

ACADEMIA DE STUDII ECONOMICE

Facultatea: Cibernetică, Statistică și Informatică Economică

Specializarea: ***Informatică economică***

# **PROIECT**

## **BAZE DE DATE**

***Tema proiectului: Sistem informatic pentru  
gestiunea unui magazin online***

**Cadrul didactic coordonator:**

**Dobrița Gabriela**

**Student:**

**Bejan Ionut**

## *Introducere*

*Motivația alegerii acestei teme constă în importanța deosebită pe care o prezintă bazele de date în societatea contemporană, lucru datorat în principal eficienței și ușurinței de care acestea dispun în momentul în care o organizație dorește să gestioneze și să stocheze informații complexe. Un sistem informatic bine structurat asigură accesul rapid și corect la datele esențiale, oferind suport pentru luarea deciziilor în timp real.*

*Tema proiectului meu are în vedere realizarea unei gestiuni eficiente a activităților dintr-un magazin online, care reprezintă una dintre cele mai importante ramuri ale comerțului modern. Acest sistem are scopul de a organiza și centraliza informațiile despre produse, clienți, comenzi și stocuri, facilitând astfel administrarea operațiunilor zilnice. Ca și etapă inițială, am implementat o structură relațională a bazei de date, care stochează detalii esențiale despre produsele disponibile, comenzile plasate de clienți, precum și legăturile dintre acestea (chei primare și chei externe). Ulterior, această schemă a fost populată cu date relevante pentru simularea proceselor unui magazin online.*

*În etapa a doua a proiectului, am implementat diferite operații SQL, incluzând operații de actualizare, ștergere și interogare a datelor. Am folosit diverse funcții și expresii pentru a gestiona eficient informațiile despre produse (ex: calcularea prețurilor cu discount, stocuri minime), comenzi (ex: generarea automată a rapoartelor de vânzări) și clienți (ex: evidența comenzilor unui anumit client). De asemenea, am realizat interogări complexe, utilizând funcții SQL avansate, precum subinterogări, DECODE, CASE și operatorii UNION, INTERSECT și MINUS.*

*Pentru implementarea aplicației am utilizat mediul de dezvoltare oferit de ORACLE și anume SQL Developer, în care am rulat toate secvențele de cod și am proiectat schema bazei de date, inclusiv relațiile și constrângerile dintre tabele. Acest sistem informatic oferă un suport puternic pentru gestionarea eficientă și transparentă a activităților specifice unui magazin online.*

## Descrierea bazei de date

Tema aleasă are ca obiectiv organizarea, evidența și gestiunea produselor, comenzilor, clienților și altor elemente esențiale care aparțin unui magazin online. Această bază de date conține informații despre produse (ID, nume, descriere, preț, stoc), clienți (ID, nume, prenume, email, telefon, adresă), comenzi (ID comandă, client asociat, dată, total) și detalii despre comenzi (ID produs, cantitate, preț unitar). Sistemul urmărește să faciliteze procesul de vânzare și să ofere o evidență clară a relațiilor dintre produsele vândute, comenzile plasate și clienții magazinului.

| Entități                      | Tip Relație | Relații   |
|-------------------------------|-------------|---|
| UTILIZATORI - COMENZI         | One to Many | Fiecare utilizator poate plasa mai multe comenzi. Relația este implementată prin coloana ID_UTILIZATOR din tabelul <b>COMENZI</b> .   |
| COMENZI - DETALII_COMENZI     | One to Many | O comandă conține mai multe produse, detaliate în tabelul <b>DETALII_COMENZI</b> . Relația este implementată prin coloana ID_COMANDA. |
| PRODUSE - DETALII_COMENZI     | One to Many | Un produs poate fi inclus în mai multe comenzi. Relația este definită prin coloana ID_PRODUS din tabelul <b>DETALII_COMENZI</b> .     |
| CATEGORII - PRODUSE           | One to Many | O categorie poate avea mai multe produse. Relația este implementată prin coloana ID_CATEGORIE din tabelul <b>PRODUSE</b> .            |
| FURNIZORI - PRODUSE_FURNIZORI | One to Many | Fiecare furnizor poate livra mai multe produse. Relația este definită prin coloana ID_FURNIZOR din tabelul <b>PRODUSE_FURNIZORI</b> . |
| PRODUSE - RECENZII            | One to Many | Fiecare produs poate avea mai multe recenzii. Relația este implementată prin coloana ID_PRODUS din tabelul <b>RECENZII</b> .          |
| UTILIZATORI - RECENZII        | One to Many | Fiecare utilizator poate lăsa mai multe recenzii. Relația este definită prin coloana ID_UTILIZATOR din tabelul <b>RECENZII</b> .      |
| PRODUSE - REDUCERI            | One to Many | Fiecare produs poate avea reduceri asociate, definite în tabelul <b>REDUCERI</b> . Relația este implementată prin coloana ID_PRODUS.  |

## ***1. Definirea schemei bazei de date***

**-- UTILIZATORI – Salvează informațiile despre utilizatori.**

```
CREATE TABLE UTILIZATORI (  
    ID_UTILIZATOR NUMBER(10) PRIMARY KEY,  
    NUME_UTILIZATOR VARCHAR2(50) UNIQUE NOT NULL,  
    EMAIL VARCHAR2(100) UNIQUE NOT NULL,  
    PAROLA VARCHAR2(100) NOT NULL,  
    NUME_COMPLET VARCHAR2(100),  
    DATA_NASTERII DATE  
);
```

**--PRODUSE – Stochează informații despre produse.**

```
CREATE TABLE PRODUSE (  
    ID_PRODUS NUMBER(10) PRIMARY KEY,  
    NUME_PRODUS VARCHAR2(100) NOT NULL,  
    PRET NUMBER(10, 2) NOT NULL,  
    STOC NUMBER DEFAULT 0,  
    ID_CATEGORIE NUMBER,  
    FOREIGN KEY (ID_CATEGORIE) REFERENCES  
CATEGORII(ID_CATEGORIE) ON DELETE SET NULL  
);
```

**--CATEGORII – Grupați produsele pe categorii.**

```
CREATE TABLE CATEGORII (  
    ID_CATEGORIE NUMBER(10) PRIMARY KEY,  
    NUME_CATEGORIE VARCHAR2(100) UNIQUE NOT NULL,  
    DESCRIERE VARCHAR2(255)  
);
```

**--COMENZI – Detalii despre comenzile utilizatorilor.**

```
CREATE TABLE COMENZI (  
    ID_COMANDA NUMBER(10) PRIMARY KEY,  
    ID_UTILIZATOR NUMBER NOT NULL,  
    DATA_COMANDA DATE DEFAULT SYSDATE,  
    TOTAL_PLATA NUMBER(10, 2) DEFAULT 0,  
    FOREIGN KEY (ID_UTILIZATOR) REFERENCES  
UTILIZATORI(ID_UTILIZATOR) ON DELETE CASCADE  
);
```

**--DETALII\_COMENZI – Lista produselor dintr-o comandă.**

```
CREATE TABLE DETALII_COMENZI (  
    ID_DETALII_COMANDA NUMBER(10) PRIMARY KEY,  
    ID_COMANDA NUMBER NOT NULL,  
    ID_PRODUS NUMBER NOT NULL,  
    CANTITATE NUMBER DEFAULT 1 CHECK (CANTITATE > 0),  
    PRET_PE_UNITATE NUMBER(10, 2) NOT NULL,  
    FOREIGN KEY (ID_COMANDA) REFERENCES COMENZI(ID_COMANDA) ON  
DELETE CASCADE,  
    FOREIGN KEY (ID_PRODUS) REFERENCES PRODUSE(ID_PRODUS) ON  
DELETE CASCADE  
);
```

**--FURNIZORI – Informații despre furnizori.**

```
CREATE TABLE FURNIZORI (  
    ID_FURNIZOR NUMBER(10) PRIMARY KEY,  
    NUME_FURNIZOR VARCHAR2(100) NOT NULL,  
    EMAIL_CONTACT VARCHAR2(100) UNIQUE,  
    TELEFON VARCHAR2(15),  
    ADRESA VARCHAR2(255)  
);
```

**--PRODUSE\_FURNIZORI – Relația dintre furnizori și produse.**

```
CREATE TABLE PRODUSE_FURNIZORI (  
    ID_PRODUS_FURNIZOR NUMBER(10) PRIMARY KEY,  
    ID_FURNIZOR NUMBER NOT NULL,  
    ID_PRODUS NUMBER NOT NULL,  
    PRET_FURNIZARE NUMBER(10, 2) NOT NULL,  
    FOREIGN KEY (ID_FURNIZOR) REFERENCES FURNIZORI(ID_FURNIZOR)  
ON DELETE CASCADE,  
    FOREIGN KEY (ID_PRODUS) REFERENCES PRODUSE(ID_PRODUS) ON  
DELETE CASCADE  
);
```

**--RECENZII – Recenzii date de utilizatori pentru produse.**

```
CREATE TABLE RECENZII (  
    ID_RECENZIE NUMBER(10) PRIMARY KEY,  
    ID_UTILIZATOR NUMBER NOT NULL,  
    ID_PRODUS NUMBER NOT NULL,  
    RATING NUMBER CHECK (RATING BETWEEN 1 AND 5),  
    TEXT_RECENZIE VARCHAR2(255),  
    DATA_RECENZIE DATE DEFAULT SYSDATE,  
    FOREIGN KEY (ID_UTILIZATOR) REFERENCES  
UTILIZATORI(ID_UTILIZATOR) ON DELETE CASCADE,  
    FOREIGN KEY (ID_PRODUS) REFERENCES PRODUSE(ID_PRODUS) ON  
DELETE CASCADE  
);
```

**--REDUCERI** – Reduceri aplicate produselor.

```
CREATE TABLE REDUCERI (
    ID_REDUCERE NUMBER(10) PRIMARY KEY,
    ID_PRODUS NUMBER NOT NULL,
    PROCENT_REDUCERE NUMBER(5, 2) CHECK (PROCENT_REDUCERE
BETWEEN 0 AND 100),
    DATA_INCEPUT DATE NOT NULL,
    DATA_SFARSIT DATE NOT NULL,
    FOREIGN KEY (ID_PRODUS) REFERENCES PRODUSE(ID_PRODUS) ON
DELETE CASCADE
);
```

***Se utilizează comenzile CREATE, ALTER, DROP***

- **Comanda ALTER** – Modificarea unor tabele existente:

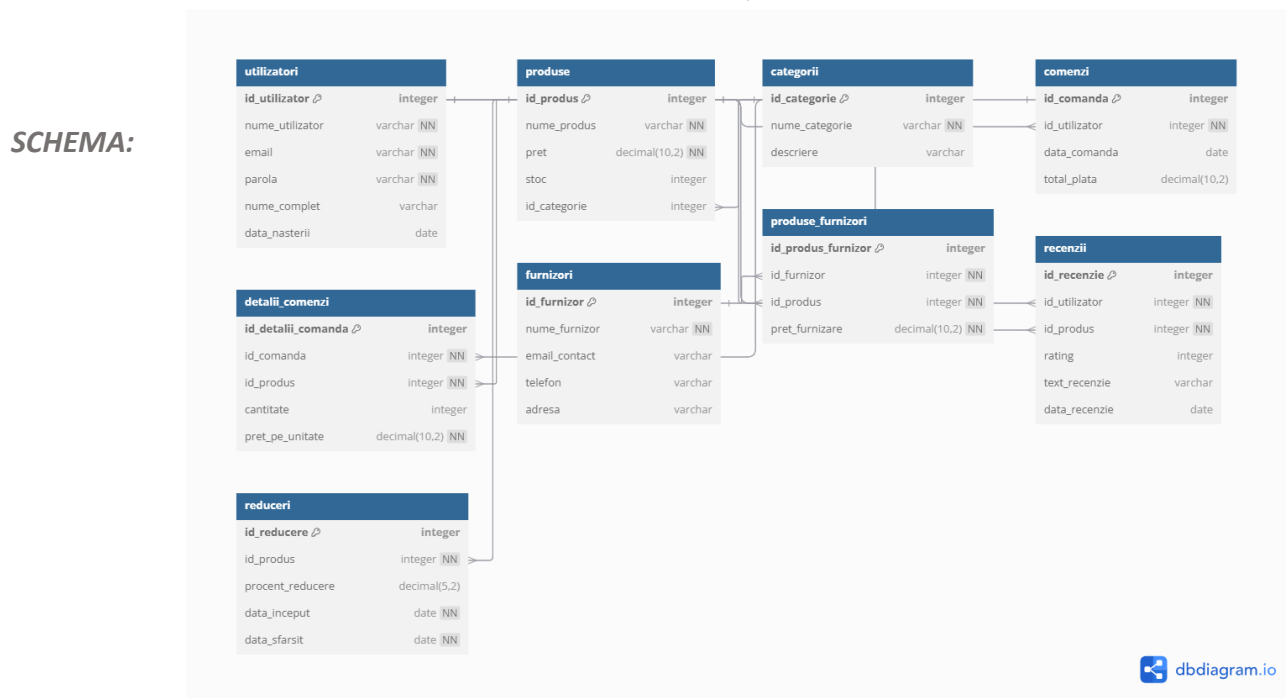
```
ALTER TABLE PRODUCE ADD
(STOC NUMBER(10) DEFAULT 0
);
```

```
ALTER TABLE UTILIZATORI ADD
DATA INREGISTRARII DATE DEFAULT SYSDATE;
```

```
ALTER TABLE PRODUCE MODIFY
(STOC NUMBER(10,2) PRIMARY KEY
);
```

- **Comanda DROP** – Ștergerea si recuperarea unui tabel:

```
DROP TABLE PRODUCE ;
FLASHBACK TABLE PRODUCE TO BEFORE DROP;
```



Oracle SQL Developer: D:\USER\Desktop\sqldeveloper\sqldeveloper\ionut\_1.sql

File Edit View Navigate Run Source Tools Window Help

Connections

Oracle Connections

ionut

Tables (Filtered)

- CATEGORIES
- COMENZI
- DETAILI\_COMENZI
- FURNIZORI
- PRODUSE
- PRODUSE\_FURNIZORI
- RECEZII
- REDOZORI
- UTILIZATORI

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Reports

- All Reports
- About Your Database
- All Objects
- Analytic View Reports
- Application Express
- ADP and ADP
- Database Administration
- Data Dictionary
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- PL/SQL
- Security
- Streams
- Table
- Text/Text Reports
- User Defined Reports
- XML

SQL Worksheet

Query Builder

```

--UTILIZATORI - Salvare informatiile despre utilizatori.
CREATE TABLE UTILIZATORI (
  ID_UTILIZATOR NUMBER(10) PRIMARY KEY,
  NUME_UTILIZATOR VARCHAR2(50) UNIQUE NOT NULL,
  EMAIL VARCHAR2(100) UNIQUE NOT NULL,
  PAROLA VARCHAR2(100) NOT NULL,
  ID_CATEGORIE NUMBER,
  DATA_INSCRIERII DATE
);

--PRODUSE - Stocarea informatiilor despre produse.
CREATE TABLE PRODUSE (
  ID_PRODUS NUMBER(10) PRIMARY KEY,
  NUME_PRODUS VARCHAR2(100) NOT NULL,
  PRET NUMBER(10, 2) NOT NULL,
  STOC NUMBER DEFAULT 0,
  ID_CATEGORIE NUMBER,
  FURNIZOR KEY (ID_CATEGORIE) REFERENCES CATEGORIES(ID_CATEGORIE) ON DELETE SET NULL
);

--CATEGORIES - Gruparea produselor pe categorii.
CREATE TABLE CATEGORIES (
  ID_CATEGORIE NUMBER(10) PRIMARY KEY,
  NUME_CATEGORIE VARCHAR2(100) UNIQUE NOT NULL,
  DESCRIERE VARCHAR2(255)
);

--COMENZI - Detalii despre comenzile utilizatorilor.
CREATE TABLE COMENZI (
  ID_COMANDA NUMBER(10) PRIMARY KEY,
  ID_UTILIZATOR NUMBER NOT NULL,
  DATA_COMANDA DATE DEFAULT SYSDATE,
  TOTAL_PLATA NUMBER(10, 2) DEFAULT 0,
  FURNIZOR KEY (ID_UTILIZATOR) REFERENCES UTILIZATORI(ID_UTILIZATOR) ON DELETE CASCADE
);

```

Script Output

Task completed in 0.913 seconds

Table RECEZII created.

Table REDOZORI created.

Messages - Log

Messages Statements Logging Page

Click on an identifier with the Control key down to perform "Go to Declaration"

Line 57 Column 1 | Insert | Modified | Windows: O

Oracle SQL Developer: Table BSEMAN\_SAPRODUSE@ionut

File Edit View Navigate Run Source Tools Window Help

Connections

Oracle Connections

ionut

Tables (Filtered)

- CATEGORIES
- COMENZI
- DETAILI\_COMENZI
- FURNIZORI
- PRODUSE
- PRODUSE\_FURNIZORI
- RECEZII
- REDOZORI
- UTILIZATORI

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Reports

- All Reports
- About Your Database
- All Objects
- Analytic View Reports
- Application Express
- ADP and ADP
- Database Administration
- Data Dictionary
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- PL/SQL
- Security
- Streams
- Table
- Text/Text Reports
- User Defined Reports
- XML

Table Editor

Columns: Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

| ID_PRODUS | NUME_PRODUS             | PRET    | STOC | ID_CATEGORIE |
|-----------|-------------------------|---------|------|--------------|
| 1         | 103 Tastatură Razer     | 3999    | 15   | 2            |
| 2         | 101 Televizor Samsung   | 1500,99 | 10   | 2            |
| 3         | 102 Mouse Logitech G502 | 249,99  | 50   | 2            |
| 4         | 104 Scaun ergonomic     | 349,99  | 20   | 3            |
| 5         | 105 Săculeț Buzie       | 99,99   | 30   | 4            |
| 6         | 106 Geacă de Iarnă      | 499,99  | 15   | 5            |
| 7         | 201 Tastatură Gaming    | 349,99  | 15   | 1            |
| 8         | 202 Mouse Profesional   | 199,99  | 20   | 1            |
| 9         | 203 Monitor 4K          | 1299,99 | 10   | 2            |
| 10        | 204 Căști Audio         | 499,99  | 30   | 2            |

Compiler - Log

Click on an identifier with the Control key down to perform "Go to Declaration"

Oracle SQL Developer: Table BEAHM\_S4.FURNIZORI@lonut

File Edit View Navigate Run Tools Window Help

Connections

Oracle Connections

BEAHM\_S4.FURNIZORI

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

Columns: ID\_FURNIZOR, Nume\_Furnizor, EMAIL\_CONTACT, TELEFON, ADRESA

| ID_FURNIZOR | Nume_Furnizor            | EMAIL_CONTACT        | TELEFON    | ADRESA                        |
|-------------|--------------------------|----------------------|------------|-------------------------------|
| 1           | Furnizor Tech            | tech@example.com     | 0712345678 | Str. Tehnologiei, Nr. 10      |
| 2           | Distributor Gaming       | gaming@example.com   | 0723456789 | Jocurilor, Nr. 15             |
| 3           | Produsator Office        | office@example.com   | 0734567890 | Str. Birourilor, Nr. 20       |
| 4           | Echipamente Profesionale | prof@example.com     | 0745678901 | Str. Profesioniștilor, Nr. 25 |
| 5           | Accesorii Universale     | access@example.com   | 0756789012 | Str. Universalului, Nr. 30    |
| 6           | Distributor IT           | it@example.com       | 0767890123 | Str. Calculatoarelor, Nr. 40  |
| 7           | Furnizor Gadget-uri      | gadget@example.com   | 0778901234 | Str. Inovatiei, Nr. 50        |
| 8           | Produsator Software      | software@example.com | 0789012345 | Str. Programatorilor, Nr. 60  |
| 9           | Megastin Online          | online@example.com   | 0790123456 | Str. E-commerce, Nr. 70       |
| 10          | Furnizor Electroanice    | electro@example.com  | 0711234567 | Str. Electroaniceilor, Nr. 80 |

Reports

Compiler - Log

Oracle SQL Developer: Table BEAHM\_S4.UTILIZATORI@lonut

File Edit View Navigate Run Tools Window Help

Connections

Oracle Connections

BEAHM\_S4.UTILIZATORI

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

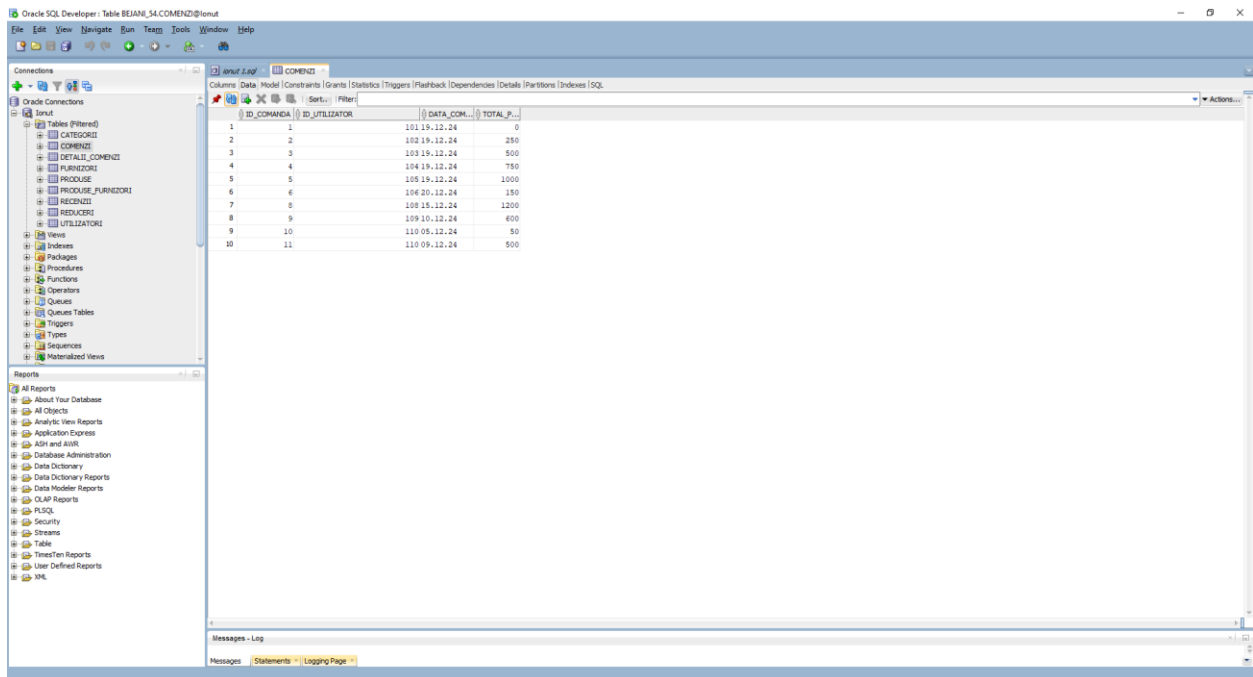
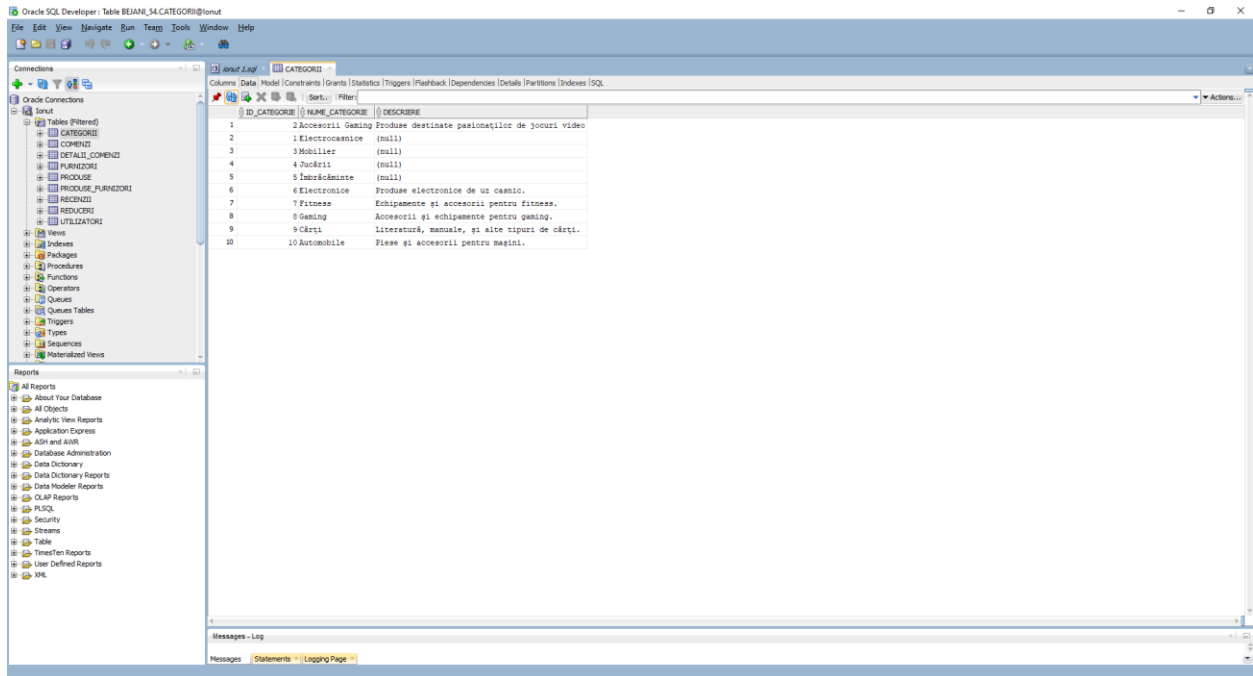
Columns: ID\_UTILIZATOR, Nume\_Utilizator, EMAIL, PAROLA, Nume\_Complet, DATA\_NASTERII

| ID_UTILIZATOR | Nume_Utilizator | EMAIL                   | PAROLA     | Nume_Complet  | DATA_NASTERII |
|---------------|-----------------|-------------------------|------------|---------------|---------------|
| 1             | Ionel Popescu   | ion.popescu@example.com | parola1456 | Ionel Popescu | 21.03.99      |
| 2             | 101 user1       | user1@example.com       | parola1    | User One      | 01.01.90      |
| 3             | 102 user2       | user2@example.com       | parola2    | User Two      | 02.02.92      |
| 4             | 103 user3       | user3@example.com       | parola3    | User Three    | 03.03.94      |
| 5             | 104 user4       | user4@example.com       | parola4    | User Four     | 04.04.96      |
| 6             | 105 user5       | user5@example.com       | parola5    | User Five     | 05.05.98      |
| 7             | 1 user1_new     | new_user@example.com    | parola123  | User One      | 01.01.00      |
| 8             | 106 user6       | user6@example.com       | parola6    | User Six      | 06.06.00      |
| 9             | 107 user7       | user7@example.com       | parola7    | User Seven    | 07.07.01      |
| 10            | 108 user8       | user8@example.com       | parola8    | User Eight    | 08.08.02      |
| 11            | 109 user9       | user9@example.com       | parola9    | User Nine     | 09.09.03      |
| 12            | 110 user10      | user10@example.com      | parola10   | User Ten      | 10.10.04      |

Reports

Compiler - Log





Oracle SQL Developer: Table BEIANI\_S4DETAILI\_COMENZI@lonut

File Edit View Navigate Run Tools Window Help

Connections Oracle Connections

- Tables (Filtered)
- CATEGORIES
- COMENZI
- DETAILI\_COMENZI
- ID\_DETAILI\_COMANDA
- ID\_COMANDA
- ID\_PRODUS
- CANTITATE
- PRET\_PE\_LINIE
- FURNIZORI
- PRODUSE
- PRODUSE\_FURNIZORI
- RECONZII
- REDACTARE
- UTILIZATORI
- Views
- Indexes
- Packages
- Procedures
- Functions
- Operators
- Queues

Reports

- All Reports
- About Your Database
- All Objects
- Analytic View Reports
- Application Express
- ASH and AWR
- Database Administration
- Data Dictionary
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- PLSQL
- Security
- Streams
- Table
- TimesTen Reports
- User Defined Reports
- XML

Columns: Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

|    | ID_DETAILI_COMANDA | ID_COMANDA | ID_PRODUS | CANTITATE | PRET_PE_LINIE |
|----|--------------------|------------|-----------|-----------|---------------|
| 1  | 1                  | 1          | 201       | 2         | 150,5         |
| 2  | 2                  | 1          | 202       | 1         | 200           |
| 3  | 3                  | 2          | 203       | 3         | 350,75        |
| 4  | 4                  | 2          | 204       | 5         | 100           |
| 5  | 11                 | 6          | 201       | 8         | 150           |
| 6  | 12                 | 6          | 202       | 4         | 225,5         |
| 7  | 13                 | 3          | 201       | 2         | 150           |
| 8  | 14                 | 4          | 202       | 3         | 200           |
| 9  | 15                 | 5          | 203       | 1         | 350           |
| 10 | 16                 | 6          | 204       | 4         | 100           |

Messages - Log

Messages Statements Logging Page

Oracle SQL Developer: Table BEIANI\_S4PRODUSE\_FURNIZORI@lonut

File Edit View Navigate Run Tools Window Help

Connections Oracle Connections

- Tables (Filtered)
- CATEGORIES
- COMENZI
- DETAILI\_COMENZI
- FURNIZORI
- PRODUSE
- PRODUSE\_FURNIZORI
- RECONZII
- REDACTARE
- UTILIZATORI
- Views
- Indexes
- Packages
- Procedures
- Functions
- Operators
- Queues
- Queues Tables
- Triggers
- Types
- Sequences
- Materialized Views

Reports

- All Reports
- About Your Database
- All Objects
- Analytic View Reports
- Application Express
- ASH and AWR
- Database Administration
- Data Dictionary
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- PLSQL
- Security
- Streams
- Table
- TimesTen Reports
- User Defined Reports
- XML

Columns: Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

|    | ID_PRODUSE_FURNIZOR | ID_FURNIZOR | ID_PRODUS | PRET_FURNIZARE |
|----|---------------------|-------------|-----------|----------------|
| 1  | 1                   | 1           | 201       | 120            |
| 2  | 2                   | 1           | 202       | 150            |
| 3  | 3                   | 2           | 203       | 200            |
| 4  | 4                   | 2           | 204       | 200            |
| 5  | 11                  | 1           | 201       | 120            |
| 6  | 12                  | 2           | 202       | 150            |
| 7  | 13                  | 3           | 203       | 300            |
| 8  | 14                  | 4           | 204       | 200            |
| 9  | 15                  | 2           | 205       | 250            |
| 10 | 16                  | 3           | 206       | 300            |

Messages - Log

Messages Statements Logging Page

Oracle SQL Developer: Table BEIANI\_SARECENZII@ionut

File Edit View Navigate Run Tools Window Help

Connections

Oracle Connections

Ionut

Tables (Filtered)

CATEGORIES

COMENZI

DETAILL\_COMENZI

FURNIZORI

PRODUSE

PRODUSE\_FURNIZORI

RECENZII

REDUCERI

UTILIZATORI

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queue Tables

Triggers

Types

Sequences

Materialized Views

Reports

All Reports

About Your Database

All Objects

Analytic View Reports

Application Express

ASH and ASHR

Database Administration

Data Dictionary

Data Dictionary Reports

Data Modeler Reports

OLAP Reports

PLSQL

Security

Streams

Table

TimeTen Reports

User Defined Reports

XPL

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

Start... Filter

| ID_RECENZIE | ID_UTILIZATOR | ID_PRODUS | RATING | TEXT_RECENZIE                                   | DATA_RECENZIE |
|-------------|---------------|-----------|--------|---|---------------|
| 1           | 1             | 101       | 201    | 5 Excelent produs!                              | 19.12.24      |
| 2           | 2             | 102       | 202    | 4 Foarte bun, dar cu mici probleme.             | 19.12.24      |
| 3           | 3             | 103       | 203    | 5 Perfect pentru nevoile mele!                  | 19.12.24      |
| 4           | 4             | 104       | 204    | 3 Mediocr, putea fi mai bun.                    | 19.12.24      |
| 5           | 5             | 105       | 201    | 5 Recomand cu incredere!                        | 19.12.24      |
| 6           | 6             | 101       | 202    | 4 Produs bun, dar livrarea a intarziat.         | 20.12.24      |
| 7           | 7             | 102       | 203    | 5 Calitate foarte buna, recomand!               | 20.12.24      |
| 8           | 8             | 103       | 204    | 3 Ambalaj defectuos, dar produsul functioneaza. | 20.12.24      |
| 9           | 9             | 104       | 205    | 2 Nu sunt multumit de performanta.              | 20.12.24      |
| 10          | 10            | 105       | 206    | 5 Un produs excelent, livrare rapida!           | 20.12.24      |

Messages - Log

Messages Statements Logging Page

Oracle SQL Developer: Table BEIANI\_SARECENZII@ionut

File Edit View Navigate Run Tools Window Help

Connections

Oracle Connections

Ionut

Tables (Filtered)

CATEGORIES

COMENZI

DETAILL\_COMENZI

FURNIZORI

PRODUSE

PRODUSE\_FURNIZORI

RECENZII

REDUCERI

UTILIZATORI

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queue Tables

Triggers

Types

Sequences

Materialized Views

Reports

All Reports

About Your Database

All Objects

Analytic View Reports

Application Express

ASH and ASHR

Database Administration

Data Dictionary

Data Dictionary Reports

Data Modeler Reports

OLAP Reports

PLSQL

Security

Streams

Table

TimeTen Reports

User Defined Reports

XPL

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

Start... Filter

| ID_RECENZIE | ID_PRODUS | PROCENT_REDUCTION | DATA_INCEPUT | DATA_SFARSIT |
|-------------|-----------|-------------------|--------------|--------------|
| 1           | 1         | 201               | 10.01.2024   | 31.12.24     |
| 2           | 2         | 202               | 15.05.2024   | 20.12.24     |
| 3           | 3         | 203               | 20.10.2024   | 25.12.24     |
| 4           | 4         | 204               | 5.15.2024    | 31.12.24     |
| 5           | 5         | 205               | 25.01.2024   | 10.12.24     |
| 6           | 6         | 206               | 30.05.2024   | 20.12.24     |
| 7           | 7         | 207               | 12.5.10.2024 | 25.12.24     |
| 8           | 8         | 208               | 8.15.12.24   | 31.12.24     |
| 9           | 9         | 209               | 10.01.2024   | 10.12.24     |
| 10          | 10        | 210               | 50.01.2024   | 31.12.24     |

Messages - Log

Messages Statements Logging Page

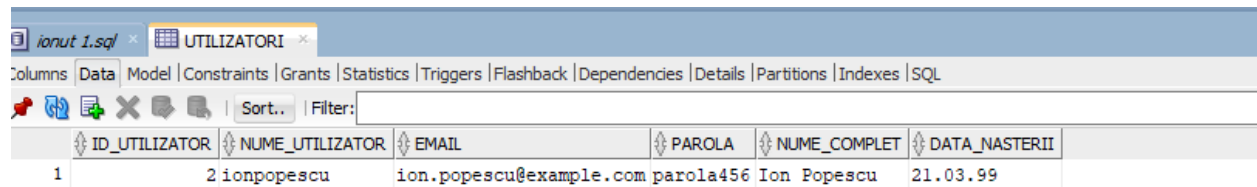
## 2.Operații DML(Insert, Update, Delete, Merge)

### 1. INSERT - Adăugare de date în tabele

- **EX 1: Adăugare utilizatori în tabelul UTILIZATORI**

```
INSERT INTO UTILIZATORI (ID_UTILIZATOR, NUME_UTILIZATOR, EMAIL,  
PAROLA, NUME_COMPLET, DATA_NASTERII)
```

```
VALUES (2, 'ionpopescu', 'ion.popescu@example.com', 'parola456',  
'Ion Popescu', TO_DATE('1999-03-21', 'YYYY-MM-DD'));
```



The screenshot shows a database management tool interface. At the top, there are tabs for 'ionut 1.sql' and 'UTILIZATORI'. Below the tabs, there is a menu bar with options: Columns, Data, Model, Constraints, Grants, Statistics, Triggers, Flashback, Dependencies, Details, Partitions, Indexes, and SQL. Below the menu bar, there is a toolbar with icons for various database operations. Below the toolbar, there is a table with the following columns: ID\_UTILIZATOR, NUME\_UTILIZATOR, EMAIL, PAROLA, NUME\_COMPLET, and DATA\_NASTERII. The table contains one row with the following values: 2, ionpopescu, ion.popescu@example.com, parola456, Ion Popescu, and 21.03.99.

| ID_UTILIZATOR | NUME_UTILIZATOR | EMAIL                   | PAROLA    | NUME_COMPLET | DATA_NASTERII |
|---------------|-----------------|-------------------------|-----------|--------------|---------------|
| 2             | ionpopescu      | ion.popescu@example.com | parola456 | Ion Popescu  | 21.03.99      |

- **EX 2: Adăugare produse în tabelul PRODUSE**

```
INSERT INTO PRODUSE (ID_PRODUS, NUME_PRODUS, PRET, STOC,  
ID_CATEGORIE)VALUES (102, 'Mouse Logitech G502', 249.99, 50, 2);
```

- **EX 3: Adăugare recenzii în tabelul RECENZII**

```
INSERT INTO RECENZII (ID_RECENZIE, ID_UTILIZATOR, ID_PRODUS,  
RATING, TEXT_RECENZIE, DATA_RECENZIE)
```

```
VALUES (1, 2, 102, 5, 'Mouse excelent pentru gaming!', SYSDATE);
```

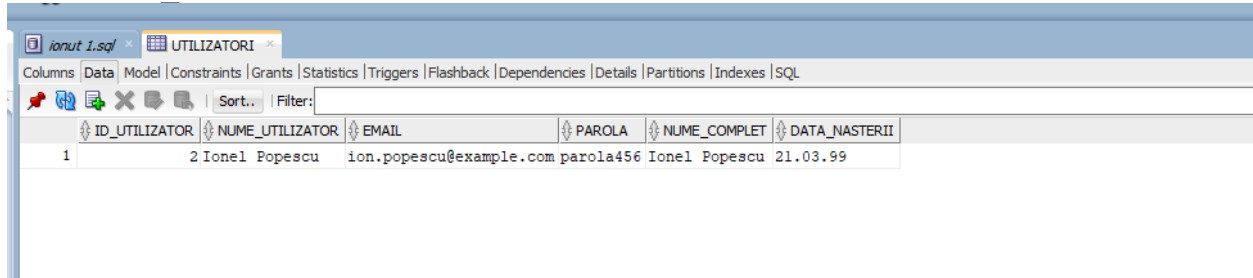
### 2. UPDATE - Modificare date existente

- **EX 1: Actualizează prețul unui produs**

```
UPDATE PRODUSE  
SET PRET = 219.99  
WHERE ID_PRODUS = 102;
```

- **EX 2: Actualizează numele complet al unui utilizator**

```
UPDATE UTILIZATORI
SET NUME_UTILIZATOR = 'Ionel Popescu'
WHERE ID_UTILIZATOR = 2;
```



| ID_UTILIZATOR | NUME_UTILIZATOR | EMAIL                   | PAROLA    | NUME_COMPLET  | DATA_NASTERII |
|---------------|-----------------|-------------------------|-----------|---------------|---------------|
| 1             |                 |                         |           |               |               |
| 2             | Ionel Popescu   | ion.popescu@example.com | parola456 | Ionel Popescu | 21.03.99      |

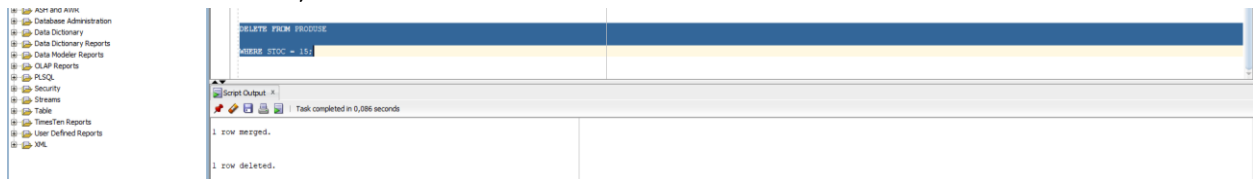
- **EX 3: Actualizează cantitatea stocului pentru un produs**

```
UPDATE PRODUSE
SET STOC = STOC + 20
WHERE ID_PRODUS = 102;
```

### 3. DELETE - Ștergerea datelor

- **EX 1: Șterge un produs care nu mai este în stoc**

```
DELETE FROM PRODUSE
WHERE STOC = 15;
```



| ID_PRODUS | NUME_PRODUS | STOC |
|-----------|-------------|------|
| 101       | Produs 1    | 15   |
| 102       | Produs 2    | 20   |
| 103       | Produs 3    | 10   |

Script Output:

```
Task completed in 0.086 seconds
1 row merged.
1 row deleted.
```

- **EX 2: Șterge toate recenziile unui utilizator specific**

```
DELETE FROM RECENZII
WHERE ID_UTILIZATOR = 2;
```

- **EX 3: Șterge un furnizor din baza de date**

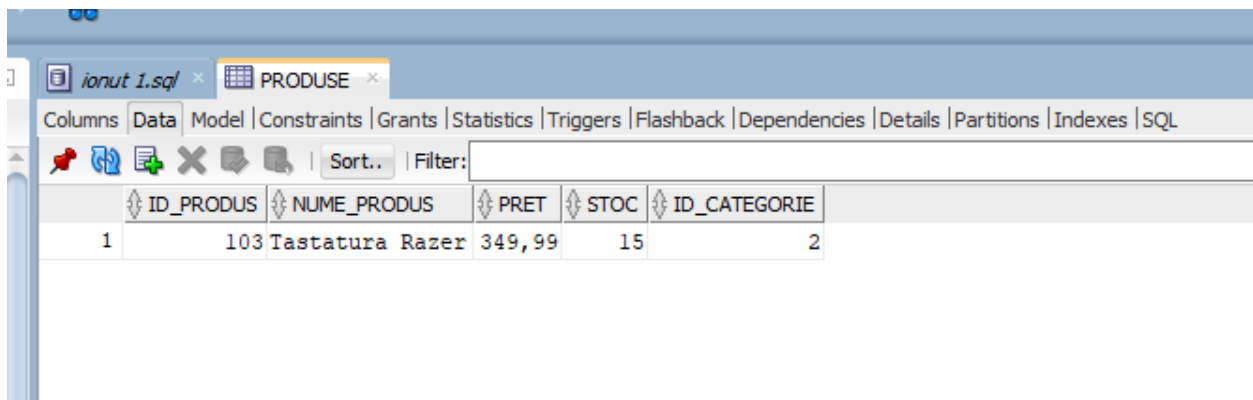
```
DELETE FROM FURNIZORI
```

```
WHERE ID_FURNIZOR = 3;
```

#### 4. **MERGE - Inserare sau actualizare condiționată**

- **EX 1: Actualizează prețul unui produs sau inserează-l dacă nu există**

```
MERGE INTO PRODUSE P
USING (SELECT 103 AS ID_PRODUS,
  'Tastatura Razer' AS NUME_PRODUS,
  349.99 AS PRET,
  15 AS STOC,
  2 AS ID_CATEGORIE
FROM DUAL) SRC
ON (P.ID_PRODUS = SRC.ID_PRODUS)
WHEN MATCHED THEN
  UPDATE SET PRET = SRC.PRET, STOC = SRC.STOC
WHEN NOT MATCHED THEN
  INSERT (ID_PRODUS, NUME_PRODUS, PRET, STOC, ID_CATEGORIE)
  VALUES (SRC.ID_PRODUS, SRC.NUME_PRODUS, SRC.PRET, SRC.STOC,
SRC.ID_CATEGORIE);
```



The screenshot shows a database management interface with a tab for 'ionut 1.sql' and a table named 'PRODUSE'. The table has five columns: ID\_PRODUS, NUME\_PRODUS, PRET, STOC, and ID\_CATEGORIE. A single row of data is displayed with the following values: 103, Tastatura Razer, 349,99, 15, and 2.

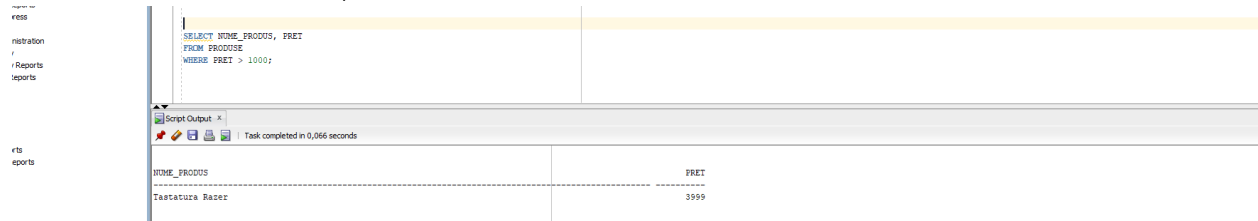
|   | ID_PRODUS | NUME_PRODUS     | PRET   | STOC | ID_CATEGORIE |
|---|-----------|-----------------|--------|------|--------------|
| 1 | 103       | Tastatura Razer | 349,99 | 15   | 2            |

### 3.Exemple de interogări variate (SELECT)

#### 1. Utilizarea operatorilor de comparație

- **EX 1: Afișează produsele cu preț mai mare de 1000.**

```
SELECT NUME_PRODUS, PRET
FROM PRODUSE
WHERE PRET > 1000;
```



The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the SQL statement for EX 1. The 'Script Output' pane shows the task completed in 0.066 seconds. The results pane displays a table with two columns: NUME\_PRODUS and PRET. The data row shows 'Tastatura Razer' with a price of 3599.

| NUME_PRODUS     | PRET |
|-----------------|------|
| Tastatura Razer | 3599 |

- **EX 2: Afișează comenzile cu totalul mai mic de 500.**

```
SELECT ID_COMANDA, TOTAL_PLATA
FROM COMENZI
WHERE TOTAL_PLATA < 500;
```

- **EX 3: Afișează produsele cu stoc între 10 și 50.**

```
SELECT NUME_PRODUS, STOC
FROM PRODUSE
WHERE STOC BETWEEN 10 AND 50;
```

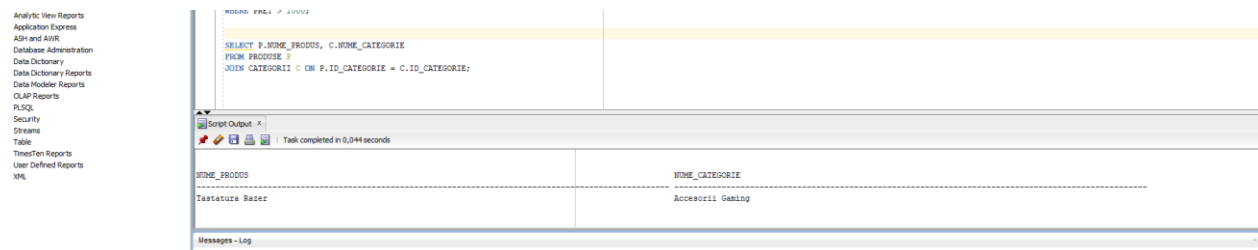
#### 2. Join-uri

- **EX 1: Afișează comenzile împreună cu numele utilizatorilor care le-au plasat.**

```
SELECT C.ID_COMANDA, U.NUME_UTILIZATOR, C.DATA_COMANDA,
C.TOTAL_PLATA
FROM COMENZI C
JOIN UTILIZATORI U ON C.ID_UTILIZATOR = U.ID_UTILIZATOR;
```

- **EX 2: Afișează produsele și categoriile lor.**

```
SELECT P.NUME_PRODUS, C.NUME_CATEGORIE
FROM PRODUSE P
JOIN CATEGORII C ON P.ID_CATEGORIE = C.ID_CATEGORIE;
```



The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the SQL statement for EX 2. The 'Script Output' pane shows the task completed in 0.044 seconds. The results pane displays a table with two columns: NUME\_PRODUS and NUME\_CATEGORIE. The data row shows 'Tastatura Razer' with the category 'Accesorii Gaming'.

| NUME_PRODUS     | NUME_CATEGORIE   |
|-----------------|------------------|
| Tastatura Razer | Accesorii Gaming |

- **EX 3: Afișează recenziile produselor împreună cu utilizatorii care le-au scris.**

```
SELECT R.ID_RECENZIE, U.NUME_UTILIZATOR, R.RATING,
R.TEXT_RECENZIE
FROM RECENZII R
JOIN UTILIZATORI U ON R.ID_UTILIZATOR = U.ID_UTILIZATOR;
```

### 3. Utilizarea funcțiilor de grup și condiții asupra acestora

- **EX 1: Afișează numărul de produse în fiecare categorie.**

```
SELECT C.NUME_CATEGORIE, COUNT(P.ID_PRODUS) AS NUMAR_PRODUSE
FROM CATEGORII C
LEFT JOIN PRODUSE P ON C.ID_CATEGORIE = P.ID_CATEGORIE
GROUP BY C.NUME_CATEGORIE;
```

The screenshot shows the SQL Developer interface with the query execution results. The query is: `SELECT C.NUME_CATEGORIE, COUNT(P.ID_PRODUS) AS NUMAR_PRODUSE FROM CATEGORII C LEFT JOIN PRODUSE P ON C.ID_CATEGORIE = P.ID_CATEGORIE GROUP BY C.NUME_CATEGORIE;` The results are displayed in a table with two columns: `NUME_CATEGORIE` and `NUMAR_PRODUSE`.

| NUME_CATEGORIE   | NUMAR_PRODUSE |
|------------------|---------------|
| Accesorii Gaming | 3             |
| Mobilier         | 1             |
| Jucării          | 1             |
| Îmbrăcăminte     | 1             |
| Electrocasnice   | 0             |

- **EX 2: Afișează suma totală a comenzilor pentru fiecare utilizator.**

```
SELECT U.NUME_UTILIZATOR, SUM(C.TOTAL_PLATA) AS TOTAL_COMENZI
FROM COMENZI C
JOIN UTILIZATORI U ON C.ID_UTILIZATOR = U.ID_UTILIZATOR
GROUP BY U.NUME_UTILIZATOR;
```

- **EX3: Afișează stocul mediu al produselor dacă acesta este mai mare de 20.**

```
SELECT AVG(STOC) AS STOC_MEDIU
FROM PRODUSE
HAVING AVG(STOC) > 20;
```

### 4. Utilizarea funcțiilor numerice, de tip caracter, pentru data și timp

- **EX 1: Afișează numele produselor în litere mari.**

```
SELECT UPPER(NUME_PRODUS) AS NUME_PRODUS_MARE
FROM PRODUSE;
```

The screenshot shows the SQL Developer interface with the query execution results. The query is: `SELECT UPPER(NUME_PRODUS) AS NUME_PRODUS_MARE FROM PRODUSE;` The results are displayed in a table with one column: `NUME_PRODUS_MARE`.

| NUME_PRODUS_MARE    |
|---------------------|
| TASTATURA RAZER     |
| TELEVIZOR SAMSUNG   |
| MOUSE LOGITECH G502 |
| SCAUN ERGONOMIC     |
| PĂPUȘĂ BARBIE       |
| GEACĂ DE IARNĂ      |

6 rows selected.



- **EX 2: Afișează produsele reduse în anul curent.**

```
SELECT NUME_PRODUS, DATA_INCEPUT
FROM REDUCERI
WHERE EXTRACT(YEAR FROM DATA_INCEPUT) = EXTRACT(YEAR FROM
SYSDATE);
```

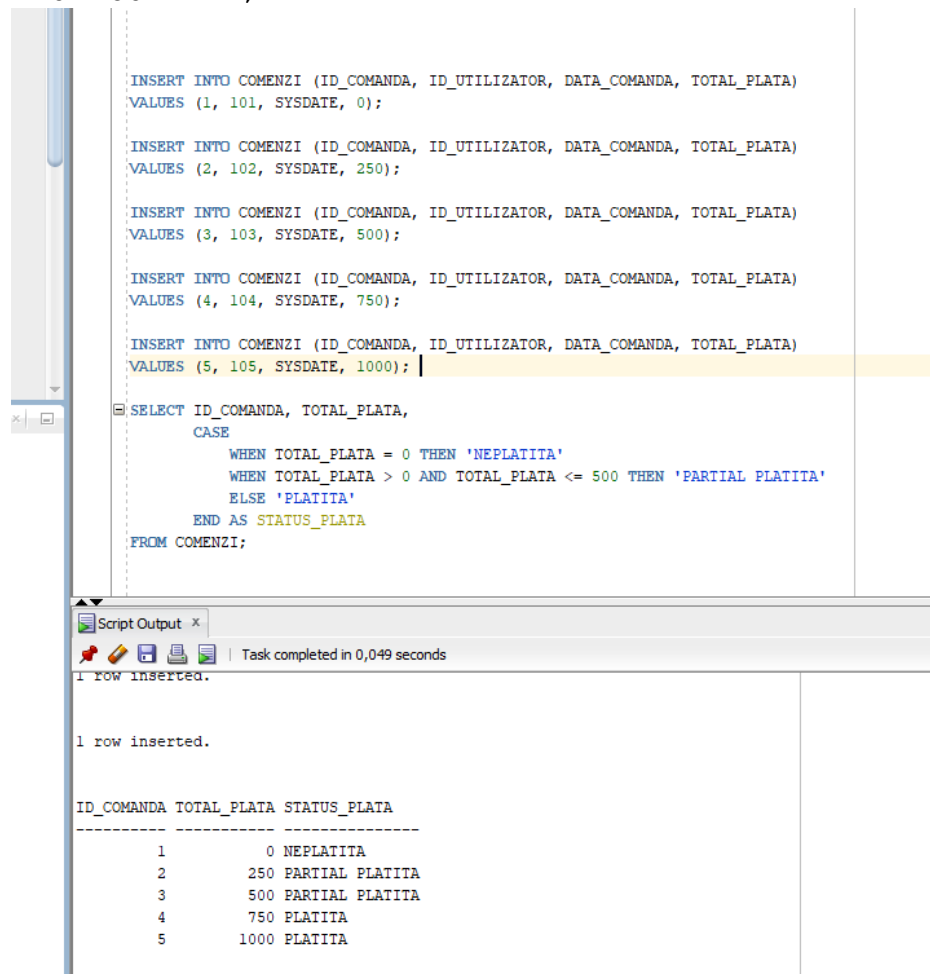
- **EX 3: Afișează numărul de zile dintre data de început și sfârșit a reducerilor.**

```
SELECT ID_REDCERE, DATA_INCEPUT, DATA_SFARSIT, (DATA_SFARSIT -
DATA_INCEPUT) AS ZILE_REDCERE
FROM REDUCERI;
```

## 5. Construirea de expresii cu DECODE și CASE

- **EX 1: Afișează comenzile cu statusul plății (NEPLĂTITĂ, PARȚIAL PLĂTITĂ, PLĂTITĂ).**

```
SELECT ID_COMANDA, TOTAL_PLATA,
CASE
    WHEN TOTAL_PLATA = 0 THEN 'NEPLATITA'
    WHEN TOTAL_PLATA > 0 AND TOTAL_PLATA <= 500 THEN
'PARTIAL PLATITA'
    ELSE 'PLATITA'
END AS STATUS_PLATA
FROM COMENZI;
```



The screenshot shows a SQL IDE with a script editor and a script output window. The script editor contains the following SQL commands:

```
INSERT INTO COMENZI (ID_COMANDA, ID_UTILIZATOR, DATA_COMANDA, TOTAL_PLATA)
VALUES (1, 101, SYSDATE, 0);

INSERT INTO COMENZI (ID_COMANDA, ID_UTILIZATOR, DATA_COMANDA, TOTAL_PLATA)
VALUES (2, 102, SYSDATE, 250);

INSERT INTO COMENZI (ID_COMANDA, ID_UTILIZATOR, DATA_COMANDA, TOTAL_PLATA)
VALUES (3, 103, SYSDATE, 500);

INSERT INTO COMENZI (ID_COMANDA, ID_UTILIZATOR, DATA_COMANDA, TOTAL_PLATA)
VALUES (4, 104, SYSDATE, 750);

INSERT INTO COMENZI (ID_COMANDA, ID_UTILIZATOR, DATA_COMANDA, TOTAL_PLATA)
VALUES (5, 105, SYSDATE, 1000);

SELECT ID_COMANDA, TOTAL_PLATA,
CASE
    WHEN TOTAL_PLATA = 0 THEN 'NEPLATITA'
    WHEN TOTAL_PLATA > 0 AND TOTAL_PLATA <= 500 THEN 'PARTIAL PLATITA'
    ELSE 'PLATITA'
END AS STATUS_PLATA
FROM COMENZI;
```

The script output window shows the following results:

```
Task completed in 0,049 seconds

1 row inserted.

1 row inserted.

ID_COMANDA TOTAL_PLATA STATUS_PLATA
-----
1          0 NEPLATITA
2         250 PARTIAL PLATITA
3         500 PARTIAL PLATITA
4         750 PLATITA
5        1000 PLATITA
```

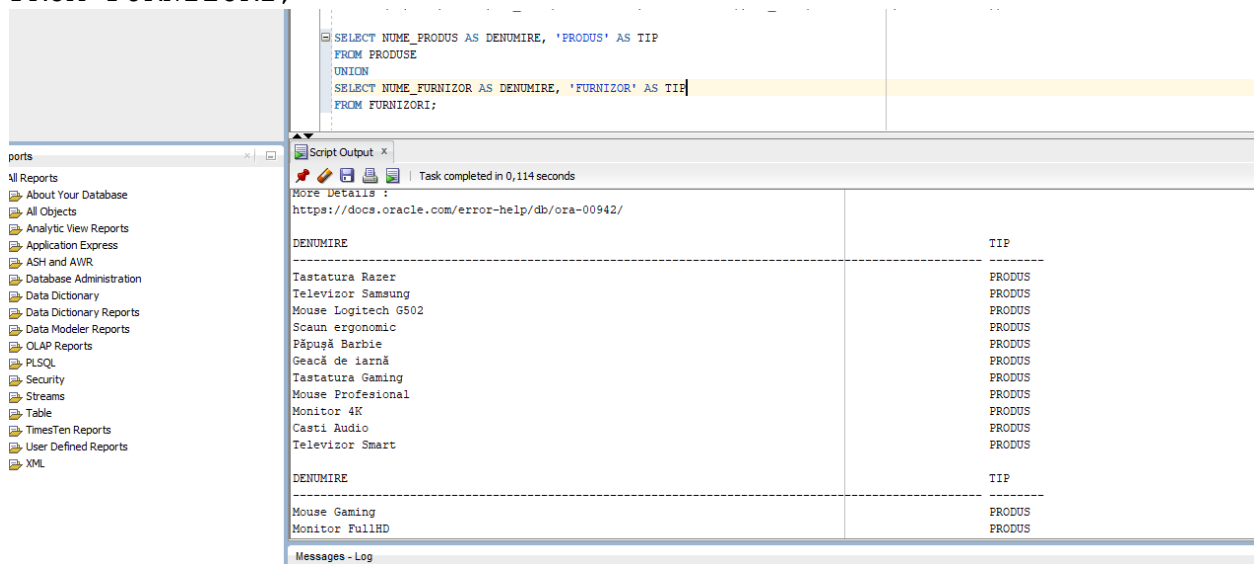
- **EX 2: Afișează produsele cu un mesaj bazat pe stoc (STOC SCAZUT, MINIM, SUFICIENT).**

```
SELECT NUME_PRODUS, STOC,
       DECODE(
         SIGN(STOC - 10),
         -1, 'STOC SCAZUT',
         0, 'STOC MINIM',
         1, 'STOC SUFICIENT'
       ) AS STATUS_STOC
FROM PRODUSE;
```

## 6. Utilizarea operatorilor UNION, MINUS, INTERSECT

- **EX 1: Am combinați numele produselor și furnizorilor într-o listă unică, care poate fi utilizată pentru a genera un catalog general al resurselor unui magazin.**

```
SELECT NUME_PRODUS AS DENUMIRE, 'PRODUS' AS TIP
FROM PRODUSE
UNION
SELECT NUME_FURNIZOR AS DENUMIRE, 'FURNIZOR' AS TIP
FROM FURNIZORI;
```



Script Output x

Task completed in 0,114 seconds

More Details :  
<https://docs.oracle.com/error-help/db/ora-00942/>

| DENUMIRE            | TIP    |
|---------------------|--------|
| Tastatura Razer     | PRODUS |
| Televizor Samsung   | PRODUS |
| Mouse Logitech G502 | PRODUS |
| Scaun ergonomic     | PRODUS |
| Păpușă Barbie       | PRODUS |
| Geacă de iarnă      | PRODUS |
| Tastatura Gaming    | PRODUS |
| Mouse Profesional   | PRODUS |
| Monitor 4K          | PRODUS |
| Casti Audio         | PRODUS |
| Televizor Smart     | PRODUS |
| DENUMIRE            | TIP    |
| Mouse Gaming        | PRODUS |
| Monitor FullHD      | PRODUS |

Messages - Log

- **EX 2: Afișează produsele care nu au recenzii.**

```
SELECT ID_PRODUS
FROM PRODUSE
MINUS
SELECT ID_PRODUS
FROM RECENZII;
```

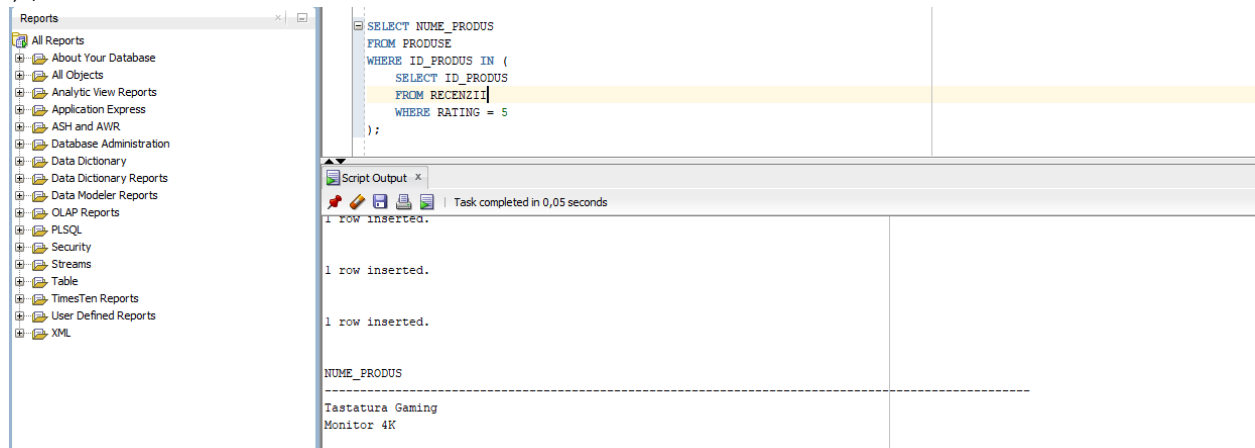
- **EX 3: Afișează produsele care au atât reduceri cât și recenzii.**

```
SELECT ID_PRODUS
FROM RECENZII
INTERSECT
SELECT ID_PRODUS
FROM REDUCERI;
```

## 7. Subcereri (cereri imbricate)

- **EX 1: Afișează produsele care au recenzii cu rating maxim.**

```
SELECT NUME_PRODUS
FROM PRODUSE
WHERE ID_PRODUS IN (
    SELECT ID_PRODUS
    FROM RECENZII
    WHERE RATING = 5
);
```



- **EX 2: Afișează utilizatorii care au plasat comenzi.**

```
SELECT NUME_UTILIZATOR
FROM UTILIZATORI
WHERE ID_UTILIZATOR IN (
    SELECT ID_UTILIZATOR
    FROM COMENZI
);
```

- **EX 3: Afișează comenzile cu suma totală peste media tuturor comenzilor.**

```
SELECT ID_COMANDA, TOTAL_PLATA
FROM COMENZI
WHERE TOTAL_PLATA > (
    SELECT AVG(TOTAL_PLATA)
    FROM COMENZI
);
```

### 3. Gestiunea altor obiecte ale bazei de date.

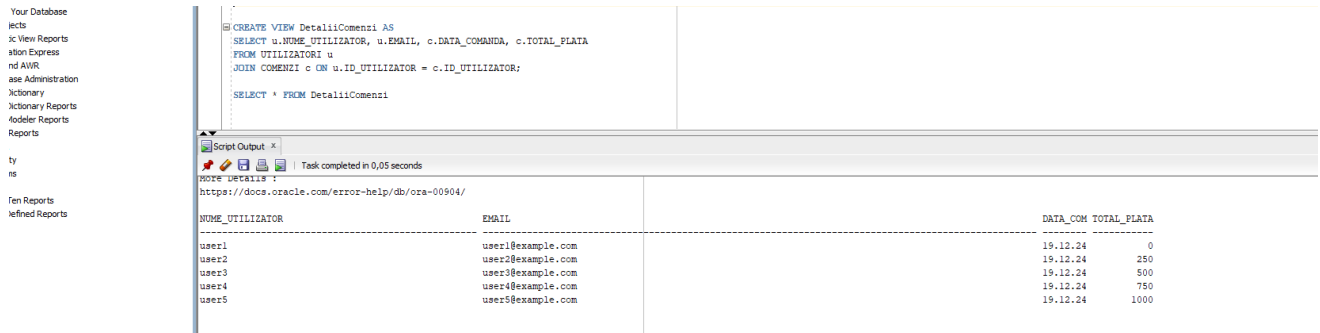
#### 1. Vederi (Views)

*Vederile sunt structuri logice ce permit accesarea datelor într-un mod simplificat. Ele pot fi folosite pentru a ascunde complexitatea interogărilor sau pentru securitate (afișând doar anumite coloane).*

```
CREATE VIEW DetaliiComenzi AS
SELECT u.NUME_UTILIZATOR, u.EMAIL, c.DATA_COMANDA, c.TOTAL_PLATA
FROM UTILIZATORI u
JOIN COMENZI c ON u.ID_UTILIZATOR = c.ID_UTILIZATOR;
```

Utilizarea vederii:

```
SELECT * FROM DetaliiComenzi
```



The screenshot shows the Oracle SQL Developer interface. On the left is a sidebar with a tree view containing 'Your Database', 'Schemas', 'Views', 'Tables', 'Sequences', 'Synonyms', 'Materialized Views', 'Database Properties', 'Database Connections', 'Database Configuration', 'Database Security', 'Database Performance', 'Database Maintenance', 'Database Tools', 'Database Reports', 'Database Administration', 'Database Dictionary', 'Database Dictionary Reports', 'Database Dictionary Models', 'Database Dictionary Reports', 'Database Dictionary Reports', 'Database Dictionary Reports'. The main window displays a SQL script in the 'Script Output' tab. The script is: 

```
CREATE VIEW DetaliiComenzi AS
SELECT u.NUME_UTILIZATOR, u.EMAIL, c.DATA_COMANDA, c.TOTAL_PLATA
FROM UTILIZATORI u
JOIN COMENZI c ON u.ID_UTILIZATOR = c.ID_UTILIZATOR;
SELECT * FROM DetaliiComenzi
```

 Below the script, a status bar indicates 'Task completed in 0,05 seconds'. The results are displayed in a table with the following data:

| NUME_UTILIZATOR | EMAIL             | DATA_COM | TOTAL_PLATA |
|-----------------|-------------------|----------|-------------|
| user1           | user1@example.com | 19.12.24 | 0           |
| user2           | user2@example.com | 19.12.24 | 250         |
| user3           | user3@example.com | 19.12.24 | 500         |
| user4           | user4@example.com | 19.12.24 | 750         |
| user5           | user5@example.com | 19.12.24 | 1000        |

#### 2. Indecși (Indexes)

*Indecșii accelerează interogările, mai ales când sunt utilizate frecvent coloane în filtre (WHERE) sau în alăturări (JOIN).*

- Coloana **DATA\_COMANDA** este utilizată frecvent pentru interogări care filtrează comenzile după dată.
- Crearea unui index pe această coloană optimizează interogările care implică sortări sau filtre pe bază de dată.

```
CREATE INDEX idx_data_comanda ON COMENZI (DATA_COMANDA);
SELECT * FROM COMENZI
WHERE DATA_COMANDA >= TO_DATE('2024-01-01', 'YYYY-MM-DD');
```

```

CREATE INDEX idx_data_comanda ON COMENZI (DATA_COMANDA);
SELECT * FROM COMENZI
WHERE DATA_COMANDA >= TO_DATE('2024-01-01', 'YYYY-MM-DD');

```

Script Output x

Task completed in 0,038 seconds

additional columns in the 'CREATE INDEX' statement.

Index IDX\_DATA\_COMANDA created.

| ID_COMANDA | ID_UTILIZATOR | DATA_COM | TOTAL_PLATA |
|------------|---------------|----------|-------------|
| 1          | 101           | 19.12.24 | 0           |
| 2          | 102           | 19.12.24 | 250         |
| 3          | 103           | 19.12.24 | 500         |
| 4          | 104           | 19.12.24 | 750         |
| 5          | 105           | 19.12.24 | 1000        |
| 6          | 106           | 20.12.24 | 150         |
| 8          | 108           | 15.12.24 | 1200        |
| 9          | 109           | 10.12.24 | 600         |
| 10         | 110           | 05.12.24 | 50          |
| 11         | 110           | 09.12.24 | 500         |

10 rows selected.

### 3. Sinonime (Synonyms)

*Sinonimele sunt aliasuri pentru obiectele bazei de date. Sunt utile pentru a simplifica accesul la tabele sau alte obiecte din baze de date diferite.*

Exemplu:

```
-- Crearea unui sinonim pentru tabelul Comenzi
CREATE SYNONYM ComenziSimplu FOR Comenzi;
```

Utilizarea sinonimului:

```
SELECT * FROM ComenziSimplu;
```

```

Exemplu:
-- Crearea unui sinonim pentru tabelul Comenzi
CREATE SYNONYM ComenziSimplu FOR Comenzi;

Utilizarea sinonimului:
SELECT * FROM ComenziSimplu;

```

Script Output x

Task completed in 0,127 seconds

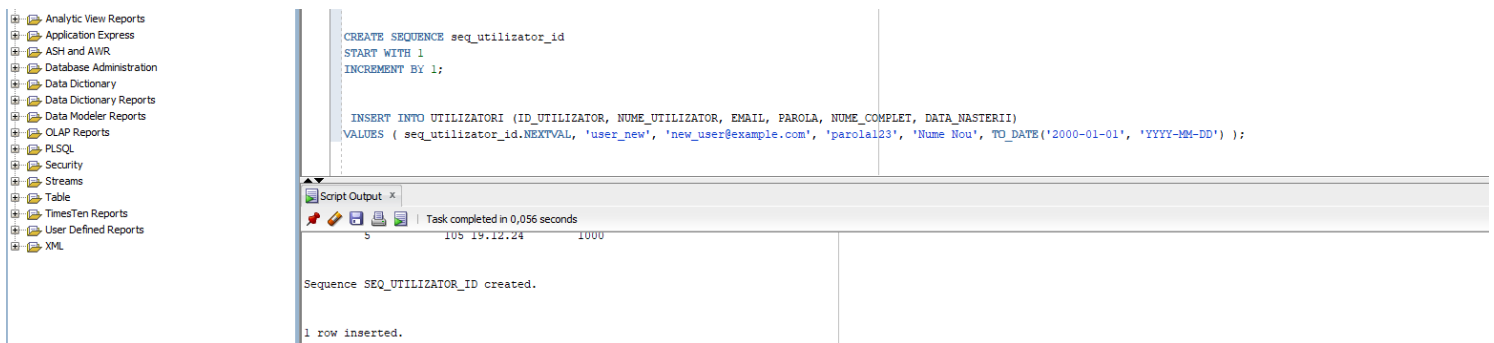
| ID_COMANDA | ID_UTILIZATOR | DATA_COM | TOTAL_PLATA |
|------------|---------------|----------|-------------|
| 1          | 101           | 19.12.24 | 0           |
| 2          | 102           | 19.12.24 | 250         |
| 3          | 103           | 19.12.24 | 500         |
| 4          | 104           | 19.12.24 | 750         |
| 5          | 105           | 19.12.24 | 1000        |

## 4. Secvențe (Sequences)

*Secvențele generează valori unice, utile pentru cheile primare.*

Exemplu:

```
-- Crearea unei secvențe pentru generarea automată a ID-urilor  
utilizatorilor  
CREATE SEQUENCE seq_utilizator_id  
START WITH 1  
INCREMENT BY 1;  
  
-- Inserarea unui utilizator nou folosind secvența pentru ID  
INSERT INTO UTILIZATORI (ID_UTILIZATOR, NUME_UTILIZATOR, EMAIL,  
PAROLA, NUME_COMPLET, DATA_NASTERII)  
VALUES ( seq_utilizator_id.NEXTVAL, 'user_new',  
'new_user@example.com', 'parola123', 'Nume Nou', TO_DATE('2000-01-01',  
'YYYY-MM-DD') );
```



## 5. Alte Obiecte: Triggere

*Triggererele sunt utilizate pentru a automatiza acțiuni la modificarea datelor (inserare, actualizare, ștergere).*

Exemplu:

```
-- Crearea unui trigger care verifică dacă suma comenzii este  
mai mare decât 0  
CREATE OR REPLACE TRIGGER trg_verificare_suma  
BEFORE INSERT OR UPDATE ON Comenzi  
FOR EACH ROW  
BEGIN  
    IF :NEW.Suma <= 0 THEN  
        RAISE_APPLICATION_ERROR(-20001, 'Suma comenzii trebuie  
să fie mai mare decât 0!');  
    END IF;  
END;
```

Materialized Views

Reports

- All Reports
- About Your Database
- All Objects
- Analytic View Reports
- Application Express
- ASH and AWR
- Database Administration
- Data Dictionary
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- PLSQL
- Security
- Streams
- Table
- TimesTen Reports
- User Defined Reports
- XML

```
CREATE OR REPLACE TRIGGER trg_verificare_suma
BEFORE INSERT OR UPDATE ON Comenzi
FOR EACH ROW
BEGIN
    IF :NEW.TOTAL_PLATA <= 0 THEN
        RAISE_APPLICATION_ERROR(-20001, 'Suma comenzii trebuie să fie mai mare decât 0!');
    END IF;
END;

INSERT INTO Comenzi (ID_COMANDA, ID_UTILIZATOR, DATA_COMANDA, TOTAL_PLATA)
VALUES (1, 101, SYSDATE, -50);
```

Script Output

Task completed in 0,08 seconds

VALUES (1, 101, SYSDATE, -50)

Error at Command Line : 309 Column : 13

Error report -

SQL Error: ORA-20001: Suma comenzii trebuie să fie mai mare decât 0!