDARI VALUES (7, 6, DATE '2022-12-20', 210, 'MOBIL');
INSERT INTO REDARI VALUES (1, 1, DATE '2022-01-10', 200, 'MOBIL');
INSERT INTO REDARI VALUES (1, 1, DATE '2022-02-15', 210, 'MOBIL');
INSERT INTO REDARI VALUES (1, 4, DATE '2022-03-01', 195, 'DESKTOP');
INSERT INTO REDARI VALUES (2, 3, DATE '2022-01-12', 200, 'MOBIL');
INSERT INTO REDARI VALUES (2, 3, DATE '2022-01-20', 190, 'TABLETA');
INSERT INTO REDARI VALUES (2, 6, DATE '2022-02-10', 210, 'DESKTOP');
INSERT INTO REDARI VALUES (3, 7, DATE '2022-02-01', 225, 'TV');
INSERT INTO REDARI VALUES (3, 7, DATE '2022-02-03', 225, 'TV');
INSERT INTO REDARI VALUES (4, 9, DATE '2022-03-05', 215, 'MOBIL');
INSERT INTO REDARI VALUES (4, 10, DATE '2022-03-07', 195, 'DESKTOP');
INSERT INTO REDARI VALUES (5, 11, DATE '2022-04-01', 205, 'MOBIL');
INSERT INTO REDARI VALUES (5, 11, DATE '2022-04-02', 200, 'MOBIL');
INSERT INTO REDARI VALUES (5, 14, DATE '2022-04-10', 215, 'TV');
INSERT INTO REDARI VALUES (6, 15, DATE '2022-05-01', 205, 'TABLETA');
INSERT INTO REDARI VALUES (6, 15, DATE '2022-05-03', 205, 'TABLETA');
INSERT INTO REDARI VALUES (7, 18, DATE '2022-06-01', 230, 'DESKTOP');
INSERT INTO REDARI VALUES (7, 18, DATE '2022-06-05', 225, 'DESKTOP');

UPDATE REDARI
SET DATA_REDARE =
    TRUNC(DATA_REDARE)
    + NUMTODSINTERVAL(
        DBMS_RANDOM.VALUE(0, 86400),
        'SECOND'
    );

```

The screenshot shows the DataGrip IDE interface. The Database Explorer panel on the left lists the database schema, including tables like 'REDARI', 'ABONAMENTE', and 'ALBUME'. The code editor panel on the right contains the SQL script provided above. The bottom status bar indicates the execution time of the last command was 35 ms.

## 6. Subprogram stocat independent care să utilizeze toate cele 3 tipuri de colecții studiate

Pentru un utilizator al unei platforme de streaming muzical, se dorește realizarea unui mecanism simplu de recomandare a noi melodii, bazat pe preferințele sale de ascultare.

Sistemul analizează istoricul redarilor utilizatorului și determină genurile muzicale pe care acesta le ascultă cel mai frecvent, **memorând frecvența acestora într-un tablou indexat**. Pe baza acestor informații, sunt selectate **primele trei genuri cu cele mai multe redari**, care sunt stocate **într-un tablou imbricat**.

Pornind de la genurile preferate identificate, se construiește **o lista de maximum șapte melodii recomandate pe care nu le-a mai ascultat încă dinainte**, stocată **într-un vector**, care este afișată utilizatorului.

```
create or replace procedure recommandari_utilizator (p_id in
UTILIZATORI.ID_UTILIZATOR%type) is
    type t_gen_cnt is table of number index by PLS_INTEGER;
    v_gen_cnt t_gen_cnt;
    type t_gen is table of genuri_muzicale.id_gen%type;
    v_gen t_gen := t_gen();
    type t_recomandari is varray(7) of varchar2(50);
    v_recomandari t_recomandari := t_recomandari();

    cursor c_redari is
        select M.ID_GEN as gen
        from REDARI R
        join MELODII M on R.ID_MELODIE = M.ID_MELODIE
        where R.ID_UTILIZATOR=p_id;

    v_max_gen  genuri_muzicale.id_gen%type;
    v_max_cnt  NUMBER;
    g_genuri_muzicale.id_gen%TYPE;
    v_user number;
begin
    begin
        select 1 into v_user
        from UTILIZATORI
        where ID_UTILIZATOR = p_id;
        exception
            when no_data_found then
                dbms_output.put_line('Utilizator neexistent');
                return;
    end;
    for redare in c_redari loop
        if v_gen_cnt.exists(redare.gen) then
            v_gen_cnt(redare.gen) := v_gen_cnt(redare.gen)+1;
        else
            v_gen_cnt(redare.gen) := 1;
        end if;
    end loop;
    for i in 1..3 loop
```

```

v_max_cnt := -1;
v_max_gen:= null;

g := v_gen_cnt.first;
while g is not null loop
    if v_gen_cnt(g) > v_max_cnt then
        v_max_cnt := v_gen_cnt(g);
        v_max_gen:= g;
    end if;
    g := v_gen_cnt.next(g);
end loop;

exit when v_max_gen is null;

v_gen.extend;
v_gen(v_gen.last) := V_MAX_GEN;

v_gen_cnt(V_MAX_GEN) := -1;
end loop;

for i in 1 .. v_gen.count loop
    for r in (
        select M.nume
        from MELODII M
        where M.ID_GEN = v_gen(i)
            and not exists (
                select 1
                from REDARI R
                where R.ID_UTILIZATOR = p_id and R.ID_MELODIE =
M.ID_MELODIE
            )
    ) loop
        exit when v_recomandari.count = 7;
        v_recomandari.extend;
        v_recomandari(v_recomandari.last) := r.nume;
    end loop;
end loop;

if v_recomandari.COUNT = 0 then
    DBMS_OUTPUT.PUT_LINE('Nu exista suficiente date pentru recomandari.');
else
    DBMS_OUTPUT.PUT_LINE('Recomandari pentru utilizatorul ' || p_id);
    for i in 1..v_recomandari.count loop
        DBMS_OUTPUT.PUT_LINE(v_recomandari(i));
    end loop;
end if;

end;

```

a) Cazul în care sunt destule date pentru a se face recomandarea (utilizatorul a ascultat măcar o melodie)

The screenshot shows a database management interface with a dark theme. In the top navigation bar, there are tabs for 'laborator' and 'cod.sql'. The main area is titled 'Database Explorer' and shows a tree view of databases and tables. A SQL editor window is open, displaying the following code:

```
1 create or replace procedure recommandari_utilizator (p_id in UTILIZATORI.ID_UTILIZATOR%type) is
 2   end;
 3
 4   begin
 5     recommandari_utilizator( p_id );
 6   end;
```

Below the code, the output of the execution is shown:

```
[2026-01-06 19:39:55] completed in 13 ms
Recomandari pentru utilizatorul 3
SKY-HIGH
RUN
BREAK-FREE
EDGE
START
SPERANTA
DAWN
```

The bottom status bar indicates '81:1 CRLF UTF-8 4 spaces'.

b) Cazul în care utilizatorul nu a redat nicio piesă

The screenshot shows a database management interface with a dark theme. In the top navigation bar, there are tabs for 'laborator' and 'cod.sql'. The main area is titled 'Database Explorer' and shows a tree view of databases and tables. A SQL editor window is open, displaying the same code as in the previous screenshot:

```
1 create or replace procedure recommandari_utilizator (p_id in UTILIZATORI.ID_UTILIZATOR%type) is
 2   end;
 3
 4   begin
 5     recommandari_utilizator( p_id );
 6   end;
```

Below the code, the output of the execution is shown:

```
[2026-01-06 19:41:26] STREAMING_USER> begin
  recommandari_utilizator(9);
end;
[2026-01-06 19:41:26] completed in 11 ms
Nu există suficiente date pentru recomandări.
```

c) Cazul în care utilizatorul dat ca parametru nu există

The screenshot shows a database management interface with a dark theme. In the top navigation bar, there are tabs for 'laborator' and 'cod.sql'. The main area is titled 'Database Explorer' and shows a tree view of databases and tables. A SQL editor window is open, displaying the same code as in the previous screenshots:

```
1 create or replace procedure recommandari_utilizator (p_id in UTILIZATORI.ID_UTILIZATOR%type) is
 2   end;
 3
 4   begin
 5     recommandari_utilizator( p_id );
 6   end;
```

Below the code, the output of the execution is shown:

```
[2026-01-06 19:41:26] completed in 11 ms
Nu există suficiente date pentru recomandări.
[2026-01-06 19:41:57] STREAMING_USER> begin
  recommandari_utilizator(999);
end;
[2026-01-06 19:41:57] completed in 11 ms
Utilizator neexistent
```

## 7. Subprogram stocat **independent** care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat, **dependent de celălalt cursor**

Se dorește afișarea, pentru fiecare **casă de discuri**, a primelor **3 melodii cele mai difuzate**, aparținând artiștilor care au avut contract cu respectiva casă de discuri.

Problema este rezolvată printr-un **subprogram stocat independent** care utilizează **două tipuri diferite de cursoare**:

- un cursor neparametrizat pentru parcurgerea caselor de discuri;
- un cursor parametrizat, dependent de primul, care determină melodiile cele mai difuzate pentru fiecare casă de discuri.

```
create or replace procedure top_melodii_case_discuri is

    cursor c_case is
        select C.ID_CASADISCURI, C.NUME
        from CASE_DISCURI C
        order by C.NUME;

    cursor c_top_melodii (p_id_casa in CASE_DISCURI.ID_CASADISCURI%type) is
        select M.NUME as melodie, count(R.ID_MELODIE) as nr_redari
        from ISTORIC_CONTRACTE IC
        join ARTISTI A on A.ID_ARTIST = IC.ID_ARTIST
        join ALBUME AL on AL.ID_ARTIST = A.ID_ARTIST
        join MELODII M on M.ID_ALBUM = AL.ID_ALBUM
        join REDARI R on R.ID_MELODIE = M.ID_MELODIE
        where IC.ID_CASADISCURI = p_id_casa and trunc(R.DATA_REDARE) between
IC.DATA_INCEPERII and NVL(IC.DATA_INCHEIERE, SYSDATE)
        group by M.NUME
        order by count(R.ID_MELODIE) desc;

    v_cnt number;

begin
    for c in c_case loop
        dbms_output.put_line('Casa de discuri: ' || c.NUME);

        v_cnt := 0;
        for m in c_top_melodii(c.ID_CASADISCURI) loop
            exit when v_cnt = 3;
            dbms_output.put_line(m.melodie || ':' || m.nr_redari || ' '
redari');
            v_cnt := v_cnt + 1;
        end loop;
    end loop;
end;
```

```

        if v_cnt = 0 then
            dbms_output.put_line('Nu exista suficiente date.');
        end if;

    end loop;
end;
/

```

The screenshot shows a database environment with the following details:

- Database Explorer:** Shows connections to @laborator (2 of 188) and localhost (2 of 39). Under the STREAMING\_USER schema, there are tables: ABONAMENT (43 rows) and ALBUME (44 rows).
- Services:** Shows a list of services including REDARI, cod.sql, MELODII, and UTILIZATORI.
- Console:** The current tab is "console [@localhost]". The code being run is a PL/SQL block that prints messages and updates a table named "top\_melodii\_case\_discuri".
- Output:** The console output shows the results of the query execution, including the count of records inserted and the top 10 entries from the table.

```

[2026-01-06 20:23:59] completed in 13 ms
Casa de discuri: ANONYMOUS
Nu exista suficiente date.
Casa de discuri: RO-MUSIC
UMBRA: 5 redari
START: 4 redari
REFLEX: 1 redari
Casa de discuri: ROYAL-MUSIC
MIDNIGHT: 3 redari
BLUE-NOTE: 2 redari
OMBRE: 1 redari
Casa de discuri: SONY-MUSIC
UMBRA: 5 redari
MIDNIGHT: 3 redari
URBAN: 2 redari
Casa de discuri: UNIVERSAL-MUSIC
START: 4 redari
RUN: 3 redari
DAWN: 3 redari
Casa de discuri: WARNER-MUSIC
START: 4 redari
RUN: 3 redari
FINN: 2 redari

```

## 8. Subprogram stocat independent de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele create.

Se dorește determinarea **genului muzical asociat unui album**, pe baza melodilor care îl compun. Problema este rezolvată printr-un **subprogram stocat independent de tip funcție**, care utilizează **interrogări SQL asupra tabelelor ALBUME, MELODII și GENURI\_MUZICALE**.

Funcția tratează următoarele situații:

- cazul în care albumul specificat nu există în baza de date;
- cazul în care albumul există, dar nu conține suficiente date pentru a determina un gen muzical;
- cazul în care albumul conține melodii încadrate în mai multe genuri muzicale, situație în care nu se poate decide un singur gen.

```
create or replace function gen_album ( p_id in ALBUME.ID_ALBUM%type) return varchar2 is
  v_gen GENURI_MUZICALE.NUME%type;
  v_cnt number;
  e_album_inexistent exception;

begin
  select count(*) into v_cnt
  from ALBUME
  where ID_ALBUM = p_id;

  if v_cnt = 0 then
    raise e_album_inexistent;
  end if;

  select distinct G.NUME into v_gen
  from ALBUME A
  join MELODII M on M.ID_ALBUM = A.ID_ALBUM
  join GENURI_MUZICALE G on G.ID_GEN = M.ID_GEN
  where A.ID_ALBUM = p_id;

  return v_gen;

exception
  when e_album_inexistent then
    return 'Albumul nu există.';
  when no_data_found then
    return 'Nu există date suficiente.';
  when too_many_rows then
    return 'Genuri multiple, nu se poate decide doar unul.';
end;
/
```

The screenshot shows a database interface with a code editor and an output window. The code editor contains the following PL/SQL block:

```
create or replace function gen_album ( p_id in ALBUME.ID_ALBUM%type) return varchar2 is
begin
    return 'Genuri multiple, nu se poate decide doar unul';
end;
/
select gen_album( p_id 1) as rezultat
from dual;
```

The output window shows the result of the query:

REZULTAT
1 POP

The screenshot shows a database interface with a code editor and an output window. The code editor contains the same PL/SQL block as the first screenshot, but with a different parameter value:

```
select gen_album( p_id 999) as rezultat
from dual;
```

The output window shows the result of the query:

REZULTAT
1 Albumul nu exista.

The screenshot shows a database interface with a code editor and an output window. The code editor contains the same PL/SQL block as the previous screenshots, but with a different parameter value:

```
select gen_album( p_id 8) as rezultat
from dual;
```

The output window shows the result of the query:

REZULTAT
1 Genuri multiple, nu se poate decide doar unul.

## 9. Subprogram stocat independent de tip procedură care să aibă minim 2 parametri și să utilizeze într-o singură comandă SQL 5 dintre tabelele create

Se dorește determinarea **artistului sau artiștilor cel mai ascultați** de către utilizatorii aparținând unei **categoriai de vîrstă date**, într-o perioadă de timp specificată.

Problema este rezolvată printr-un **subprogram stocat independent de tip procedură**, care utilizează o interogare SQL asupra tabelelor **UTILIZATORI**, **REDARI**, **MELODII**, **ALBUME** și **ARTISTI**.

Procedura tratează următoarele situații:

- cazul în care nu există utilizatori care să se încadreze în categoria de vîrstă specificată, raportată la momentul efectuării redărilor;
- cazul în care există utilizatori, dar nu există redări în perioada analizată;
- cazul în care mai mulți artiști au același număr maxim de redări, situație în care sunt afișați toți artiștii aflați la egalitate.

```
create or replace procedure artist_varsta_perioada (p_varsta_min in number, p_varsta_max in number,
p_data_start in date, p_data_end in date) is
  type t_artist_tab is table of ARTISTI.NUME_SCENA%type;
  v_artisti t_artist_tab;

  type t_cnt_tab is table of number;
  v_cnturi t_cnt_tab;

  v_cnt_user number;
  v_cnt_redari number;

  e_fara_utilizatori exception;
  e_fara_redari   exception;

begin
  select count(*) into v_cnt_user
  from UTILIZATORI U
  where trunc(months_between(sysdate, U.DATA_NASTERE) / 12)
    between p_varsta_min and p_varsta_max;

  if v_cnt_user = 0 then
    raise e_fara_utilizatori;
  end if;

  select count(*) into v_cnt_redari
  from UTILIZATORI U
  join REDARI R on R.ID_UTILIZATOR = U.ID_UTILIZATOR
```

```

where trunc(months_between(R.DATA_REDARE, U.DATA_NASTERE) / 12)
      between p_varsta_min and p_varsta_max
and trunc(R.DATA_REDARE) between p_data_start and p_data_end;

if v_cnt_redari = 0 then
    raise e_fara_redari;
end if;

select A.NUME_SCENA, count(*) bulk collect into v_artisti, v_cnturi
from UTILIZATORI U
join REDARI R on R.ID_UTILIZATOR = U.ID_UTILIZATOR
join MELODII M on M.ID_MELODIE = R.ID_MELODIE
join ALBUME AL on AL.ID_ALBUM = M.ID_ALBUM
join ARTISTI A on A.ID_ARTIST = AL.ID_ARTIST
where trunc(months_between(R.DATA_REDARE, U.DATA_NASTERE) / 12)
      between p_varsta_min and p_varsta_max
and trunc(R.DATA_REDARE) between p_data_start and p_data_end
group by A.NUME_SCENA
order by count(*) desc;

dbms_output.put_line('Artistii cei mai ascultati sunt:');

for i in 1 .. v_artisti.count loop
    exit when v_cnturi(i) < v_cnturi(1);
    dbms_output.put_line(v_artisti(i));
end loop;

exception
when e_fara_utilizatori then
    dbms_output.put_line('Nu exista utilizatori in categoria de varsta data.');
when e_fara_redari then
    dbms_output.put_line('Nu exista redari in perioada specificata.');

```

The screenshot shows the Visual Studio Code interface with the following components:

- Database Explorer** pane on the left, showing a tree structure of databases and tables.
- Code Editor** pane in the center, displaying a SQL procedure definition:

```
1  create or replace procedure artist_varsta_perioada (p_varsta_min in number, p_varsta_max in number, p_data_start in date, p_data_end in date) is
2      begin
3          artist_varsta_perioada(p_varsta_min 18, p_varsta_max 30, p_data_start date '2022-01-01', p_data_end date '2022-12-31');
4      end;
5  /
```

- Services** pane at the bottom, showing a list of running services, including a database connection to "localhost".

The screenshot shows a database interface with a code editor and a terminal window. The code editor contains a stored procedure named `artist_varsta_perioada`. The terminal window shows the execution of this procedure with parameters `p_varsta_min = 18` and `p_varsta_max = 30`, returning the message: "[2026-01-07 16:22:43] STREAMING\_USER> begin artist\_varsta\_perioada(18,30,date '1950-01-01',date '1950-12-31'); end;" and "[2026-01-07 16:22:43] completed in 6 ms Nu exista utilizatori in categoria de varsta data."

```
create or replace procedure artist_varsta_perioada (p_varsta_min in number, p_varsta_max in number, p_data_start in date, p_data_end in date) is
begin
    artist_varsta_perioada(p_varsta_min 18, p_varsta_max 30, p_data_start date '1950-01-01', p_data_end date '1950-12-31');
end;
```

The screenshot shows the same database interface. The stored procedure `artist_varsta_perioada` is executed again with parameters `p_varsta_min = 80` and `p_varsta_max = 100`, returning the message: "[2026-01-07 16:25:26] STREAMING\_USER> begin artist\_varsta\_perioada(80,100,date '2022-01-01',date '2022-12-31'); end;" and "[2026-01-07 16:25:26] completed in 6 ms Nu exista utilizatori in categoria de varsta data."

```
create or replace procedure artist_varsta_perioada (p_varsta_min in number, p_varsta_max in number, p_data_start in date, p_data_end in date) is
begin
    artist_varsta_perioada(p_varsta_min 80, p_varsta_max 100, p_data_start date '2022-01-01', p_data_end date '2022-12-31');
end;
```

## 10. Definiți un *trigger* de tip LMD la nivel de comandă. Declanșați *trigger*-ul.

- Un utilizator nu poate avea 2 abonamente în același timp, implicit nu poate avea 2 abonamente active.

```
CREATE OR REPLACE TRIGGER TRG_IA_FARA_SUPRAPUNERI
AFTER INSERT OR UPDATE ON ISTORIC_ABONAMENTE
DECLARE
    v_user ISTORIC_ABONAMENTE.ID_UTILIZATOR%TYPE;
BEGIN
    SELECT DISTINCT a.ID_UTILIZATOR INTO v_user
    FROM ISTORIC_ABONAMENTE a
    JOIN ISTORIC_ABONAMENTE b
        ON a.ID_UTILIZATOR = b.ID_UTILIZATOR
       AND a.DATA_INCEPERII < NVL(b.DATA_ANULARE, DATE '9999-12-31')
       AND b.DATA_INCEPERII < NVL(a.DATA_ANULARE, DATE '9999-12-31')
       AND (a.ID_ABONAMENT != b.ID_ABONAMENT
             OR a.DATA_INCEPERII != b.DATA_INCEPERII)
    FETCH FIRST 1 ROW ONLY;

    RAISE_APPLICATION_ERROR(-20050, 'UTILIZATORUL ' || v_user || ' ARE
ABONAMENTE CU PERIOADE SUPRAPUSE.');

```

```
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        NULL;
END;
/
```

The screenshot shows the Oracle SQL Developer interface with the following details:

- Database Explorer:** Shows the schema structure under the `@localhost` connection, including tables `ISTO`, `cc`, `kr`, `fc`, `in`, `ct`, and `tr`.
- cod.sql:** A script file containing the trigger definition shown above.
- console:** A tab showing the results of the trigger creation command.

```
[2026-01-07 18:14:27] 2 rows retrieved starting from 1 in 405 ms (execution: 21 ms, fetching: 384 ms)
[2026-01-07 18:16:24] STREAMING_USER> CREATE OR REPLACE TRIGGER TRG_IA_FARA_SUPRAPUNERI
AFTER INSERT OR UPDATE ON ISTORIC_ABONAMENTE
DECLARE
    v_user ISTORIC_ABONAMENTE.ID_UTILIZATOR%TYPE;
BEGIN
    SELECT DISTINCT a.ID_UTILIZATOR INTO v_user
    FROM ISTORIC_ABONAMENTE a
    JOIN ISTORIC_ABONAMENTE b
        ON a.ID_UTILIZATOR = b.ID_UTILIZATOR
       AND a.DATA_INCEPERII < NVL(b.DATA_ANULARE, DATE '9999-12-31')
       AND b.DATA_INCEPERII < NVL(a.DATA_ANULARE, DATE '9999-12-31')
       AND (a.ID_ABONAMENT != b.ID_ABONAMENT
             OR a.DATA_INCEPERII != b.DATA_INCEPERII)
    FETCH FIRST 1 ROW ONLY;

    RAISE_APPLICATION_ERROR(-20050, 'UTILIZATORUL ' || v_user || ' ARE
ABONAMENTE CU PERIOADE SUPRAPUSE.');

```
- STREAMING\_USER.ISTORIC\_ABONAMENTE:** A table viewer showing two rows of data.
- Services:** Shows the status of various services like Database, REDA, and URM.
- Logs:** Shows the execution log with error messages.

ORA-20050: UTILIZATORUL 3 ARE ABONAMENTE CU PERIOADE SUPRAPUSE.  
ORA-06512: at "STREAMING\_USER.TRG\_IA\_FARA\_SUPRAPUNERI", line 15  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_IA\_FARA\_SUPRAPUNERI'

- Un artist nu poate avea contracte cu perioade suprapuse, indiferent de casa de discuri, implicit nu poate avea 2 contracte active.

```

CREATE OR REPLACE TRIGGER TRG_CONT_FARA_SUPRAPUNERI
AFTER INSERT OR UPDATE ON ISTORIC_CONTRACTE
DECLARE
    v_artist ISTORIC_CONTRACTE.ID_ARTIST%TYPE;
BEGIN
    SELECT DISTINCT a.ID_ARTIST INTO v_artist
    FROM ISTORIC_CONTRACTE a
    JOIN ISTORIC_CONTRACTE b
        ON a.ID_ARTIST = b.ID_ARTIST
        AND a.DATA_INCEPERII < NVL(b.DATA_INCHEIERE, DATE '9999-12-31')
        AND b.DATA_INCEPERII < NVL(a.DATA_INCHEIERE, DATE '9999-12-31')
        AND (a.NR_CONTRACT != b.NR_CONTRACT)
    FETCH FIRST 1 ROW ONLY;

    RAISE_APPLICATION_ERROR( -20060, 'ARTISTUL ' || v_artist || ' ARE
CONTRACTE CU PERIOADE SUPRAPUSE.');

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        NULL;
END;
/

```

The screenshot shows the Oracle SQL Developer interface. In the Database Explorer, a trigger named `TRG_CONT_FARA_SUPRAPUNERI` is being created. The trigger body includes a `WHEN NO_DATA_FOUND THEN NULL;` block and an `INSERT INTO` statement. The trigger is associated with a table named `ISTORIC_CONTRACTE`. A message window displays the following error:

```

ORA-20060: ARTISTUL 3 ARE CONTRACTE CU PERIOADE SUPRAPUSE.
ORA-06512: at "STREAMING_USER.TRG_CONT_FARA_SUPRAPUNERI", line 14
ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_CONT_FARA_SUPRAPUNERI'
[2026-01-07 18:28:15] [72000][20060]
[2026-01-07 18:28:15] https://docs.oracle.com/error-help/db/ora-20060/
[2026-01-07 18:28:15] Position: 39

```

The Services pane shows a transaction named `cod.sql` with a duration of 115 ms.

The screenshot shows the Oracle SQL Developer interface. In the Database Explorer, a table named `ISTORIC_ABONAMENTE` is selected. A trigger named `TRG_LA_FAF` is highlighted. In the SQL editor, an `UPDATE` statement is being run:

```

UPDATE ISTORIC_CONTRACTE
SET DATA_INCEPERII = DATE '2018-01-01'
WHERE NR_CONTRACT = 2;

```

A message window displays the following error:

```

ORA-20060: ARTISTUL 1 ARE CONTRACTE CU PERIOADE SUPRAPUSE.
ORA-06512: at "STREAMING_USER.TRG_CONT_FARA_SUPRAPUNERI", line 14
ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_CONT_FARA_SUPRAPUNERI'
[2026-01-07 19:30:19] [72000][20060]
[2026-01-07 19:30:19] https://docs.oracle.com/error-help/db/ora-20060/
[2026-01-07 19:30:19] Position: 7

```

The Services pane shows a transaction named `cod.sql` with a duration of 88 ms.

- Un utilizator nu poate urmări un artist în perioade care se suprapun, implicit nu poate avea 2 urmăriri active pentru același artist.

```

CREATE OR REPLACE TRIGGER TRG_URM_FARA_SUPRAPUNERI
AFTER INSERT OR UPDATE ON URMARIRI
DECLARE
    v_u URMARIRI.ID_UTILIZATOR%TYPE;
    v_a URMARIRI.ID_ARTIST%TYPE;
BEGIN
    SELECT ID_UTILIZATOR, ID_ARTIST INTO v_u, v_a
    FROM (
        SELECT a.ID_UTILIZATOR, a.ID_ARTIST
        FROM URMARIRI a
        JOIN URMARIRI b ON a.ID_UTILIZATOR = b.ID_UTILIZATOR
        WHERE a.ID_ARTIST = v_a
        AND b.ID_ARTIST = v_u
        AND a.ID_UTILIZATOR <> b.ID_UTILIZATOR
    );
    IF v_u IS NOT NULL THEN
        DELETE FROM URMARIRI WHERE ID_UTILIZATOR = v_u AND ID_ARTIST = v_a;
    END IF;
END;

```

```

        AND a.ID_ARTIST      = b.ID_ARTIST
        AND a.DATA_URMARIRII < NVL(b.DATA_INCHEIERE, DATE '9999-12-31')
        AND b.DATA_URMARIRII < NVL(a.DATA_INCHEIERE, DATE '9999-12-31')
        AND a.DATA_URMARIRII <> b.DATA_URMARIRII
    FETCH FIRST 1 ROW ONLY
);

RAISE_APPLICATION_ERROR( -20042, 'UTILIZATORUL ' || v_u || ' ARE PERIOADE
DE URMARIRE SUPRAPUSE PENTRU ARTISTUL ' || v_a);

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        NULL;
END;
/

```

The screenshot shows an Oracle SQL developer interface. The code editor has a tab named "cod.sql" containing the trigger creation script. To the right, there are two panes: "STREAMING\_USER.console" showing the trigger definition and "STREAMING\_USER.URMARIRI" showing a table with three rows of data. The table has columns: ID\_UTILIZATOR, ID\_ARTIST, DATA\_URMARIRII, and DATA\_INCHEIERE. The data is as follows:

ID_UTILIZATOR	ID_ARTIST	DATA_URMARIRII	DATA_INCHEIERE
1	1	2019-05-01	<null>
2	1	2020-06-01	<null>
3	1	2021-07-01	<null>

The status bar at the bottom indicates [72000][20042] ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1 ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17 ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

The screenshot shows the same Oracle SQL developer interface after running the update statement. The code editor now includes an UPDATE statement:

```

        ...
        /
select * from URMARIRI where ID_UTILIZATOR=1;
UPDATE URMARIRI
SET DATA_INCHEIERE = NULL
WHERE ID_UTILIZATOR = 1
    AND ID_ARTIST = 1
    AND DATA_INCHEIERE IS NOT NULL;

```

The "STREAMING\_USER.URMARIRI" pane shows the updated data with the fourth row having a non-null value in the DATA\_INCHEIERE column. The status bar at the bottom indicates [72000][20042] ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1 ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17 ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

**Database Explorer**

```

@localhost
  - ISTORIC_ABONAMENTE
    - columns 4
    - keys 1
    - foreign keys 2
    - indexes 1
    - checks 1
    - triggers 4
      - TRG_IA_AF after insert, update
      - TRG_URM_FARA_SUPRAPUNERI
      - TRG_INIT_IA_BF before insert, update
      - TRG_MARK_IA_BFR before insert
  - ISTORIC_CONTRACTE
  - MELODI
  - REDARI
  - URMARIRI
  - UTILIZATORI
    - columns 6
    - keys 2
    - indexes 2

```

**Services**

- Database
  - @localhost
    - console 74 ms
    - cod.sql

**console [@localhost] cod.sql**

```

24   END;
25 /
26
27 select * from URMARIRI where ID_UTILIZATOR=1;
28
29 ? UPDATE URMARIRI
30 SET DATA_URMARIRII = DATE '2018-01-02'
31 WHERE ID_UTILIZATOR = 1
32   AND ID_ARTIST = 1
33   AND DATA_INCHEIERE IS NULL;

```

[72000][20042]

ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

**console - console**

ID_UTILIZATOR	ID_ARTIST	DATA_URMARIRII	DATA_INCHEIERE
1	1	1 2019-05-01	<null>
2	1	3 2020-06-01	<null>
3	1	4 2021-07-01	<null>
4	1	1 2018-01-01	2019-01-01

4 rows

**Full Message**

SET DATA\_URMARIRII = DATE '2018-01-02'  
WHERE ID\_UTILIZATOR = 1  
AND ID\_ARTIST = 1  
AND DATA\_INCHEIERE IS NULL

[2026-01-07 19:22:53] [72000][20042]  
[2026-01-07 19:22:53] ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
[2026-01-07 19:22:53] ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
[2026-01-07 19:22:53] ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI'  
[2026-01-07 19:22:53] [2026-01-07 19:22:53] <https://docs.oracle.com/error-help/db/ora-20042/>  
[2026-01-07 19:22:53] Position: 7

**Database Explorer**

```

@localhost
  - ISTORIC_ABONAMENTE
    - columns 4
    - keys 1
    - foreign keys 2
    - indexes 1
    - checks 1
    - triggers 4
      - TRG_IA_AF after insert, update
      - TRG_URM_FARA_SUPRAPUNERI
      - TRG_INIT_IA_BF before insert, update
      - TRG_MARK_IA_BFR before insert
  - ISTORIC_CONTRACTE
  - MELODI
  - REDARI
  - URMARIRI
  - UTILIZATORI
    - columns 6
    - keys 2
    - indexes 2

```

**Services**

- Database
  - @localhost
    - console 48 ms
    - cod.sql

**console [@localhost] cod.sql**

```

1 CREATE OR REPLACE TRIGGER TRG_URM_FARA_SUPRAPUNERI
2 WHEN NO_DATA_FOUND THEN
3   NULL;
4 /
5
6 select * from URMARIRI where ID_UTILIZATOR=1;
7
8 ? INSERT INTO URMARIRI
9 SELECT ID_UTILIZATOR 1, ID_ARTIST 1,
10       DATA_URMARIRII DATE '2020-01-01',
11       DATA_INCHEIERE DATE '2021-01-01', NOTIFICARI 'Y' FROM dual
12 UNION ALL
13 SELECT ID_UTILIZATOR 1, ID_ARTIST 1,
14       DATA_URMARIRII DATE '2020-06-01', DATA_INCHEIERE DATE '2020-12-31', NOTIFICARI 'Y' FROM dual;

```

[72000][20042]

ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

**console - console**

ID_UTILIZATOR	ID_ARTIST	DATA_URMARIRII	DATA_INCHEIERE
1	1	1 2019-05-01	<null>
2	1	3 2020-06-01	<null>
3	1	4 2021-07-01	<null>
4	1	1 2018-01-01	2019-01-01

2019-01-01

**Full Message**

ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

**Database Explorer**

```

@localhost
  - ISTORIC_ABONAMENTE
    - columns 4
    - keys 1
    - foreign keys 2
    - indexes 1
    - checks 1
    - triggers 4
      - TRG_IA_AF after insert, update
      - TRG_URM_FARA_SUPRAPUNERI
      - TRG_INIT_IA_BF before insert, update
      - TRG_MARK_IA_BFR before insert
  - ISTORIC_CONTRACTE
  - MELODI
  - REDARI
  - URMARIRI
  - UTILIZATORI
    - columns 6
    - keys 2
    - indexes 2

```

**Services**

- Database
  - @localhost
    - console 70 ms
    - cod.sql

**console [@localhost] cod.sql**

```

1 CREATE OR REPLACE TRIGGER TRG_URM_FARA_SUPRAPUNERI
2
3   );
4
5   RAISE_APPLICATION_ERROR( -20042, 'UTI';
6
7 EXCEPTION
8   WHEN NO_DATA_FOUND THEN
9     NULL;
10  END;
11 /
12
13 select * from URMARIRI where ID_UTILIZATOR=1;
14
15 ? INSERT INTO URMARIRI
16 VALUES ( ID_UTILIZATOR 1, ID_ARTIST 1, DATA_URMARIRII DATE '2020-01-01', DATA_INCHEIERE NULL, NOTIFICARI 'Y' );

```

[72000][20042]

ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

**console - console**

ID_UTILIZATOR	ID_ARTIST	DATA_URMARIRII	DATA_INCHEIERE
1	1	1 2019-05-01	<null>
2	1	3 2020-06-01	<null>
3	1	4 2021-07-01	<null>

3 rows

**Full Message**

ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI' ...

[2026-01-07 19:16:20] [72000][20042]  
[2026-01-07 19:16:20] ORA-20042: UTILIZATORUL 1 ARE PERIOADE DE URMARIRE SUPRAPUSE PENTRU ARTISTUL 1  
[2026-01-07 19:16:20] ORA-06512: at "STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI", line 17  
[2026-01-07 19:16:20] ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_FARA\_SUPRAPUNERI'  
[2026-01-07 19:16:20] [2026-01-07 19:16:20] <https://docs.oracle.com/error-help/db/ora-20042/>  
[2026-01-07 19:16:20] Position: 12

## 11. Definiți un *trigger* de tip LMD la nivel de linie. Declanșați *trigger*-ul.

- Durata unei redări nu poate depăși durata melodiei redăte.

```
CREATE OR REPLACE TRIGGER TRG_REDARI_DURATA
BEFORE INSERT OR UPDATE ON REDARI
FOR EACH ROW
DECLARE
    V_DURATA_MELODIE NUMBER;
BEGIN
    SELECT DURATA INTO V_DURATA_MELODIE
    FROM MELODII
    WHERE ID_MELODIE = :NEW.ID_MELODIE;

    IF :NEW.DURATA_REDARE > V_DURATA_MELODIE THEN
        RAISE_APPLICATION_ERROR(-20010, 'DURATA REDARII NU POATE DEPASI
DURATA MELODIEI.');
    END IF;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20099, 'Melodie inexistentă.');
END;
/
```

The screenshot shows the Oracle SQL Developer interface. On the left, the Database Explorer pane displays the schema structure, including triggers under the REDARI table. One trigger, TRG\_REDARI\_DURATA, is selected. The code for this trigger is visible in the central editor area. In the bottom-left, the Services pane shows a connection to the localhost database. The bottom-right corner displays the system status: 3:13, CRLF, UTF-8, 4 spaces.

```
select * from MELODII where ID_MELODIE=1;
```

```
INSERT INTO REDARI
VALUES (1, 1, DATE '2023-01-02', 300, 'MOBIL');
```

ID_MELODIE	NUME	DURATA	DATA_LANSARE	ID_ALBUM	ID_GEN
1	START	210	2018-06-01		1

[72000][20010]  
ORA-20010: DURATA REDARII NU POATE DEPASI DURATA MELODIEI.  
ORA-06512: at "STREAMING\_USER.TRG\_REDARI\_DURATA", line 9  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_REDARI\_DURATA' ...

[2026-01-07 19:45:51] 1 row retrieved starting from 1 in 410 ms (execution: 4 ms, fetching: 406 ms)  
[2026-01-07 19:45:21] STREAMING\_USER> INSERT INTO REDARI  
VALUES (1, 1, DATE '2023-01-02', 300, 'MOBIL')  
[2026-01-07 19:45:21] [72000][20010]  
[2026-01-07 19:45:21] ORA-20010: DURATA REDARII NU POATE DEPASI DURATA MELODIEI.  
[2026-01-07 19:45:21] ORA-06512: at "STREAMING\_USER.TRG\_REDARI\_DURATA", line 9  
[2026-01-07 19:45:21] ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_REDARI\_DURATA'  
[2026-01-07 19:45:21] https://docs.oracle.com/error-help/db/ora-20010/  
[2026-01-07 19:45:21] Position: 12

```

Database Explorer
@localhost
  triggers 5
    TRG_LA_AF after insert, update
    TRG_LA_DATA_NASTERE before
    TRG_LA_FARA_SUPRAPUNERI
    TRG_INIT_LA_BF before insert, update
    TRG_MARK_LA_BFR before insert
  ISTORIC_CONTRACTE
  MELODII
  REDARI
    columns 5
    keys 1
    foreign keys 2
    indexes 1
    checks 1
    triggers 2
  URMARIRI
  UTILIZATORI

Services
Database Explorer
@localhost
  Database
    console 52 ms
  cod.sql

Output
[2020-01-07 19:46:29] [ZU001] [ZU001]
[2020-01-07 19:46:29] ORA-20010: DURATA REDARII NU poate depasi DURATA MELODIEI.
[2020-01-07 19:46:29] ORA-06512: at "STREAMING_USER.TRG_REDARI_DURATA", line 9
[2020-01-07 19:46:29] ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_REDARI_DURATA'
[2020-01-07 19:46:29] https://docs.oracle.com/error-help/db/ora-20010/
[2020-01-07 19:46:29] Position: 7

```

```

Database Explorer
@localhost
  triggers 5
    TRG_LA_AF after insert, update
    TRG_LA_DATA_NASTERE before
    TRG_LA_FARA_SUPRAPUNERI
    TRG_INIT_LA_BF before insert, update
    TRG_MARK_LA_BFR before insert
  ISTORIC_CONTRACTE
  MELODII
  REDARI
    columns 5
    keys 1
    foreign keys 2
    indexes 1
    checks 1
    triggers 2
  URMARIRI
  UTILIZATORI

Services
Database Explorer
@localhost
  Database
    console 55 ms
  cod.sql

Output
[2020-01-07 19:48:02] [ZU001] [ZU001]
[2020-01-07 19:48:02] ORA-20010: DURATA REDARII NU poate depasi DURATA MELODIEI.
[2020-01-07 19:48:02] ORA-06512: at "STREAMING_USER.TRG_REDARI_DURATA", line 9
[2020-01-07 19:48:02] ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_REDARI_DURATA'
[2020-01-07 19:48:02] https://docs.oracle.com/error-help/db/ora-20010/
[2020-01-07 19:48:02] Position: 12

```

- Data urmăririi unui artist nu poate fi anterioară datei de naștere a utilizatorului.

```

CREATE OR REPLACE TRIGGER TRG_URM_DATA_NASTERE
BEFORE INSERT OR UPDATE ON URMARIRI
FOR EACH ROW
DECLARE
    V_DATA_NASTERII DATE;
BEGIN
    SELECT DATA_NASTERE INTO V_DATA_NASTERII
    FROM UTILIZATORI
    WHERE ID_UTILIZATOR = :NEW.ID_UTILIZATOR;

```

```

    IF :NEW.DATA_URMARIRII < V_DATA_NASTERII THEN
        RAISE_APPLICATION_ERROR(-20020, 'DATA URMARIRII NU POATE FI
ANTERIOARA DATEI NASTERII UTILIZATORULUI.');
    END IF;
END;
/

```

The screenshot shows the Oracle SQL Developer interface. In the Database Explorer, under the triggers node for the STREAMING\_USER schema, there is a trigger named TRG\_URM\_DATA\_NASTERE. In the code editor, a query is being run:

```

select * from UTILIZATORI where ID_UTILIZATOR=1;

```

In the SQL console, the following command is being executed:

```

INSERT INTO URMARIRI
VALUES (1, 1, DATE '2000-05-01', NULL, 'Y');

```

The Services pane shows a database connection named 'console' at localhost. The Output pane displays the following error message:

```

[72000][20020]
ORA-20020: DATA URMARIRII NU POATE FI ANTERIOARA DATEI NASTERII UTILIZATORULUI.
ORA-06512: at "STREAMING_USER.TRG_URM_DATA_NASTERE", line 9
ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_URM_DATA_NASTERE' ...

```

This screenshot shows the same Oracle SQL Developer environment. The Database Explorer and Services pane are identical to the previous screenshot. In the code editor, a different query is being run:

```

select * from UTILIZATORI where ID_UTILIZATOR=1;

```

In the SQL console, the following command is being executed:

```

UPDATE URMARIRI
SET DATA_URMARIRII = DATE '2001-01-01'
WHERE ID_UTILIZATOR = 1
AND ID_ARTIST = 1
AND DATA_URMARIRII = DATE '2019-05-01';

```

The Services pane shows a database connection named 'console' at localhost. The Output pane displays the following error message:

```

[72000][20020]
ORA-20020: DATA URMARIRII NU POATE FI ANTERIOARA DATEI NASTERII UTILIZATORULUI.
ORA-06512: at "STREAMING_USER.TRG_URM_DATA_NASTERE", line 9
ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_URM_DATA_NASTERE' ...

```

```

1 select * from UTILIZATORI where ID_UTILIZATOR=1;
2
3 ① INSERT INTO URMARIRI
4     SELECT ID_UTILIZATOR_1, ID_ARTIST_1, DATA_URMARIRI DATE '2018-01-02', DATA_INCHEIERE NULL, NOTIFICARI 'Y' FROM dual
5 UNION ALL
6     SELECT ID_UTILIZATOR_1, ID_ARTIST_2, DATA_URMARIRI DATE '2000-01-01', DATA_INCHEIERE NULL, NOTIFICARI 'Y' FROM dual;
7
8
9
10
11

```

ID_UTILIZATOR	NUME	PRENUME	EMAIL	DATA_NASTERE	TARA
1	WHITE	DANIELLE	danielle@email.com	2005-12-10	UK

[72000][20020]  
ORA-20020: DATA URMARIRI NU POATE FI ANTERIOARA DATEI NASTERII UTILIZATORULUI.  
ORA-06512: at "STREAMING\_USER.TRG\_URM\_DATA\_NASTERE", line 9  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_URM\_DATA\_NASTERE' ...

- Daat începerii unui abonament nu este anterioară datei de naștere a utilizatorului.

```

CREATE OR REPLACE TRIGGER TRG_IA_DATA_NASTERE
BEFORE INSERT OR UPDATE ON ISTORIC_ABONAMENTE
FOR EACH ROW
DECLARE
    V_DATA_NASTERII DATE;
BEGIN
    SELECT DATA_NASTERE INTO V_DATA_NASTERII
    FROM UTILIZATORI
    WHERE ID_UTILIZATOR = :NEW.ID_UTILIZATOR;

    IF :NEW.DATA_INCEPERII < V_DATA_NASTERII THEN
        RAISE_APPLICATION_ERROR(-20021, 'DATA INCEPERII ABONAMENTULUI NU
POATE FI ANTERIOARA DATEI NASTERII UTILIZATORULUI.');
    END IF;
END;
/

```

Database Explorer

```
console [@localhost] cod.sql
1 select * from UTILIZATORI where ID_UTILIZATOR=1;
2
3 INSERT INTO ISTORIC_ABONAMENTE
4 VALUES (1, 3, DATE '2000-01-01', NULL);
5
6
7 [72000][20021]
8 ORA-20021: DATA INCEPERII ABONAMENTULUI NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.
9 ORA-06512: at "STREAMING_USER.TRG_IA_DATA_NASTERE", line 9
10 ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_IA_DATA_NASTERE' ...
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259]
```

Services

```
[2026-01-07 19:57:58] [72000][20021]
[2026-01-07 19:57:58] ORA-20021: DATA INCEPERII ABONAMENTULUI NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.
[2026-01-07 19:57:58] ORA-06512: at "STREAMING_USER.TRG_IA_DATA_NASTERE", line 9
[2026-01-07 19:57:58] ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_IA_DATA_NASTERE'
[2026-01-07 19:57:58] https://docs.oracle.com/error-help/db/ora-20021/
[2026-01-07 19:57:58] Position: 12
```

Database Explorer

```
console [@localhost] cod.sql
1 select * from UTILIZATORI where ID_UTILIZATOR=1;
2
3 UPDATE ISTORIC_ABONAMENTE
4 SET DATA_INCEPERII = DATE '2001-01-01'
5 WHERE ID_UTILIZATOR = 1
6 AND ID_ABONAMENT = 3;
7
8
9 [72000][20021]
10 ORA-20021: DATA INCEPERII ABONAMENTULUI NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.
11 ORA-06512: at "STREAMING_USER.TRG_IA_DATA_NASTERE", line 9
12 ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_IA_DATA_NASTERE' ...
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
229]
```

Services

```
[2026-01-07 19:58:57] [72000][20021]
[2026-01-07 19:58:57] ORA-20021: DATA INCEPERII ABONAMENTULUI NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.
[2026-01-07 19:58:57] ORA-06512: at "STREAMING_USER.TRG_IA_DATA_NASTERE", line 9
[2026-01-07 19:58:57] ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_IA_DATA_NASTERE'
[2026-01-07 19:58:57] https://docs.oracle.com/error-help/db/ora-20021/
[2026-01-07 19:58:57] Position: 7
```

Database Consoles > @localhost > console [@localhost]

The screenshot shows the Oracle SQL Developer interface. The Database Explorer on the left lists schemas like @localhost, triggers, and tables such as ISTORIC\_CONTRACTE, MELODII, REDARI, URMARIRI, and UTILIZATORI. The central workspace has two tabs: 'console [@localhost]' and 'cod.sql'. The 'cod.sql' tab contains a script with several SQL statements, including triggers and a select query. The 'Output' tab at the bottom shows the execution results and errors, including ORA-20021, ORA-06512, and ORA-04088 errors related to trigger execution. The Services panel on the right shows a database connection to 'localhost'.

```
1  Select * from UTILIZATORI where ID_UTILIZATOR=1;
2
3  INSERT INTO ISTORIC_ABONAMENTE
4    SELECT ID_UTILIZATOR 1, ID_ABONAMENT_1, DATA_INCEPERI DATE '2018-01-01', DATA_ANULARE NULL FROM dual
5  UNION ALL
6  SELECT ID_UTILIZATOR 1, ID_ABONAMENT_2, DATA_INCEPERI DATE '2000-01-01', DATA_ANULARE DATE '2008-12-01' FROM dual;
```

ID_UTILIZATOR	NUME	PRENUME	EMAIL	DATA_NASTERE
1	WHITE	DANIELLE	white.danielle@email.com	2005-12-10

[72000][20021]  
ORA-20021: DATA INCEPERII ABONAMENTULUI NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.  
ORA-06512: at "STREAMING\_USER.TRG\_IA\_DATA\_NASTERE", line 9  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_IA\_DATA\_NASTERE' ...

Full Message

```
[2026-01-07 20:00:17] (72000)[20021]
[2026-01-07 20:00:17] ORA-20021: DATA INCEPERII ABONAMENTULUI NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.
[2026-01-07 20:00:17] ORA-06512: at "STREAMING_USER.TRG_IA_DATA_NASTERE", line 9
[2026-01-07 20:00:17] ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_IA_DATA_NASTERE'
[2026-01-07 20:00:17] https://docs.oracle.com/error-help/db/ora-20021/
[2026-01-07 20:00:17] Position: 12
```

- Data unei redări nu este anterioară datei de naștere a utilizatorului și nici datei de lansare a melodiei.

```
CREATE OR REPLACE TRIGGER TRG_REDARI_DATE
BEFORE INSERT OR UPDATE ON REDARI
FOR EACH ROW
DECLARE
    V_DATA_NASTERE DATE;
    V_DATA_LANSARE DATE;
BEGIN
    SELECT DATA_NASTERE INTO V_DATA_NASTERE
    FROM UTILIZATORI
    WHERE ID_UTILIZATOR = :NEW.ID_UTILIZATOR;

    SELECT DATA_LANSARE INTO V_DATA_LANSARE
    FROM MELODII
    WHERE ID_MELODIE = :NEW.ID_MELODIE;

    IF :NEW.DATA_REDARE < V_DATA_NASTERE THEN
        RAISE_APPLICATION_ERROR(-20030, 'DATA REDARII NU POATE FI
ANTERIOARA DATEI NASTERII UTILIZATORULUI.');
    END IF;

    IF V_DATA_LANSARE IS NOT NULL
        AND :NEW.DATA_REDARE < V_DATA_LANSARE THEN
        RAISE_APPLICATION_ERROR(-20031, 'DATA REDARII NU POATE FI
ANTERIOARA DATEI DE LANSARE A MELODIEI.');
    END IF;
END;
/WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20099, 'Utilizator sau melodie inexistentă.');
```

```

END;
/

```

Database Explorer

- @localhost
  - triggers 5
    - TRG\_IA\_AF after insert, update
    - TRG\_IA\_DATA\_NASTERE before
    - TRG\_IA\_FARA\_SUPRANUERI
    - TRG\_INIT\_IA\_BF before insert, update
    - TRG\_MARKIA\_BFR before insert
  - ISTORIC\_CONTRACTE
  - MELODII
  - REDARI
  - columns 5
  - keys 1
  - foreign keys 2
  - indexes 1
  - checks 1
  - triggers 2
  - URMARIRI
  - UTILIZATORI

console [localhost] x cod.sql

```

1 select * from UTILIZATORI where ID_UTILIZATOR=1;
2 select * from MELODII where ID_MELODIE =1;
3
4 ① INSERT INTO REDARI
5   VALUES (ID_UTILIZATOR_1, ID_MELODIE_1, DATA_REDARE_DATE '2004-06-02', DURATA_REDARE_180, DISPOZITIV 'MOBIL');
6

```

[72000][20030]

ORA-20030: DATA REDARII NU poate fi ANTERIOARA DATEI NASTERII UTILIZATORULUI.  
ORA-06512: at "STREAMING\_USER.TRG\_REDARDATE", line 14  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_REDARDATE' ...

Full Message x

Services

File - console

AMING\_USER.MELODII x

MELODIE	NUME	DURATA	DATA_LANSARE
1 START		210	2018-06-01

1 row ▾

STREAMING\_USER.UTILIZATORI x

ID_UTILIZATOR	NUME	PRENUME	EMAIL	DATA_NASTERE
1	WHITE	DANIELLE	white.danielle@email.com	2005-12-10

1 row ▾

Database Consoles > @localhost > console [localhost]

4:13 CRLF UTF-8 4 spaces

```

END;
/

```

Database Explorer

- @localhost
  - triggers 5
    - TRG\_IA\_AF after insert, update
    - TRG\_IA\_DATA\_NASTERE before
    - TRG\_IA\_FARA\_SUPRANUERI
    - TRG\_INIT\_IA\_BF before insert, update
    - TRG\_MARKIA\_BFR before insert
  - ISTORIC\_CONTRACTE
  - MELODII
  - REDARI
  - columns 5
  - keys 1
  - foreign keys 2
  - indexes 1
  - checks 1
  - triggers 2
  - URMARIRI
  - UTILIZATORI

console [localhost] x cod.sql

```

1 select * from UTILIZATORI where ID_UTILIZATOR=1;
2 select * from MELODII where ID_MELODIE =1;
3
4 ① INSERT INTO REDARI
5   VALUES (ID_UTILIZATOR_1, ID_MELODIE_1, DATA_REDARE_DATE '2009-06-02', DURATA_REDARE_180, DISPOZITIV 'MOBIL');
6

```

[72000][20031]

ORA-20031: DATA REDARII NU poate fi ANTERIOARA DATEI DE LANSARE A MELODIEI.  
ORA-06512: at "STREAMING\_USER.TRG\_REDARDATE", line 19  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_REDARDATE' ...

Full Message x

Services

File - console

AMING\_USER.MELODII x

MELODIE	NUME	DURATA	DATA_LANSARE
1 START		210	2018-06-01

1 row ▾

STREAMING\_USER.UTILIZATORI x

ID_UTILIZATOR	NUME	PRENUME	EMAIL	DATA_NASTERE
1	WHITE	DANIELLE	white.danielle@email.com	2005-12-10

1 row ▾

Database Consoles > @localhost > console [localhost]

4:13 CRLF UTF-8 4 spaces

The screenshot shows the Oracle SQL Developer interface. In the Database Explorer, under the @localhost connection, there is a triggers node. One of the triggers, TRG\_IC\_DATA\_NASTERE, has an error icon next to it. The error message is:

```

ORA-20030: DATA REDARII NU POATE FI ANTERIOARA DATEI DE LANSARE A MELODIEI.
ORA-06512: at "STREAMING_USER.TRG_REDARILDATE", line 19
ORA-04080: error during execution of trigger 'STREAMING_USER.TRG_REDARILDATE' ...

```

In the bottom right corner of the main window, there is a 'Full Message' button.

This screenshot is similar to the previous one, showing the same trigger error message:

```

ORA-20030: DATA REDARII NU POATE FI ANTERIOARA DATEI NASTERII UTILIZATORULUI.
ORA-06512: at "STREAMING_USER.TRG_REDARILDATE", line 14
ORA-04080: error during execution of trigger 'STREAMING_USER.TRG_REDARILDATE' ...

```

- Data începerii unui contract nu este anterioară datei de înființare a casei de discuri

```

CREATE OR REPLACE TRIGGER TRG_IC_DATA_INFIINTARE
BEFORE INSERT OR UPDATE ON ISTORIC_CONTRACTE
FOR EACH ROW
DECLARE
    V_DATA_INFIINTARE DATE;
BEGIN
    SELECT DATA_INFIINTARE INTO V_DATA_INFIINTARE
    FROM CASE_DISCURI

```

```

        WHERE ID_CASADISCURI = :NEW.ID_CASADISCURI;

        IF :NEW.DATA_INCEPERII < V_DATA_INFINTARE THEN
            RAISE_APPLICATION_ERROR(-20022, 'DATA INCEPERII CONTRACTULUI NU
POATE FI ANTERIORA DATI INFINTARII CASEI DE DISCURI.');
        END IF;
END;
/

```

258  
259     SELECT \* FROM CASE\_DISCURI;  
260  
261  
262     INSERT INTO ISTORIC\_CONTRACTE  
263       VALUES ( NR\_CONTRACT SEQ\_CONTRACTE.NEXTVAL, ID\_ARTIST 1, ID\_CASADISCURI 4, DATA\_INCEPERII DATE '2005-12-31', DATA\_INCHIEIRE NULL);  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276

console - console

STREAMING\_USER.CASE\_DISCURI

ID_CASADISCURI	NUME	DATA_INFINTARE	TARA
4	RO-MUSIC	2006-05-20	ROMANIA
5	ROYAL-MUSIC	2004-03-01	UK
6	ANONYMOUS	2020-02-20	UK

[72000][20022] Full Message  
ORA-20022: DATA INCEPERII CONTRACTULUI NU POATE FI ANTERIORA DATI INFINTARII CASEI DE DISCURI.  
ORA-06512: at "STREAMING\_USER.TRG\_IC\_DATA\_INFINTARE", line 9  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_IC\_DATA\_INFINTARE' ...

258  
259     SELECT \* FROM CASE\_DISCURI;  
260  
261  
262     UPDATE ISTORIC\_CONTRACTE  
263       SET DATA\_INCEPERII = DATE '2000-01-01'  
264       WHERE ID\_CASADISCURI = 4  
265       AND ROWNUM = 1;

Output

[2026-01-08 22:49:55] https://docs.oracle.com/error-help/db/ora-20022/  
[2026-01-08 22:49:55] Position: 39

console - console

STREAMING\_USER.CASE\_DISCURI

ID_CASADISCURI	NUME	DATA_INFINTARE	TARA
4	RO-MUSIC	2006-05-20	ROMANIA
5	ROYAL-MUSIC	2004-03-01	UK
6	ANONYMOUS	2020-02-20	UK

[72000][20022] Full Message  
ORA-20022: DATA INCEPERII CONTRACTULUI NU POATE FI ANTERIORA DATI INFINTARII CASEI DE DISCURI.  
ORA-06512: at "STREAMING\_USER.TRG\_IC\_DATA\_INFINTARE", line 9  
ORA-04088: error during execution of trigger 'STREAMING\_USER.TRG\_IC\_DATA\_INFINTARE' ...

- O melodie trebuie lansata impreună sau după albumul aferent

```

CREATE OR REPLACE TRIGGER TRG_MEL_DATA
BEFORE INSERT OR UPDATE ON MELODII
FOR EACH ROW
DECLARE
    v_data_alb ALBUME.DATA_LANSARE%type;
BEGIN
    SELECT DATA_LANSARE INTO v_data_alb
    FROM ALBUME

```

```

        WHERE ALBUME.ID_ALBUM = :NEW.ID_ALBUM;

        IF :NEW.DATA_LANSARE < v_data_alb THEN
            RAISE_APPLICATION_ERROR(-20029, 'DATA DE LANSARE A MELODIEI NU
POATE FI ANTERIOARA DATEI ALBUMULUI');
        END IF;

end;

```

The screenshot shows a database session in Oracle SQL Developer. The Database Explorer on the left lists several tables: ABONAMENTE, ALBUME, ARTISTI, CASE\_DISCURSI, GENURLMUZICALE, ISTORIC\_ABONAMENTE, ISTORIC\_CONTRAC, MELODII, and URMARIRI. The central area has a 'console' tab open, showing the following code and its execution:

```

    select * from ALBUME;
    INSERT INTO MELODII
    VALUES (SEQ_MELODII.NEXTVAL, 'TEST', 210, DATE '2010-06-01', 1, 1);

```

The output window shows the error:

```

[72000][20029]
ORA-20029: DATA DE LANSARE A MELODIEI NU POATE FI ANTERIOARA DATEI ALBUMULUI
ORA-06512: at "STREAMING_USER.TRG_MEL_DATA", line 9
ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_MEL_DATA' ...

```

The right side of the interface displays two tables: STREAMING\_USER.ALBUME and STREAMING\_USER.MELODII. The ALBUME table contains 5 rows with data like ID\_ALBUM 1, DENUMIRE PRIMUL\_PAS, DATA\_LANSARE 2018-06-01, and ID\_ARTIST 1. The MELODII table contains 5 rows with data like ID\_MELODIE 1, NUME START, DURATA 210, DATA\_LANSARE 2018-06-01, and ID\_ALBUM 1.

This screenshot shows the same database session after an update to the MELODII table. The code executed is:

```

    select * from ALBUME;
    select * from MELODII;
    update MELODII
    set ID_ALBUM = 2
    where ID_MELODIE = 1;

```

The output window shows the error:

```

[72000][20029]
ORA-20029: DATA DE LANSARE A MELODIEI NU POATE FI ANTERIOARA DATEI ALBUMULUI
ORA-06512: at "STREAMING_USER.TRG_MEL_DATA", line 9
ORA-04088: error during execution of trigger 'STREAMING_USER.TRG_MEL_DATA' ...

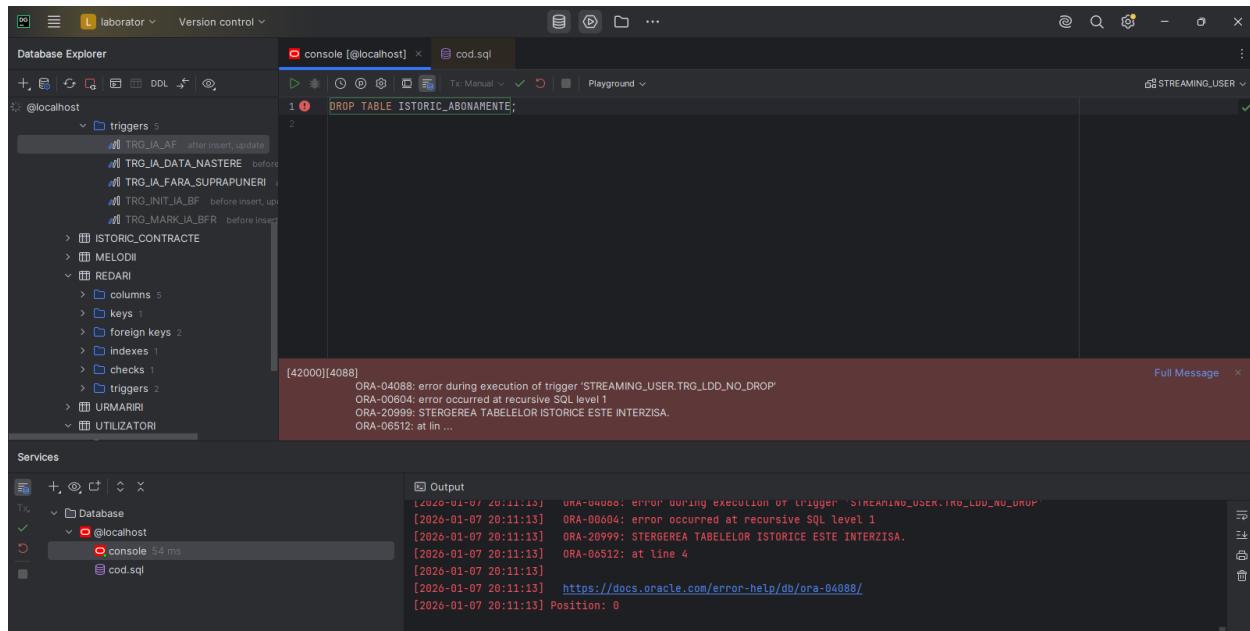
```

The right side of the interface shows the same tables as before. The ALBUME table now has 2 rows. The MELODII table has been updated with ID\_ALBUM 2 for the first row.

## 12. Definiți un *trigger* de tip LDD. Declanșați *trigger*-ul.

A fost definit un trigger de tip LDD care previne ștergerea tabelelor istorice din schemă, asigurând păstrarea informațiilor istorice ale aplicației.

```
CREATE OR REPLACE TRIGGER TRG_LDD_NO_DROP
BEFORE DROP ON SCHEMA
BEGIN
    IF ORA_DICT_OBJ_NAME IN ('ISTORIC_ABONAMENTE', 'URMARIRI',
'ISTORIC_CONTRACTE')
    THEN
        RAISE_APPLICATION_ERROR(-20999, 'STERGEREA TABELELOR ISTORICE ESTE
INTERZISA.');
    END IF;
END;
/
```



## 13.Cerintă optională

Pachetul PKG\_ABONAMENTE a fost conceput pentru a implementa și controla operațiile principale legate de abonamentele utilizatorilor din platforma de muzica.

- **Verificarea existenței unui abonament activ**

Funcția are\_abonament\_activ determină dacă un utilizator are un abonament activ, validând în prealabil existența utilizatorului.

- **Recomandarea unui tip de abonament**

Functia recomanda\_abonament analizează varsta utilizatorului și opțiunea de partajare a contului pentru a returna un tip de abonament adecvat (PUPIL, STUDENT, STANDARD sau opțiuni de tip DUO/FAMILY).

- **Generarea statisticilor pentru abonamente**

Funcția statistici\_toate\_abonamentele realizează o analiză globală a abonamentelor active, calculând:

- categoria de vîrstă dominantă
- naționalitatea predominantă
- durata medie de ascultare per utilizator

Rezultatele sunt returnate sub forma unei colectii de tip tabel, iar abonamentele fără utilizatori activi sunt excluse automat din statistici.

- **Adăugarea unui abonament**

Procedura adauga\_abonament permite atribuirea unui abonament unui utilizator, validând separat existența utilizatorului și a abonamentului. În cazul în care utilizatorul are deja un abonament activ, acesta este inchis automat înainte de adăugarea noului abonament

- **Anularea unui abonament**

Procedura anuleaza\_abonament anulează abonamentul activ al unui utilizator.

```
create or replace package pkg_abonamente is

type t_stat_abon is record (
    id_abonament ABONAMENTE.ID_ABONAMENT%type,
    denumire ABONAMENTE.DENUMIRE%type,
    categorie_varsta varchar2(20),
    nationalitate varchar2(50),
    medie_ascultare number
);

type t_stat_tab is table of t_stat_abon;
```

```

function are_abonament_activ (p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type)
    return boolean;

function recomanda_abonament (p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type,p_imparte in char)
    return varchar2;

function statistici_toate_abonamentele
    return t_stat_tab;

procedure adauga_abonament ( p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type,
                                p_id_abonament in ABONAMENTE.ID_ABONAMENT%type, p_data_start in date);

procedure anuleaza_abonament (p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type, p_data_anulare in date);

end pkg_abonamente;
/

create or replace package body pkg_abonamente is

function are_abonament_activ (p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type) return boolean is
    v_cnt number;
    v_test number;
begin
    begin
        select 1 into v_test
        from UTILIZATORI
        where ID_UTILIZATOR = p_id_utilizator;
    exception
        when no_data_found then
            raise_application_error(-20211, 'Utilizator inexistent');
    end;

    select count(*) into v_cnt
    from ISTORIC_ABONAMENTE
    where ID_UTILIZATOR=p_id_utilizator
        and DATA_ANULARE is null;

    return v_cnt > 0;
end are_abonament_activ;

function recomanda_abonament ( p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type, p_imparte in char)
return varchar2 is
    v_data_nastere date;
    v_varsta number;
begin
    select DATA_NASTERE into v_data_nastere
    from UTILIZATORI
    where ID_UTILIZATOR=p_id_utilizator;

    v_varsta := trunc(months_between(sysdate, v_data_nastere) / 12);

```

```

if p_imparte = 'Y' then
    return 'DUO, FAMILY';
else
    if v_varsta < 18 then
        return 'PUPIL';
    elsif v_varsta <= 25 then
        return 'STUDENT';
    else
        return 'STANDARD';
    end if;
end if;

exception
when no_data_found then
    raise_application_error(-20211, 'Utilizator inexistent');
end recomanda_abonament;

function statistici_toate_abonamentele return t_stat_tab is
v_rez t_stat_tab := t_stat_tab();

v_sub18 number;
v_18_25 number;
v_26_45 number;
v_45plus number;

type t_nat_cnt is table of number index by varchar2(50);
v_nat t_nat_cnt;

v_total_secunde number;
v_nr_utilizatori number;

v_varsta number;
v_cat varchar2(20);
v_nat_maj varchar2(50);
v_max number;
k varchar2(50);

begin
for a in (select ID_ABONAMENT, DENUMIRE from ABONAMENTE) loop

    v_nat.delete;

    v_sub18 := 0;
    v_18_25 := 0;
    v_26_45 := 0;
    v_45plus := 0;

    v_total_secunde := 0;
    v_nr_utilizatori := 0;
    v_nat_maj := 'N/A';

    for r in (

```

```

select IA.ID_UTILIZATOR,
       U.DATA_NASTERE,
       U.TARA,
       IA.DATA_INCEPERII,
       nvl((IA.DATA_ANULARE, sysdate) DATA_STOP
from ISTORIC_ABONAMENTE IA
join UTILIZATORI U on U.ID_UTILIZATOR=IA.ID_UTILIZATOR
where IA.ID_ABONAMENT=a.ID_ABONAMENT
      and IA.DATA_ANULARE is null
) loop
  v_nr_utilizatori := v_nr_utilizatori + 1;

  v_varsta := trunc(months_between(sysdate, r.DATA_NASTERE) / 12);

  if v_varsta < 18 then
    v_sub18 := v_sub18 + 1;
  elsif v_varsta <= 25 then
    v_18_25 := v_18_25 + 1;
  elsif v_varsta <= 45 then
    v_26_45 := v_26_45 + 1;
  else
    v_45plus := v_45plus + 1;
  end if;

  if v_nat.exists(r.TARA) then
    v_nat(r.TARA) := v_nat(r.TARA) + 1;
  else
    v_nat(r.TARA) := 1;
  end if;

begin
  select nvl(sum(R2.DURATA_REDARE), 0) into v_max
  from REDARI R2
  where R2.ID_UTILIZATOR=r.ID_UTILIZATOR
        and R2.DATA_REDARE between r.DATA_INCEPERII and r.DATA_STOP;

  if v_max is null then
    v_max := 0;
  end if;
exception
  when no_data_found then
    v_max := 0;
end;

  v_total_secunde := v_total_secunde + v_max;
end loop;

if v_nr_utilizatori = 0 then
  continue;
end if;

v_cat :=
  case

```

```

when v_sub18 >= greatest(v_18_25, v_26_45, v_45plus) then 'SUB_18'
when v_18_25 >= greatest(v_sub18, v_26_45, v_45plus) then '18_25'
when v_26_45 >= greatest(v_sub18, v_18_25, v_45plus) then '26_45'
else 'PESTE_45'
end;

v_max := -1;
k := v_nat.first;
while k is not null loop
  if v_nat(k) > v_max then
    v_max := v_nat(k);
    v_nat_maj := k;
  end if;
  k := v_nat.next(k);
end loop;

v_rez.extend;
v_rez(v_rez.last) := t_stat_abon(
  a.ID_ABONAMENT,
  a.DENUMIRE,
  v_cat,
  v_nat_maj,
  round(v_total_secunde / v_nr_utilizatori, 2)
);
end loop;

return v_rez;

end statistici_toate_abonamentele;

procedure adauga_abonament (p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type, p_id_abonament in ABONAMENTE.ID_ABONAMENT%type,
                           p_data_start in date) is
  v_test number;
begin
  begin
    select 1 into v_test
    from UTILIZATORI
    where ID_UTILIZATOR = p_id_utilizator;
  exception
    when no_data_found then
      raise_application_error(-20211, 'Utilizator inexistent');
  end;

  begin
    select 1 into v_test
    from ABONAMENTE
    where ID_ABONAMENT = p_id_abonament;
  exception
    when no_data_found then
      raise_application_error(-20212,'Abonament inexistent');
  end;

```

```

update ISTORIC_ABONAMENTE
set DATA_ANULARE=p_data_start - 1
where ID_UTILIZATOR=p_id_utilizator
and DATA_ANULARE is null;

insert into ISTORIC_ABONAMENTE
values (p_id_utilizator, p_id_abonament, p_data_start, null);
end adauga_abonament;


procedure anuleaza_abonament (
    p_id_utilizator in UTILIZATORI.ID_UTILIZATOR%type,
    p_data_anulare in date
) is
    v_test number;
begin
    begin
        select 1 into v_test
        from UTILIZATORI
        where ID_UTILIZATOR = p_id_utilizator;
    exception
        when no_data_found then
            raise_application_error(-20211, 'Utilizator inexistent');
    end;

    update ISTORIC_ABONAMENTE
    set DATA_ANULARE=p_data_anulare
    where ID_UTILIZATOR=p_id_utilizator
    and DATA_ANULARE is null;
end anuleaza_abonament;

end pkg_abonamente;
/
-----  

declare
    v_ok boolean;
begin
    v_ok := pkg_abonamente.are_abonament_activ(1);
    if v_ok then
        dbms_output.put_line('Utilizatorul are abonament activ.');
    else
        dbms_output.put_line('Utilizatorul NU are abonament activ.');
    end
    if;
end;
/
select pkg_abonamente.recomanda_abonament(333, 'N')
from dual;

declare

```

```

v_stat pkg_abonamente.t_stat_tab;
begin
  v_stat := pkg_abonamente.statistici_toate_abonamentele;

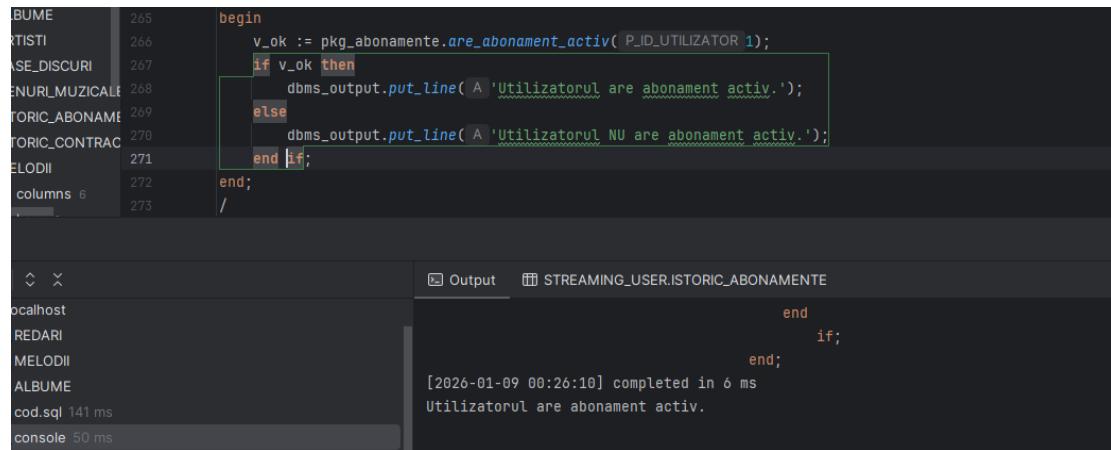
  if v_stat.count = 0 then
    dbms_output.put_line('Nu exista abonamente active.');
  end if;

  for i in 1 .. v_stat.count loop
    dbms_output.put_line(
      'Abonament: ' || v_stat(i).denumire ||
      ' | Varsta dominanta: ' || v_stat(i).categorie_varsta ||
      ' | Nationalitate dominanta: ' || v_stat(i).nationalitate ||
      ' | Medie secunde: ' || v_stat(i).medie_ascultare
    );
  end loop;
end;
/

begin
  pkg_abonamente.adauga_abonament(7,6, date '2025-01-01');
end;
/
select *
from ISTORIC_ABONAMENTE
where ID_UTILIZATOR = 7
order by DATA_INCEPERII;

begin
  pkg_abonamente.anuleaza_abonament( 7,date '2025-03-01');
end;
/

```



The screenshot shows the Oracle SQL Developer interface with the following details:

- Code Area:** Displays the PL/SQL code from the previous block.
- Output Area:** Shows the execution results of the code. It includes the following output:
 

```

localhost
REDARI
MELODII
ALBUME
cod.sql 141 ms
console 50 ms
          
```

```

[2026-01-09 00:26:10] completed in 6 ms
Utilizatorul are abonament activ.
          
```

```

REDARI          273   /
URMARIRI       274
UTILIZATORI    275 ✓  select pkg_abonamente.recomanda_abonament( P_ID_UTILIZATOR 1, P_IMPARTE 'N') as recomandare
                     from dual;
276
277
278
279
280
281

```

Output: recomandare:(VARCHAR2 as varchar2) ×

	RECOMANDARE
1	STUDENT

REDARI  
MELODII  
ALBUME  
UTILIZATORI  
cod.sql 141 ms

```

triggers 1
REDARI          277
URMARIRI       278
UTILIZATORI    279 ✓  declare
                     v_stat pkg_abonamente.t_stat_tab;
280 begin
281   v_stat := pkg_abonamente.statistici_toate_abonamentele;
282
283   if v_stat.count = 0 then
284     dbms_output.put_line( A 'Nu exista abonamente active.');
285   end if;
286
287

```

Output

[2026-01-09 00:29:11] completed in 4 ms

Abonament: PUPIL | Varsta dominanta: SUB\_18 | Nationalitate dominanta: FRANTA | Medie secunde: 312.5  
 Abonament: STUDENT | Varsta dominanta: 18\_25 | Nationalitate dominanta: UK | Medie secunde: 795  
 Abonament: FAMILY | Varsta dominanta: 26\_45 | Nationalitate dominanta: ROMANIA | Medie secunde: 900

REDARI  
MELODII  
ALBUME  
UTILIZATORI  
cod.sql 141 ms  
console 45 ms

```

95
96   begin
97     pkg_abonamente.adauga_abonament( P_ID_UTILIZATOR 7, P_ID_ABONAMENT 6, P_DATA_START date '2025-01-01');
98   end;
99
100 ✓  select *
101   from ISTORIC_ABONAMENTE
102  where ID_UTILIZATOR = 7
103  order by DATA_INCEPERII;
104
105 begin

```

Output: STREAMING\_USER.ISTORIC\_ABONAMENTE ×

ID_UTILIZATOR	ID_ABONAMENT	DATA_INCEPERII	DATA_ANULARE
1	7	1 2022-02-01	2024-12-31
2	7	6 2025-01-01	2025-03-01

CSV | ↴

2 rows | ⋮

console [@localhost] x cod.sql

```
293     end;
294   /
295
296 begin
297   pkg_abonamente.adauga_abonament( P_ID_UTILIZATOR 7, P_ID_ABONAMENT 6, P_DATA_START date '2025-01-01');
298 end;
299
300 ✓ select *
301 from ISTORIC_ABONAMENTE
302 where ID_UTILIZATOR = 7
303 order by DATA_INCEPERII;
```

Output STREAMING\_USER.ISTORIC\_ABONAMENTE

ID_UTILIZATOR	ID_ABONAMENT	DATA_INCEPERII	DATA_ANULARE
1	7	1 2022-02-01	2024-12-31
2	7	6 2025-01-01	<null>

ARI  
ODI  
UME  
ZATORI  
sql 141 ms  
ole 411 ms

laborator Version control

Database Explorer

console [@localhost] x cod.sql

```
298     end;
299   /
300 ✓ select *
301 from ISTORIC_ABONAMENTE
302 where ID_UTILIZATOR = 7
303 order by DATA_INCEPERII;
304
305 begin
306   pkg_abonamente.anuleaza_abonament( P_ID_UTILIZATOR 7, P_DATA_ANULARE date '2025-03-01');
307 end;
308 /
```

Services

REDARI  
MELODII  
ALBUME  
UTILIZATORI  
cod.sql 141 ms  
console 425 ms

Output STREAMING\_USER.ISTORIC\_ABONAMENTE

ID_UTILIZATOR	ID_ABONAMENT	DATA_INCEPERII	DATA_ANULARE
1	7	1 2022-02-01	2024-12-31
2	7	6 2025-01-01	2025-03-01