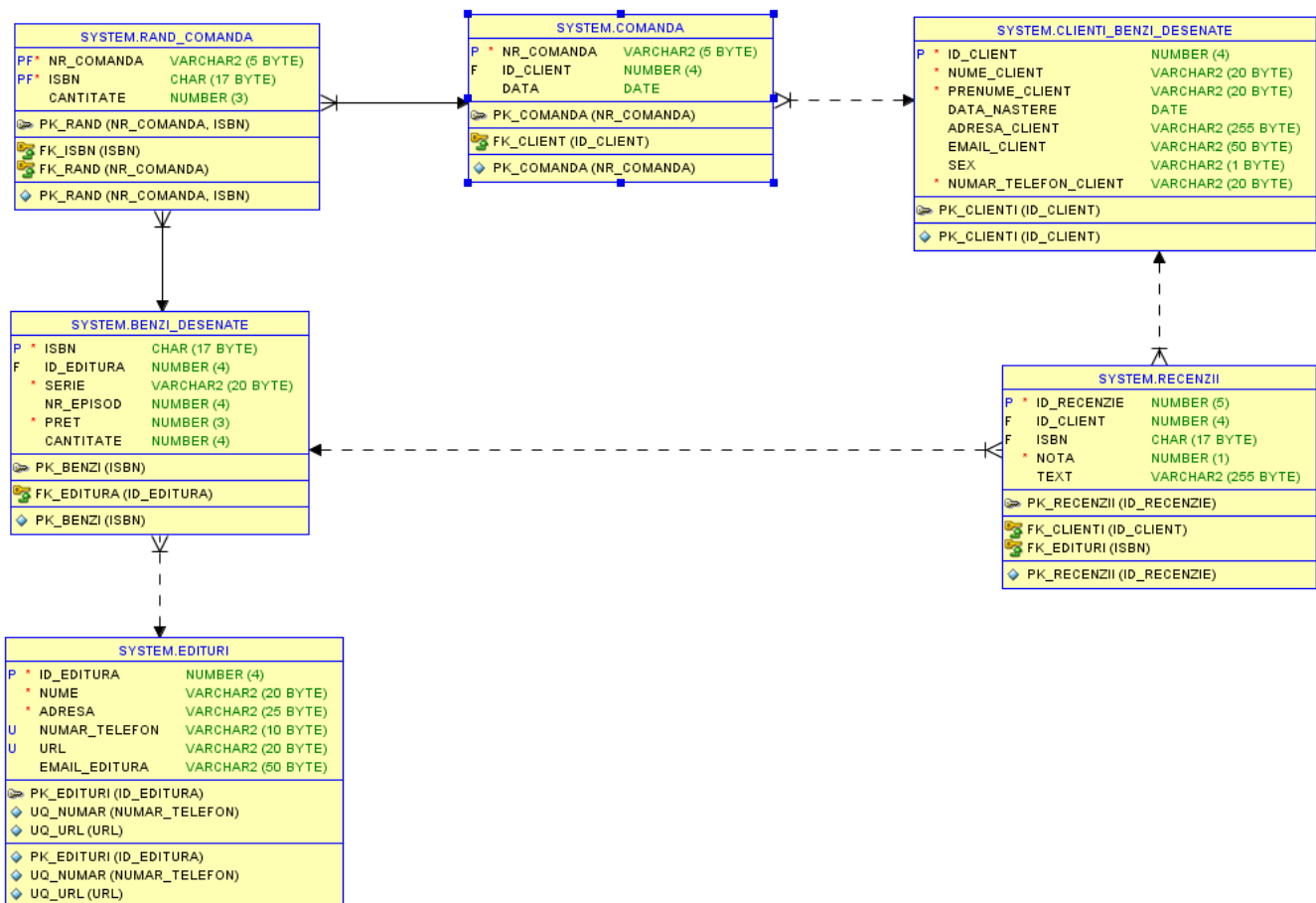


Grupa: 1046

Proiectul tratează gestiunea activității unui magazin online de benzi desenate.

Astfel, un client poate cumpăra prin intermediul aplicației online benzi desenate provenind de la diferite edituri, făcând parte din serii și episoade diferite într-una sau mai multe comenzi.

Diagrama entitate-asociere finală:



--TEMA1:

-->Creați tabelele aferente BD de la proiect

-->Adăugați date în tabelele create.

--creare tabele

```
CREATE TABLE edituri (  
    id_editura NUMBER(4),  
    nume VARCHAR2(20) CONSTRAINT nn_nume NOT NULL,  
    adresa VARCHAR2(25) CONSTRAINT nn_prenume NOT NULL,  
    numar_telefon VARCHAR2(10) CONSTRAINT uq_numar UNIQUE,  
    url VARCHAR2(20) CONSTRAINT uq_url UNIQUE,  
    CONSTRAINT pk_edituri PRIMARY KEY (id_editura)  
);
```

```
CREATE TABLE clienti_benzi_desenate (  
    id_client NUMBER(4),  
    nume_client VARCHAR2(20) CONSTRAINT nn_nume_client NOT NULL,  
    prenume_client VARCHAR2(20) CONSTRAINT nn_prenume_client NOT NULL,  
    data_nastere DATE,  
    adresa_client VARCHAR(255),  
    email_client VARCHAR2(50),  
    sex VARCHAR2(1),  
    numar_telefon_client VARCHAR2(20) CONSTRAINT nn_numar_client NOT NULL,  
    CONSTRAINT pk_clienti PRIMARY KEY (id_client),  
    CONSTRAINT check_email CHECK(email_client LIKE '%@%.%')  
);
```

```
CREATE TABLE comanda (  
    nr_comanda VARCHAR2(5),  
    id_client NUMBER(4),  
    data DATE,  
    CONSTRAINT pk_comanda PRIMARY KEY (nr_comanda),  
    CONSTRAINT fk_client FOREIGN KEY (id_client) REFERENCES clienti_benzi_desenate(id_client)  
);
```

```
CREATE TABLE benzi_desenate (  
    isbn CHAR(10),  
    id_editura NUMBER(4),  
    serie VARCHAR2(20) CONSTRAINT nn_serie NOT NULL,  
    nr_episod NUMBER(4),  
    pret NUMBER(3) CONSTRAINT nn_pret NOT NULL,  
    cantitate NUMBER(4),  
    CONSTRAINT pk_benzi PRIMARY KEY (isbn),  
    CONSTRAINT fk_editura FOREIGN KEY (id_editura) REFERENCES edituri(id_editura),  
    CONSTRAINT pret CHECK(pret > 0)  
);
```

```
CREATE TABLE rand_comanda (  
    nr_comanda VARCHAR2(5),  
    isbn CHAR(10),
```

```

cantitate NUMBER(3),
CONSTRAINT pk_rand PRIMARY KEY (nr_comanda, isbn),
CONSTRAINT fk_rand FOREIGN KEY (nr_comanda) REFERENCES comanda(nr_comanda),
CONSTRAINT fk_isbn FOREIGN KEY (isbn) REFERENCES benzi_desenate(isbn)
);

CREATE TABLE recenzii (
    id_recenzie number(5),
    id_client number(4),
    isbn char(10),
    nota number(1) CONSTRAINT nn_nota NOT NULL,
    text varchar2(255),
    CONSTRAINT pk_recenzii PRIMARY KEY (id_recenzie),
    CONSTRAINT fk_edituri FOREIGN KEY (isbn) REFERENCES benzi_desenate(isbn),
    CONSTRAINT fk_clienti FOREIGN KEY (id_client) REFERENCES clienti_benzi_desenate(id_client)
);

--clienti
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('100', 'Dobrescu', 'Marius', TO_DATE('26.03.2000', 'DD.MM.YYYY'), 'Marinarilor 7', 'd
obrescu@gmail.com', 'M', '0724456710');

INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('101', 'Dorian', 'Maria', TO_DATE('16.04.1993', 'DD.MM.YYYY'), 'Mercenarilor 76', 'do
rian@gmail.com', 'F', '0709456710');

INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('102', 'Matei', 'Bianca', TO_DATE('20.05.2001', 'DD.MM.YYYY'), 'Violetelor 2', 'mateo@
gmail.com', 'F', '0724418710');

INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('103', 'Popescu', 'Ioana', TO_DATE('16.05.2003', 'DD.MM.YYYY'), 'Zambilelor 5', 'pope
scu@gmail.com', 'F', '0784456710');

INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('104', 'Filipescu', 'Vlad', TO_DATE('01.01.1997', 'DD.MM.YYYY'), 'Infratirii 4', 'fil
ipescu@gmail.com', 'M', '0724456900');

INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('105', 'Mardare', 'Andrei', TO_DATE('02.06.1998', 'DD.MM.YYYY'), 'Ionescu Brad 3', 'm
ardare@gmail.com', 'M', '0721226710');

INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)

```

```
VALUES ('106', 'Vladescu', 'Mihai', TO_DATE('05.07.2006', 'DD.MM.YYYY'), 'Carmangeriei 2', 'vladescu@gmail.com', 'M', '0724412310');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('107', 'Coriolian', 'Ion', TO_DATE('20.08.2004', 'DD.MM.YYYY'), 'Bisericii 666', 'coriolian@gmail.com', 'M', '0724246710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('108', 'Caldare', 'Iulian', TO_DATE('06.10.2005', 'DD.MM.YYYY'), 'Uneltelor 2', 'caldare@gmail.com', 'M', '0739456710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('109', 'Cristian', 'Mircea', TO_DATE('21.12.2005', 'DD.MM.YYYY'), 'Trandafirilor 3', 'cristian@gmail.com', 'M', '0724457891');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('110', 'Onetiu', 'Valentin', TO_DATE('30.11.1983', 'DD.MM.YYYY'), 'Revolutiei 89', 'onetiu@gmail.com', 'M', '0739956710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('111', 'Casin', 'Diana', TO_DATE('11.08.1956', 'DD.MM.YYYY'), 'Craciunului 25', 'casin@gmail.com', 'F', '0724402710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('112', 'Mircea', 'Andreea', TO_DATE('12.06.1981', 'DD.MM.YYYY'), 'Pastelui 3', 'mircea@gmail.com', 'F', '0724450110');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('113', 'Zlatan', 'Ibrahimovic', TO_DATE('02.05.1989', 'DD.MM.YYYY'), 'Soldatilor 4', 'zlatan@gmail.com', 'M', '070016710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('114', 'Lorin', 'Catalin', TO_DATE('18.04.1994', 'DD.MM.YYYY'), 'Primariei 5', 'lorin@gmail.com', 'M', '0720036710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('115', 'Dobrescu', 'Veronica', TO_DATE('07.09.1976', 'DD.MM.YYYY'), 'Principala 10', 'dobrescuv@gmail.com', 'F', '0724401710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
```

```
VALUES ('116', 'Georgescu', 'Mariana', TO_DATE('09.10.1987', 'DD.MM.YYYY'), 'Dorobanti 1', 'georgescu@gmail.com', 'F', '0726675710');
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADRESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)  
VALUES ('117', 'Mancu', 'Matei', TO_DATE('13.07.1990', 'DD.MM.YYYY'), 'Protopopescu 19', 'mancu@gmail.com', 'M', '0724491710');
```

```
--edituri
```

```
ALTER TABLE edituri
```

```
ADD email_editura VARCHAR2(50);
```

```
INSERT INTO edituri (ID_EDITURA, NUME, ADRESA, EMAIL_EDITURA, NUMAR_TELEFON, URL)  
VALUES ('1000', 'ARTHUR', 'Tipografiei 1', 'arthur@gmail.com', '0212456731', 'www.arthur.ro');
```

```
INSERT INTO edituri (ID_EDITURA, NUME, ADRESA, EMAIL_EDITURA, NUMAR_TELEFON, URL)  
VALUES ('1001', 'KOOB', 'Tipografiei 2', 'koob@gmail.com', '0212493231', 'www.koob.ro');
```

```
INSERT INTO edituri (ID_EDITURA, NUME, ADRESA, EMAIL_EDITURA, NUMAR_TELEFON, URL)  
VALUES ('1002', 'CRISAN', 'Tipografiei 3', 'crisan@gmail.com', '0212409121', 'www.crisan.ro');
```

```
INSERT INTO edituri (ID_EDITURA, NUME, ADRESA, EMAIL_EDITURA, NUMAR_TELEFON, URL)  
VALUES ('1003', 'DC_COMICS', 'Tipografiei 4', 'dccomics@gmail.com', '0210923731', 'www.dccomics.com');
```

```
INSERT INTO edituri (ID_EDITURA, NUME, ADRESA, EMAIL_EDITURA, NUMAR_TELEFON, URL)  
VALUES ('1004', 'MARVEL', 'Tipografiei 5', 'marvel@gmail.com', '0212412331', 'www.marvel.com')  
;
```

```
--benzi desenate
```

```
ALTER TABLE benzi_desenate
```

```
MODIFY isbn char(17);
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)  
VALUES ('978-3-16-148410-0', '1003', 'SPIDER-MAN', '1', '15', '10');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)  
VALUES ('978-3-16-148411-9', '1003', 'SPIDER-MAN', '2', '15', '9');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)  
VALUES ('978-3-16-148411-8', '1003', 'SPIDER-MAN', '3', '15', '8');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)  
VALUES ('978-3-16-148411-7', '1003', 'SPIDER-MAN', '4', '15', '6');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)  
VALUES ('978-3-16-148411-6', '1003', 'SPIDER-MAN', '5', '15', '3');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)  
VALUES ('978-3-16-148411-5', '1003', 'SPIDER-MAN', '6', '15', '16');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148411-4', '1003', 'SPIDER-MAN', '7', '15', '13');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148411-3', '1003', 'SPIDER-MAN', '8', '15', '1');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148411-2', '1003', 'SPIDER-MAN', '9', '15', '29');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148411-1', '1003', 'SPIDER-MAN', '10', '15', '7');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148411-0', '1003', 'SPIDER-MAN', '11', '16', '6');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-9', '1003', 'SPIDER-MAN', '12', '16', '5');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-8', '1003', 'SPIDER-MAN', '13', '16', '11');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-7', '1003', 'SPIDER-MAN', '14', '16', '21');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-6', '1003', 'SPIDER-MAN', '15', '16', '10');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-5', '1003', 'SPIDER-MAN', '16', '16', '40');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-4', '1003', 'SPIDER-MAN', '17', '16', '10');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-3', '1003', 'SPIDER-MAN', '18', '16', '45');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-2', '1003', 'SPIDER-MAN', '19', '16', '30');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-148410-1', '1003', 'SPIDER-MAN', '20', '16', '21');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-018410-1', '1004', 'IRON-MAN', '1', '16', '21');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-028410-1', '1004', 'IRON-MAN', '2', '16', '20');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
```

```
VALUES ('978-3-16-038410-1', '1004', 'IRON-MAN', '3', '16', '10');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-048410-1', '1004', 'IRON-MAN', '4', '16', '15');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-058410-1', '1004', 'IRON-MAN', '5', '16', '16');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-068410-1', '1004', 'IRON-MAN', '6', '16', '9');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-078410-1', '1004', 'IRON-MAN', '7', '16', '19');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-088410-1', '1004', 'IRON-MAN', '8', '16', '20');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-098410-1', '1004', 'IRON-MAN', '9', '16', '15');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-108410-1', '1004', 'IRON-MAN', '10', '16', '12');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-101410-1', '1002', 'PEANUTS', '10', '16', '12');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-102410-1', '1002', 'PEANUTS', '11', '16', '7');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-103410-1', '1002', 'PEANUTS', '5', '16', '10');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-101410-1', '1001', 'HILLBILLIES', '56', '10', '8');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-103490-1', '1001', 'HILLBILLIES', '23', '10', '10');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-112410-1', '1001', 'HILLBILLIES', '67', '10', '1');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('978-3-16-101423-1', '1001', 'HILLBILLIES', '65', '9', '1');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('901-3-16-101410-1', '1000', 'MACHINE-MAN', '1', '18', '11');

INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('902-3-16-101410-1', '1000', 'MACHINE-MAN', '2', '18', '1');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('903-3-16-101410-1', '1000', 'MACHINE-MAN', '3', '18', '32');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('904-3-16-101410-1', '1000', 'MACHINE-MAN', '4', '18', '2');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('905-3-16-101410-1', '1000', 'MACHINE-MAN', '5', '18', '20');
```

```
INSERT INTO benzi_desenate (ISBN, ID_EDITURA, SERIE, NR_EPISOD, PRET, CANTITATE)
VALUES ('906-3-16-101410-1', '1000', 'MACHINE-MAN', '6', '18', '10');
```

--comanda

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('1', '101', TO_DATE('21.02.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('2', '102', TO_DATE('11.03.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('3', '104', TO_DATE('19.03.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('4', '103', TO_DATE('20.03.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('5', '101', TO_DATE('17.04.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('6', '103', TO_DATE('18.05.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('7', '115', TO_DATE('19.05.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('8', '116', TO_DATE('1.06.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('9', '110', TO_DATE('28.06.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('10', '101', TO_DATE('09.07.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('11', '108', TO_DATE('09.07.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('12', '117', TO_DATE('20.09.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
```



```
VALUES ('13', '116', TO_DATE('25.12.2019', 'DD.MM.YYYY'));
```

```
INSERT INTO comanda (NR_COMANDA, ID_CLIENT, DATA)
VALUES ('14', '107', TO_DATE('27.12.2019', 'DD.MM.YYYY'));
```

```
--rand_comanda
```

```
ALTER TABLE rand_comanda
MODIFY isbn char(17);
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('1', '978-3-16-108410-1', '7');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('1', '978-3-16-148411-5', '3');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('2', '903-3-16-101410-1', '2');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('3', '978-3-16-101410-1', '12');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('4', '978-3-16-148411-2', '11');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('4', '978-3-16-048410-1', '10');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('4', '978-3-16-148411-5', '1');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('5', '978-3-16-148411-5', '10');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('6', '978-3-16-148411-7', '3');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('7', '978-3-16-102410-1', '1');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('8', '978-3-16-148411-9', '1');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('9', '903-3-16-101410-1', '5');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('10', '906-3-16-101410-1', '3');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('11', '978-3-16-101423-1', '19');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('12', '978-3-16-098410-1', '24');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('13', '978-3-16-078410-1', '14');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('14', '978-3-16-108410-1', '5');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('14', '978-3-16-112410-1', '6');
```

```
INSERT INTO rand_comanda(NR_COMANDA, ISBN, CANTITATE)
VALUES ('14', '978-3-16-148410-0', '1');
```

```
--recenzii
```

```
ALTER TABLE recenzii
MODIFY isbn char(17);
```

```
INSERT INTO recenzii(ID_RECENZIE, ID_CLIENT, ISBN, NOTA, TEXT)
VALUES ('1', '107', '978-3-16-148410-0', '5', 'foarte fain');
```

```
INSERT INTO recenzii(ID_RECENZIE, ID_CLIENT, ISBN, NOTA, TEXT)
VALUES ('2', '107', '978-3-16-112410-1', '4', 'oarecum fain');
```

```
INSERT INTO recenzii(ID_RECENZIE, ID_CLIENT, ISBN, NOTA, TEXT)
VALUES ('3', '107', '978-3-16-108410-1', '3', 'merge');
```

```
INSERT INTO recenzii(ID_RECENZIE, ID_CLIENT, ISBN, NOTA, TEXT)
VALUES ('4', '101', '978-3-16-108410-1', '3', '');
```

```
INSERT INTO recenzii(ID_RECENZIE, ID_CLIENT, ISBN, NOTA, TEXT)
VALUES ('5', '101', '978-3-16-148411-5', '1', 'nu prea mi-a placut, baieti');
```

```
INSERT INTO recenzii(ID_RECENZIE, ID_CLIENT, ISBN, NOTA, TEXT)
VALUES ('6', '115', '978-3-16-102410-1', '5', 'decent');
```

```
--TEMA2:  
-->min. 5 operații de actualizare pe BD de la proiect  
-->min. 5 interogări pe BD de la proiect (cu joncțiuni și grupări)
```

```
--actualizare
```

```
ALTER TABLE comanda  
RENAME TO comanda_benzi_desenate;
```

```
ALTER TABLE edituri  
RENAME TO edituri_benzi_desenate;
```

```
ALTER TABLE rand_comanda  
RENAME TO rand_comanda_benzi_desenate;
```

```
ALTER TABLE recenzii  
RENAME TO recenzii_benzi_desenate;
```

```
ALTER TABLE benzi_desenate  
MODIFY serie DEFAULT 'Serie necunoscuta';
```

```
ALTER TABLE benzi_desenate  
ADD CONSTRAINT nn_seria check (serie IS NOT NULL);
```

```
ALTER TABLE benzi_desenate  
DROP CONSTRAINT nn_seria;
```

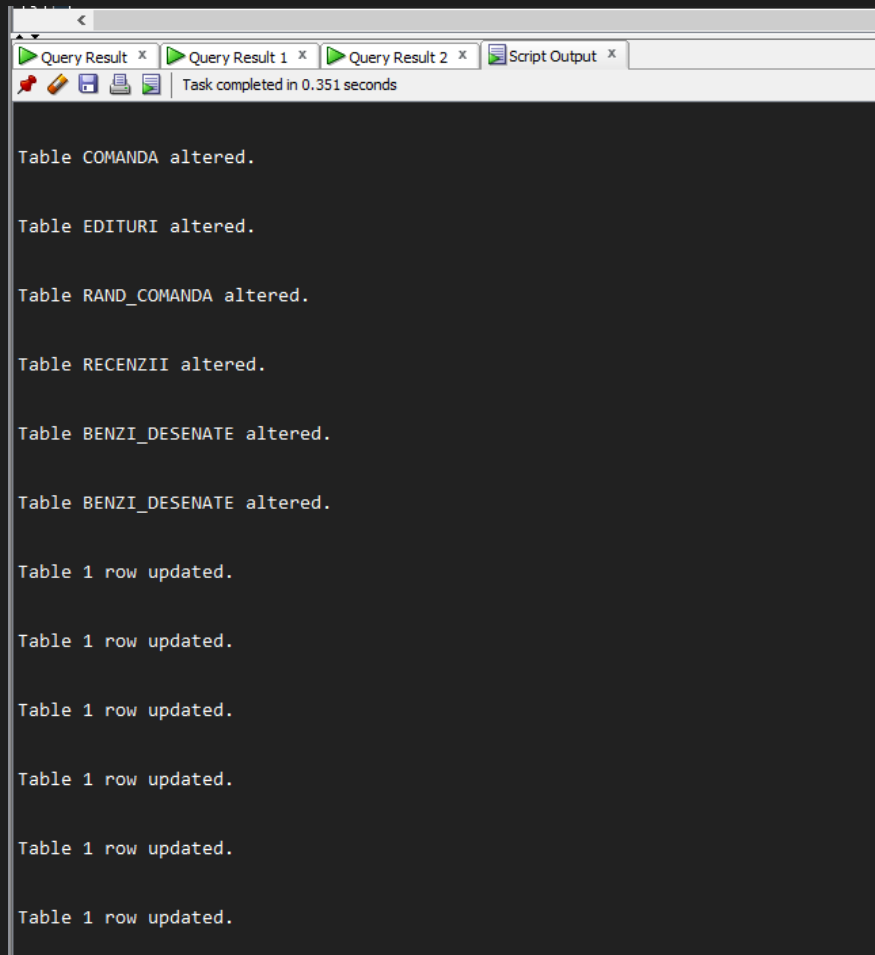
```
UPDATE clienti_benzi_desenate  
SET email_client = 'mgeorgescu@gmail.com'  
WHERE id_client = 116;
```

```
UPDATE clienti_benzi_desenate  
SET adresa_client = 'Razbunarii 5'  
WHERE id_client = 117;
```

```
UPDATE clienti_benzi_desenate  
SET prenume_client = 'George'  
WHERE id_client = 106;
```

```
UPDATE clienti_benzi_desenate  
SET numar_telefon_client = '0721246903'  
WHERE id_client = 108;
```

```
UPDATE clienti_benzi_desenate  
SET numar_telefon_client = '0771246953'  
WHERE nume_client = 'Zlatan';
```



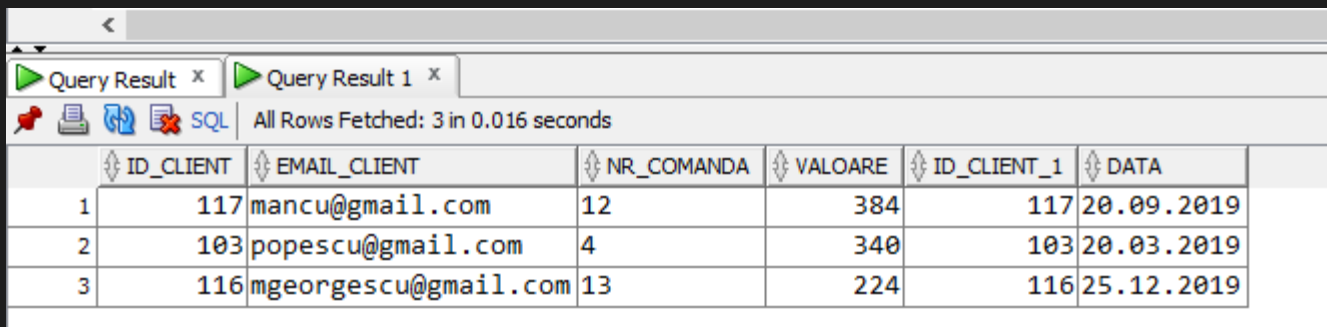
--interogari

--1. Sa se afiseze clientii care au efectuat comenzi

```
SELECT cl.id_client, UPPER(cl.ume_client), cl.prenume_client
FROM clienti_benzi_desenate cl, comanda_benzi_desenate co
WHERE cl.id_client = co.id_client
GROUP BY cl.id_client, cl.prenume_client, UPPER(cl.ume_client)
ORDER BY cl.id_client;
```

	ID_CLIENT	UPPER(CL.UME_CLIENT)	PRENUME_CLIENT
1	101	DORIAN	Maria
2	102	MATEI	Bianca
3	103	POPESCU	Ioana
4	104	FILIPESCU	Vlad
5	107	CORIOLIAN	Ion
6	108	CALDARE	Iulian
7	110	ONETIU	Valentin
8	115	DOBRESCU	Veronica
9	116	GEORGESCU	Mariana
10	117	MANCU	Matei

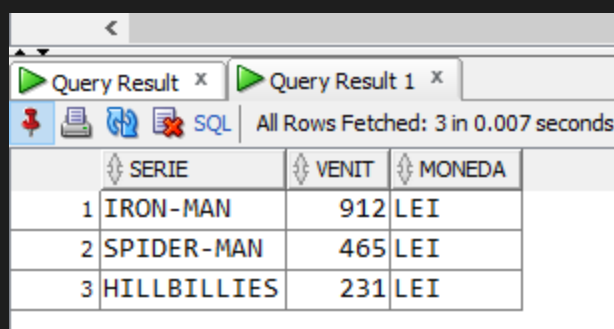
```
--2. Sa se afiseze adresa de mail si id-
ul primilor 3 clienti care au efectuat comenzi in ordinea descrescatoare a valorii lor
SELECT c.id_client, cb.email_client , r.nr_comanda, SUM(b.pret*r.cantitate) as valoare, c.id_
client, TO_CHAR(c.data, 'DD.MM.YYYY') AS data
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi
_desenate cb
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client
GROUP BY r.nr_comanda, c.id_client, cb.ume_client, cb.email_client,
TO_CHAR(c.data, 'DD.MM.YYYY')
ORDER BY SUM(b.pret*r.cantitate) DESC
FETCH FIRST 3 ROWS ONLY;
```



Query Result 1 x All Rows Fetched: 3 in 0.016 seconds

	ID_CLIENT	EMAIL_CLIENT	NR_COMANDA	VALOARE	ID_CLIENT_1	DATA
1	117	mancu@gmail.com	12	384	117	20.09.2019
2	103	popescu@gmail.com	4	340	103	20.03.2019
3	116	mgeorgescu@gmail.com	13	224	116	25.12.2019

```
--3. Sa se afiseze cele mai cumparate 3 serii de benzi desenate si
veniturile rezultate din vanzarea acestora
SELECT b.serie, SUM(r.cantitate * b.pret) as venit, UPPER('LEI') moneda
FROM rand_comanda_benzi_desenate r, benzi_desenate b
WHERE b.isbn = r.isbn
GROUP BY b.serie
ORDER BY SUM(r.cantitate) DESC
FETCH FIRST 3 ROWS ONLY;
```



Query Result 1 x All Rows Fetched: 3 in 0.007 seconds

	SERIE	VENIT	MONEDA
1	IRON-MAN	912	LEI
2	SPIDER-MAN	465	LEI
3	HILLBILLIES	231	LEI

```
--4. Sa se afiseze id-
ul editurii, numele editurii, emailul editurii si sumele datorate daca pretul la care au fost
vandute benzile desenate = pret editura + TVA(5%) + adaos comercial(10%)
SELECT b.id_editura, e.numa, e.email_editura, (SUM(r.cantitate * b.pret) -
0.1 * SUM(r.cantitate * b.pret) - 0.5 * (SUM(r.cantitate * b.pret) -
0.1 * SUM(r.cantitate * b.pret))) suma_datorata_editura
FROM rand_comanda_benzi_desenate r, benzi_desenate b, edituri_benzi_desenate e
WHERE b.isbn = r.isbn AND b.id_editura = e.id_editura
GROUP BY b.id_editura, e.numa, e.email_editura
```

ORDER BY b.id_editura DESC;

	ID_EDITURA	NUME	EMAIL_EDITURA	SUMA_DATORATA_EDITURA
1	1004	MARVEL	marvel@gmail.com	410.4
2	1003	DC_COMICS	dccomics@gmail.com	209.25
3	1002	CRISAN	crisan@gmail.com	93.6
4	1001	KOOB	koob@gmail.com	103.95
5	1000	ARTHUR	arthur@gmail.com	81

```
--5. Sa se afiseze date despre toti clientii care au facut comenzi in luna martie
SELECT cl.id_client, UPPER(cl.ume_client), cl.prenume_client, cl.email_client, cl.numar_telefon_client, co.data
FROM clienti_benzi_desenate cl, comanda_benzi_desenate co
WHERE cl.id_client = co.id_client
GROUP BY cl.id_client, cl.prenume_client, UPPER(cl.ume_client), co.data, cl.email_client, cl.numar_telefon_client
HAVING EXTRACT(MONTH FROM co.data)=3
ORDER BY co.data;
```

	ID_CLIENT	UPPER(CL.NUME_CLIENT)	PRENUME_CLIENT	EMAIL_CLIENT	NUMAR_TELEFON_CLIENT	DATA
1	102	MATEI	Bianca	mateo@gmail.com	0724418710	11-MAR-19
2	104	FILIPESCU	Vlad	filipescu@gmail.com	0724456900	19-MAR-19
3	103	POPESCU	Ioana	popescu@gmail.com	0784456710	20-MAR-19

```
--6. Sa se afiseze clientii care au facut comenzi in luna in care sunt nascuti
SELECT cl.id_client, cl.ume_client, cl.prenume_client, cl.data_nastere, co.data data_comanda
FROM clienti_benzi_desenate cl, comanda_benzi_desenate co
WHERE cl.id_client = co.id_client
GROUP BY cl.id_client, co.data, cl.data_nastere, cl.prenume_client, cl.ume_client
HAVING EXTRACT(MONTH FROM co.data) = EXTRACT(MONTH FROM cl.data_nastere)
ORDER BY id_client;
```

	ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	DATA_NASTERE	DATA_COMANDA
1	101	Dorian	Maria	16-APR-93	17-APR-19
2	103	Popescu	Ioana	16-MAY-03	18-MAY-19

--TEMA3:

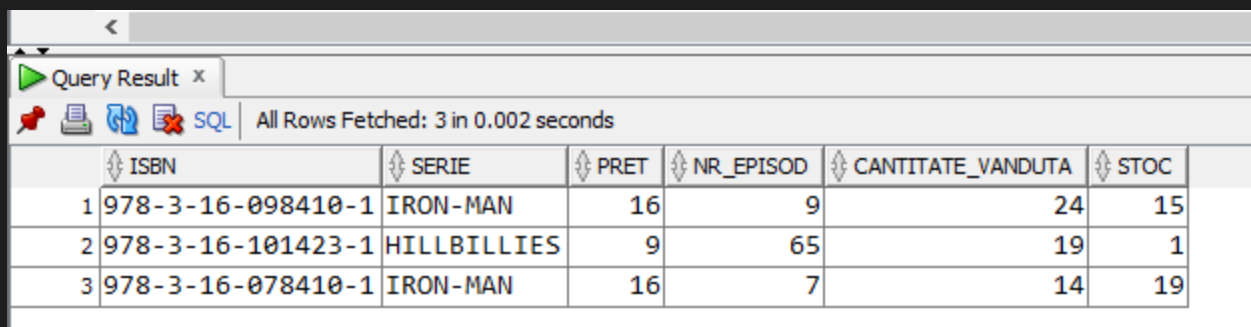
-->min. 3 interogări cu cereri ierarhice (dacă niciuna dintre tabelele voastre nu este construită astfel încât să permită parcurgere ierarhică, atunci construiți min. 3 interogări cu grupări diverse)

-->min. 5 interogări cu subcereri

--interogari cu grupari diverse

--1. Sa se afiseze primele 3 cel mai bine vandute editii si stocul

```
SELECT b.isbn, b.serie, b.pret, b.nr_episod, r.cantitate cantitate_vanduta, b.cantitate stoc
FROM rand_comanda_benzi_desenate r, benzi_desenate b
WHERE b.isbn = r.isbn
GROUP BY b.isbn, b.serie, b.pret, b.nr_episod, r.cantitate, b.cantitate
ORDER BY r.cantitate DESC
FETCH FIRST 3 ROWS ONLY;
```



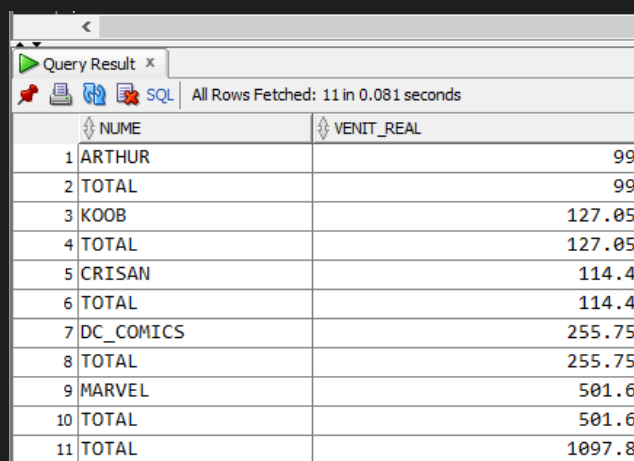
Query Result x

All Rows Fetched: 3 in 0.002 seconds

	ISBN	SERIE	PRET	NR_EPISOD	CANTITATE_VANDUTA	STOC
1	978-3-16-098410-1	IRON-MAN	16	9	24	15
2	978-3-16-101423-1	HILLBILLIES	9	65	19	1
3	978-3-16-078410-1	IRON-MAN	16	7	14	19

--2. Sa se afiseze venitul real rezultat in urma comenzilor deja efectuate

```
SELECT CASE GROUPING(e.ume)
WHEN 1 THEN 'TOTAL' ELSE e.ume END ume,
SUM(r.cantitate * b.pret) - (SUM(r.cantitate * b.pret) - 0.1 * SUM(r.cantitate * b.pret) -
0.5 * (SUM(r.cantitate * b.pret) - 0.1 * SUM(r.cantitate * b.pret))) venit_real
FROM rand_comanda_benzi_desenate r, benzi_desenate b, edituri_benzi_desenate e
WHERE b.isbn = r.isbn AND b.id_editura = e.id_editura
GROUP BY ROLLUP(b.id_editura, e.ume)
ORDER BY b.id_editura ASC;
```



Query Result x

All Rows Fetched: 11 in 0.081 seconds

	NUME	VENIT_REAL
1	ARTHUR	99
2	TOTAL	99
3	KOOB	127.05
4	TOTAL	127.05
5	CRISAN	114.4
6	TOTAL	114.4
7	DC_COMICS	255.75
8	TOTAL	255.75
9	MARVEL	501.6
10	TOTAL	501.6
11	TOTAL	1097.8

--3. Sa se afiseze toate editiile al caror stoc poate fi suplimentat. CONDITII: Sa se fi vandut minim 5 exemplare, iar stocul sa fie sub 10 unitati. Sa se afiseze cantitatile care pot fi comandate judecand dupa cantitatea vanduta si coturile

```
SELECT b.isbn, b.serie, b.pret, b.nr_episod, r.cantitate cantitate_vanduta, b.cantitate stoc,
CASE
WHEN r.cantitate < 5 THEN 1
WHEN r.cantitate > 5 AND r.cantitate < 10 THEN 5
WHEN r.cantitate > 10 AND r.cantitate < 20 THEN 10
ELSE 0 END cantitate_de_comandat
FROM rand_comanda_benzi_desenate r, benzi_desenate b
WHERE b.isbn = r.isbn
GROUP BY b.isbn, b.serie, b.pret, b.nr_episod, r.cantitate, b.cantitate
HAVING b.cantitate < 10 AND r.cantitate > 5
ORDER BY r.cantitate DESC;
```

	ISBN	SERIE	PRET	NR_EPISOD	CANTITATE_VANDUTA	STOC	CANTITATE_DE_COMANDAT
1	978-3-16-101423-1	HILLBILLIES		9	65	19	1
2	978-3-16-112410-1	HILLBILLIES		10	67	6	1

--subcereri

--1. Sa se afiseze toti clientii care nu au efectuat nicio comanda

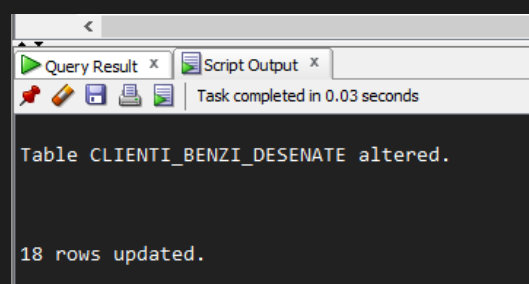
```
SELECT id_client, nume_client, prenume_client
FROM clienti_benzi_desenate
WHERE id_client NOT IN
(SELECT cl.id_client
FROM clienti_benzi_desenate cl, comanda_benzi_desenate co
WHERE cl.id_client = co.id_client);
```

	ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT
1	113	Zlatan	Ibrahimovic
2	109	Cristian	Mircea
3	105	Mardare	Andrei
4	112	Mircea	Andreea
5	111	Casin	Diana
6	114	Lorin	Catalin
7	106	Vladescu	George
8	100	Dobrescu	Marius

--2. Sa se adauge o coloana in tabela clienti_benzi_desenate (discount) prin care fiecare client va fi premiat cu un discount calculat astfel:
--Daca a cheltuit mai mult de 50 lei, va primi o reducere suplimentara de 5% pentru urmatoarea comanda
--Daca suma cheltuita depaseste 100 lei, va primi o reducere de 10% pentru urmatoarea comanda
--Daca suma cheltuita depaseste 150 lei, va primi o reducere de 20% pentru urmatoarea comanda

```
ALTER TABLE clienti_benzi_desenate  
ADD discount VARCHAR2(3);
```

```
UPDATE clienti_benzi_desenate  
SET discount =  
CASE  
WHEN id_client IN (  
SELECT c.id_client  
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi_desenate cb  
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client  
GROUP BY c.id_client  
HAVING SUM(b.pret*r.cantitate) BETWEEN 50 AND 100)  
THEN 0.05  
  
WHEN id_client IN (  
SELECT c.id_client  
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi_desenate cb  
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client  
GROUP BY c.id_client  
HAVING SUM(b.pret*r.cantitate) BETWEEN 100 AND 160)  
THEN 0.1  
  
WHEN id_client IN (  
SELECT c.id_client  
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi_desenate cb  
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client  
GROUP BY c.id_client  
HAVING SUM(b.pret*r.cantitate) BETWEEN 160 AND 1000)  
THEN 0.2  
ELSE 0  
END;
```



CLIENTI_BENZI_DESENATE									
ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	DATA_NASTERE	ADRESA_CLIENT	EMAIL_CLIENT	SEX	NUMAR_TELEFON_CLIENT	DISCOUNT	
1	100	Dobrescu	Marius	26-MAR-00	Marinarilor 7	dobrescu@gmail.com	M	0724456710	0
2	101	Dorian	Maria	16-APR-93	Mercenarilor 76	dorian@gmail.com	F	0709456710	.2
3	102	Matei	Bianca	20-MAY-01	Violetelor 2	mateo@gmail.com	F	0724418710	0
4	103	Popescu	Ioana	16-MAY-03	Zambilelor 5	popescu@gmail.com	F	0784456710	.2
5	104	Filipescu	Vlad	01-JAN-97	Infratirii 4	filipescu@gmail.com	M	0724456900	.2
6	105	Mardare	Andrei	02-JUN-98	Ionescu Brad 3	mardare@gmail.com	M	0721226710	0
7	106	Vladescu	George	05-JUL-06	Carmangeriei 2	vladescu@gmail.com	M	0724412310	0
8	107	Coriolian	Ion	20-AUG-04	Bisericii 666	coriolian@gmail.com	M	0724246710	.1
9	108	Caldare	Iulian	06-OCT-05	Uneltelor 2	caldare@gmail.com	M	0721246903	.2
10	109	Cristian	Mircea	21-DEC-05	Trandafirilor 3	cristian@gmail.com	M	0724457891	0
11	110	Onetiu	Valentin	30-NOV-83	Revolutiei 89	onetiu@gmail.com	M	0739956710	.05
12	111	Casin	Diana	11-AUG-56	Craciunului 25	casin@gmail.com	F	0724402710	0
13	112	Mircea	Andreea	12-JUN-81	Pastelui 3	mircea@gmail.com	F	0724450110	0
14	113	Zlatan	Ibrahimovic	02-MAY-89	Soldatilor 4	zlatan@gmail.com	M	0771246953	0
15	114	Lorin	Catalin	18-APR-94	Primariei 5	lorin@gmail.com	M	0720036710	0
16	115	Dobrescu	Veronica	07-SEP-76	Principalai 10	dobrescu@gmail.com	F	0724401710	0
17	116	Georgescu	Mariana	09-OCT-87	Dorobanti 1	mgeorgescu@gmail...	F	0726675710	.2
18	117	Mancu	Matei	13-JUL-90	Razbunarii 5	mancu@gmail.com	M	0724491710	.2

--3. Sa se afiseze editiile (cantitate + detalii) care pot fi returnate editurii. CONDITII: sa nu se fi vandut niciun exemplar, stocul sa fie mai mare de 10 unitati

```

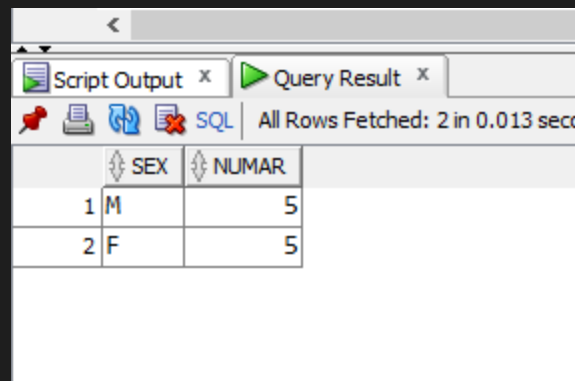
SELECT *
FROM benzi_desenate
WHERE isbn NOT IN
(SELECT b.isbn
FROM rand_comanda_benzi_desenate r, benzi_desenate b
WHERE b.isbn = r.isbn
GROUP BY b.isbn)
AND cantitate > 10;

```

ISBN	ID_EDITURA	SERIE	NR_EPISOD	PRET	CANTITATE
1 978-3-16-148410-7	1003	SPIDER-MAN	14	16	21
2 978-3-16-148411-4	1003	SPIDER-MAN	7	15	13
3 978-3-16-018410-1	1004	IRON-MAN	1	16	21
4 978-3-16-148410-5	1003	SPIDER-MAN	16	16	40
5 978-3-16-148410-1	1003	SPIDER-MAN	20	16	21
6 978-3-16-058410-1	1004	IRON-MAN	5	16	16
7 905-3-16-101410-1	1000	MACHINE-MAN	5	18	20
8 978-3-16-148410-2	1003	SPIDER-MAN	19	16	30
9 978-3-16-088410-1	1004	IRON-MAN	8	16	20
10 978-3-16-148410-8	1003	SPIDER-MAN	13	16	11
11 978-3-16-028410-1	1004	IRON-MAN	2	16	20
12 901-3-16-101410-1	1000	MACHINE-MAN	1	18	11
13 978-3-16-148410-3	1003	SPIDER-MAN	18	16	45

--4. Sa se afiseze numarul de comenzi efectuat de clientii de sex feminin si cei de sex masculin

```
SELECT sex, COUNT(sex) numar
FROM clienti_benzi_desenate cl1
WHERE cl1.id_client IN
(SELECT cl.id_client
FROM clienti_benzi_desenate cl, comanda_benzi_desenate co
WHERE cl.id_client = co.id_client
GROUP BY cl.id_client)
GROUP BY sex;
```

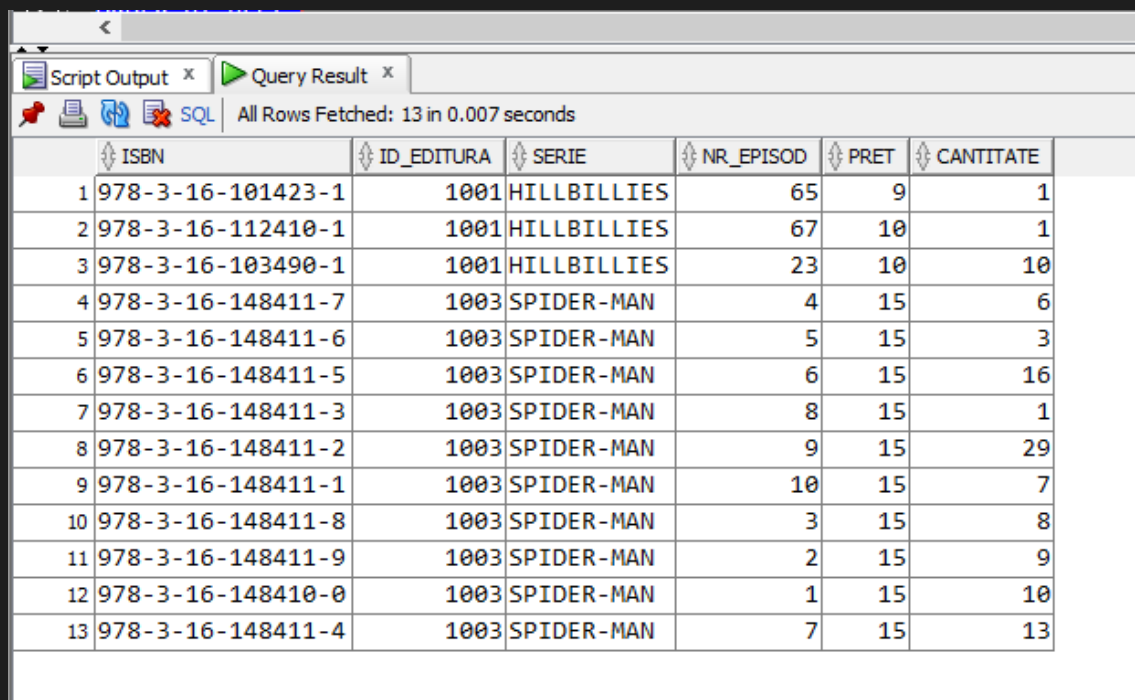


The screenshot shows a SQL query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying the results of the query. The status bar indicates 'All Rows Fetched: 2 in 0.013 seconds'. The results are shown in a table with two columns: 'SEX' and 'NUMAR'.

	SEX	NUMAR
1	M	5
2	F	5

--5. Sa se afiseze toate informatiile despre editiile care au pretul mai mic decat pretul mediu

```
SELECT *
FROM benzi_desenate
WHERE pret < (SELECT ROUND(AVG(pret), 2) FROM benzi_desenate)
ORDER BY pret;
```

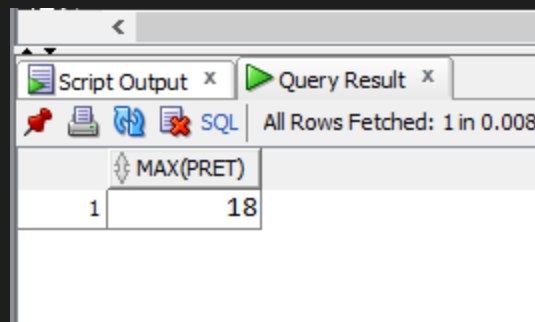


The screenshot shows a SQL query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying the results of the query. The status bar indicates 'All Rows Fetched: 13 in 0.007 seconds'. The results are shown in a table with seven columns: 'ISBN', 'ID_EDITURA', 'SERIE', 'NR_EPISOD', 'PRET', and 'CANTITATE'.

	ISBN	ID_EDITURA	SERIE	NR_EPISOD	PRET	CANTITATE
1	978-3-16-101423-1	1001	HILLBILLIES	65	9	1
2	978-3-16-112410-1	1001	HILLBILLIES	67	10	1
3	978-3-16-103490-1	1001	HILLBILLIES	23	10	10
4	978-3-16-148411-7	1003	SPIDER-MAN	4	15	6
5	978-3-16-148411-6	1003	SPIDER-MAN	5	15	3
6	978-3-16-148411-5	1003	SPIDER-MAN	6	15	16
7	978-3-16-148411-3	1003	SPIDER-MAN	8	15	1
8	978-3-16-148411-2	1003	SPIDER-MAN	9	15	29
9	978-3-16-148411-1	1003	SPIDER-MAN	10	15	7
10	978-3-16-148411-8	1003	SPIDER-MAN	3	15	8
11	978-3-16-148411-9	1003	SPIDER-MAN	2	15	9
12	978-3-16-148410-0	1003	SPIDER-MAN	1	15	10
13	978-3-16-148411-4	1003	SPIDER-MAN	7	15	13

--6. Sa se afiseze pretul celei mai scumpe editii dintre cele mai vandute editii

```
SELECT MAX(pret)
FROM benzi_desenate
WHERE isbn IN
(SELECT b.isbn
FROM rand_comanda_benzi_desenate r, benzi_desenate b
WHERE b.isbn = r.isbn
GROUP BY b.isbn);
```



The screenshot shows a SQL query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying a single row of data. The column header is 'MAX(PRET)' and the value is '18'. The status bar indicates 'All Rows Fetched: 1 in 0.008'.

MAX(PRET)
18

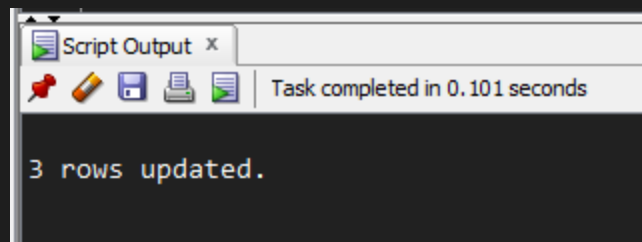
--TEST DE EVALUARE:

--1.Diminuează cu 10% stocul de benzi de desenate care au fost comandate în ultimele 3 luni de către clienți din Bucuresti

--Am modificat conditia cu adresa: nu am adrese care sa contina si orasul, asa ca am cautat dupa numele unei anumite strazi: Bisericii

--Am modificat si conditia cu lunile, pentru ca nu aveam comenzi asa recente

```
UPDATE benzi_desenate
SET cantitate = cantitate - 0.1*cantitate
WHERE isbn IN (
SELECT DISTINCT r.isbn
FROM comanda_benzi_desenate co, clienti_benzi_desenate cl, rand_comanda_benzi_desenate r
WHERE co.id_client = cl.id_client AND co.nr_comanda = r.nr_comanda AND ROUND(MONTHS_BETWEEN(s
ysdate, data)) <= 15 AND cl.adresa_client LIKE 'Bisericii%');
```






--2. Afișează clienții care au dat cel puțin 3 recenzii bune (cu note peste o limită) pentru benzile desenate care au prețul mai mare decât prețul mediu al tuturor benzilor desenate.

```
SELECT DISTINCT cl.*
FROM clienti_benzi_desenate cl, recenzii_benzi_desenate re, benzi_desenate b
WHERE cl.id_client = re.id_client AND b.isbn = re.isbn AND nota >= 2 AND b.pret <(SELECT ROUN
D(AVG(pret), 2) FROM benzi_desenate);
```

Script Output x

Query Result x



All Rows Fetched: 1 in 0.139 seconds

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	DATA_NASTERE	ADRESA_CLIENT	EMAIL_CLIENT	SEX	NUMAR_TELEFON_CLIENT	DISCOUNT	
1	107	Coriolian	Ion	20-AUG-04	Bisericii 666	coriolian@gmail.com	M	0724246710	.1

--3. Construiește o interogare care să implementeze operatorul relațional de reuniune. De asemenea, comanda va folosi o funcție de grup și una scalară.

--Sa se afiseze discountul majorat pentru fiecare client(cadou de Craciun), astfel:

--clientilor care au cheltuit intre 50 si 100 lei li se va majora discountul cu 10%

--clientilor care au cheltuit intre 100 si 160 lei li se va majora discountul cu 20%

--clientilor care au cheltuit intre 160 si 1000 lei li se va majora discountul cu 30%%

```
SELECT c.id_client, UPPER(cb.numa_client) nume_client, cb.discount,cb.discount + 0.1*cb.disco
unt discount_actualizat
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi
_desenate cb
```

```

WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client
GROUP BY c.id_client, cb.discount, cb.num_client
HAVING SUM(b.pret*r.cantitate) BETWEEN 50 AND 100
UNION
SELECT c.id_client, UPPER(cb.num_client) num_client, cb.discount, cb.discount + 0.2*cb.discount discount_actualizat
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi_desenate cb
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client
GROUP BY c.id_client, cb.discount, cb.num_client
HAVING SUM(b.pret*r.cantitate) BETWEEN 100 AND 160
UNION
SELECT c.id_client, UPPER(cb.num_client) num_client, cb.discount, cb.discount + 0.3*cb.discount discount_actualizat
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi_desenate cb
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client
GROUP BY c.id_client, cb.discount, cb.num_client
HAVING SUM(b.pret*r.cantitate) BETWEEN 160 AND 1000;

```




Query Result x				
All Rows Fetched: 8 in 0.017 seconds				
	ID_CLIENT	NUME_CLIENT	DISCOUNT	DISCOUNT_ACTUALIZAT
1	101	DORIAN	.2	0.26
2	103	POPESCU	.2	0.26
3	104	FILIPESCU	.2	0.26
4	107	CORIOLIAN	.1	0.12
5	108	CALDARE	.2	0.26
6	110	ONETIU	.05	0.055
7	116	GEORGESCU	.2	0.26
8	117	MANCU	.2	0.26

--COMENZI SUPLIMENTARE PROIECT:

--1. Sa se afiseze clientii care nu au efectuat nicio comanda pana la data de 1 iunie 2019

```
SELECT *
FROM clienti_benzi_desenate
MINUS
SELECT cl.*
FROM clienti_benzi_desenate cl, comanda_benzi_desenate co
WHERE cl.id_client = co.id_client AND co.data < TO_DATE('01.06.2019', 'DD.MM.YYYY');
```

Query Result 1 x Script Output x Query Result 1 x Query Result 2 x

 All Rows Fetched: 13 in 0.005 seconds

	ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	DATA_NASTERE	ADRESA_CLIENT	EMAIL_CLIENT	SEX	NUMAR_TELEFON_CLIENT	DISCOUNT
1	100	Dobrescu	Marius	26-MAR-00	Marinarilor 7	dobrescu@gmail.com	M	0724456710	0
2	105	Mardare	Andrei	02-JUN-98	Ionescu Brad 3	mardare@gmail.com	M	0721226710	0
3	106	Vladescu	George	05-JUL-06	Carmangeriei 2	vladescu@gmail.com	M	0724412310	0
4	107	Coriolian	Ion	20-AUG-04	Bisericii 666	coriolian@gmail.com	M	0724246710	.1
5	108	Caldare	Iulian	06-OCT-05	Uneltelor 2	caldare@gmail.com	M	0721246903	.2
6	109	Cristian	Mircea	21-DEC-05	Trandafirilor 3	cristian@gmail.com	M	0724457891	0
7	110	Onetiu	Valentin	30-NOV-83	Revolutiei 89	onetiu@gmail.com	M	0739956710	.05
8	111	Casin	Diana	11-AUG-56	Craciunului 25	casin@gmail.com	F	0724402710	0
9	112	Mircea	Andreea	12-JUN-81	Pastelui 3	mircea@gmail.com	F	0724450110	0
10	113	Zlatan	Ibrahimovic	02-MAY-89	Soldatilor 4	zlatan@gmail.com	M	0771246953	0
11	114	Lorin	Catalin	18-APR-94	Primariei 5	lorin@gmail.com	M	0720036710	0
12	116	Georgescu	Mariana	09-OCT-87	Dorobanti 1	mgeorgescu@gmail.com	F	0726675710	.2
13	117	Mancu	Matei	13-JUL-90	Razbunarii 5	mancu@gmail.com	M	0724491710	.2

--2. Sa se afiseze toti clientii care sunt de sex masculin, au comandat produse de cel putin 100 lei si au oferit recenzii

```
SELECT *
FROM clienti_benzi_desenate
WHERE sex = 'M'
INTERSECT
SELECT *
FROM clienti_benzi_desenate
WHERE id_client IN
(SELECT c.id_client
FROM rand_comanda_benzi_desenate r, benzi_desenate b, comanda_benzi_desenate c, clienti_benzi_desenate cb
WHERE r.isbn = b.isbn AND r.nr_comanda = c.nr_comanda AND cb.id_client = c.id_client
GROUP BY c.id_client
HAVING SUM(b.pret*r.cantitate) > 100)
MINUS
SELECT cl.*
FROM clienti_benzi_desenate cl, recenzii_benzi_desenate b
WHERE cl.id_client = b.id_client;
```

Query Result

Script Output

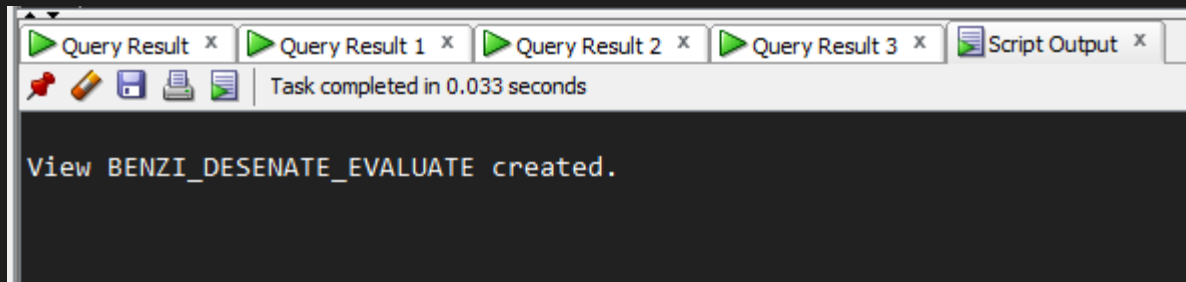
Query Result 1

Query Result 2

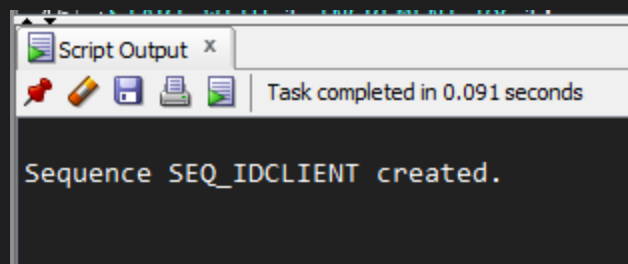
All Rows Fetched: 3 in 0.093 seconds

ID_C...	NUME_CLIENT	PRENUME_CLIENT	DATA_NASTERE	ADRESA_CLIENT	EMAIL_CLIENT	SEX	NUMAR_TELEFON_CLIENT	DISCOUNT	
1	104	Filipescu	Vlad	01-JAN-97	Infratirii 4	filipescu@gmail.com	M	0724456900	.2
2	108	Caldare	Iulian	06-OCT-05	Uneltelor 2	caldare@gmail.com	M	0721246903	.2
3	117	Mancu	Matei	13-JUL-90	Razbunarii 5	mancu@gmail.com	M	0724491710	.2

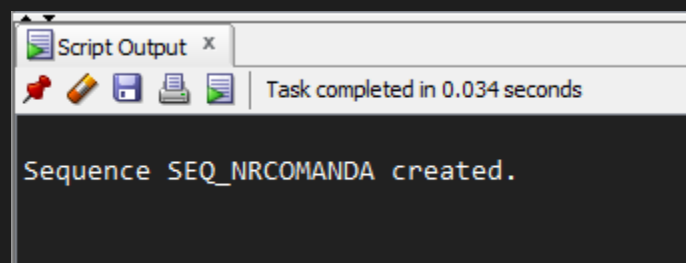
--3. Sa se realizeze o tabela virtuala cu toate benzile desenate care au primit review
CREATE OR REPLACE VIEW benzi_desenate_evaluate AS
SELECT b.*, r.nota
FROM benzi_desenate b, recenzii_benzi_desenate r
WHERE b.isbn = r.isbn



--4. Sa se creeze o secventa pentru asigurarea unicitatii cheii primare din tabela clienti_benzi_desenate
CREATE SEQUENCE seq_idclient
START WITH 100 INCREMENT BY 1
MAXVALUE 100000 NOCYCLE;



--5. Sa se creeze o scventa pentru asigurarea unicitatii cheii primare din tabela comanda_benzi_desenate
CREATE SEQUENCE seq_nrcomanda
START WITH 1 INCREMENT BY 1;

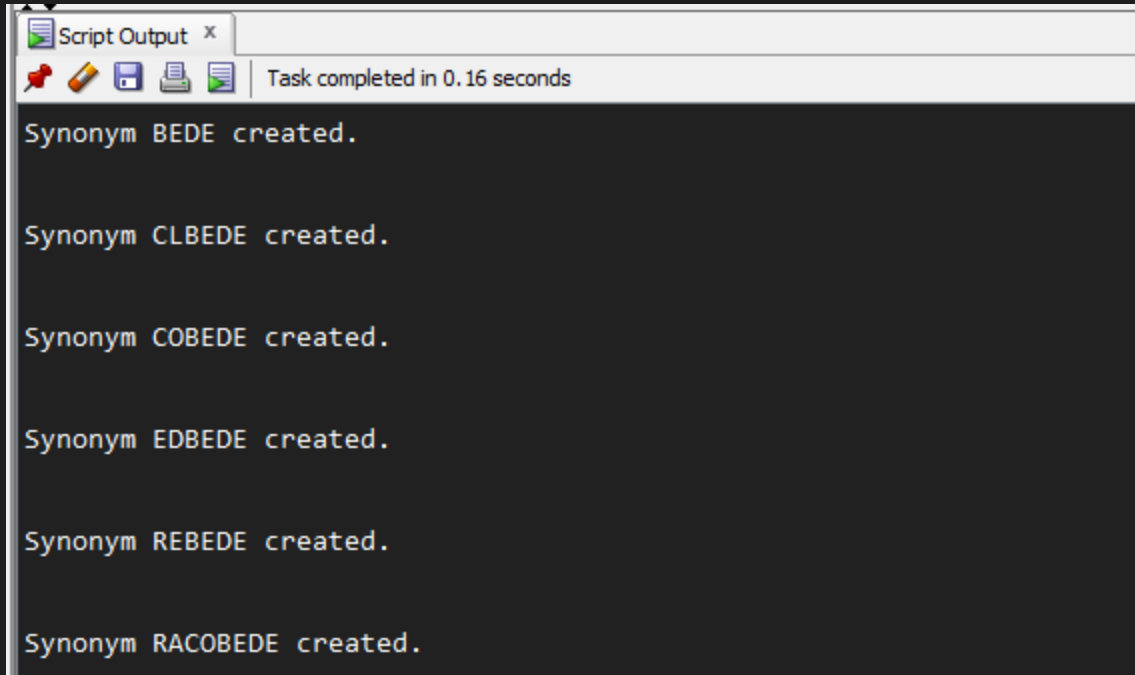


--6. Sa se creeze sinonime pentru toate tebelele, pastrandu-se doar primele 2 litere din fiecare cuvant component

```
CREATE SYNONYM bede for benzi_desenate;  
CREATE SYNONYM clbede for clienti_benzi_desenate;  
CREATE SYNONYM cobede for comanda_benzi_desenate;  
CREATE SYNONYM edbede for edituri_benzi_desenate;
```



```
CREATE SYNONYM rebede for recenzii_benzi_desenate;  
CREATE SYNONYM racobede for rand_comanda_benzi_desenate;
```



```
Script Output x  
Task completed in 0.16 seconds  
Synonym BEDE created.  
  
Synonym CLBEDE created.  
  
Synonym COBEDE created.  
  
Synonym EDBEDE created.  
  
Synonym REBEDE created.  
  
Synonym RACOBED created.
```

--7. Sa se afiseze pretul crescut al fiecarui produs in functie de editura, astfel:

--ARTHUR: + 1 leu

--KOOB: + 2 lei

--CRISAN: + 0.5 lei

--DC_COMICS: + 3 lei

--MARVEL: + 3.5 lei

```
SELECT DECODE(e.ume, 'ARTHUR', 1, 'KOOB', 2, 'CRISAN', 0.5, 'DC_COMICS', 3, 'MARVEL', 3.5) +  
       b.pret pret_actualizat  
FROM bede b, edbede e  
WHERE b.id_editura = e.id_editura;
```

--8. Sa se adauge inregistrari suplimentare in tabela recenzii_benzi_desenate

```
CREATE SEQUENCE seq_recenzii
```

```
START WITH 6 INCREMENT BY 1
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '107', '978-3-16-148410-0', '5');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '100', '978-3-16-148410-0', '4');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '101', '978-3-16-148411-9', '4');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '102', '978-3-16-148411-8', '2');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '103', '978-3-16-148411-7', '5');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '104', '978-3-16-148411-6', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '105', '978-3-16-148411-5', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '106', '978-3-16-148411-4', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '107', '978-3-16-148411-3', '5');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '108', '978-3-16-148411-2', '5');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '109', '978-3-16-148411-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '110', '978-3-16-148411-0', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '111', '978-3-16-148410-9', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '112', '978-3-16-148410-8', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '113', '978-3-16-148410-7', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '114', '978-3-16-148410-6', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '115', '978-3-16-148410-5', '5');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '116', '978-3-16-148410-4', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '117', '978-3-16-148410-3', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '100', '978-3-16-148410-2', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '101', '978-3-16-148410-1', '4');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '102', '978-3-16-018410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '103', '978-3-16-028410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '104', '978-3-16-038410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '105', '978-3-16-048410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '106', '978-3-16-058410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '107', '978-3-16-068410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '108', '978-3-16-078410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '109', '978-3-16-088410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '110', '978-3-16-098410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '111', '978-3-16-108410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '112', '978-3-16-101410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '113', '978-3-16-102410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '114', '978-3-16-103410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '115', '978-3-16-112410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '116', '978-3-16-101423-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '117', '978-3-16-103490-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
```

```
VALUES (seq_recenzii.NEXTVAL, '100', '901-3-16-101410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '101', '902-3-16-101410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '102', '903-3-16-101410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '103', '904-3-16-101410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '104', '905-3-16-101410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '105', '906-3-16-101410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '106', '978-3-16-148410-0', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '107', '978-3-16-148410-0', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '108', '978-3-16-148411-9', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '109', '978-3-16-148411-8', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '110', '978-3-16-148411-7', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '111', '978-3-16-148411-6', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '112', '978-3-16-148411-5', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '113', '978-3-16-148411-4', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '114', '978-3-16-148411-3', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '115', '978-3-16-148411-2', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '116', '978-3-16-148411-1', '4');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '117', '978-3-16-148411-0', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '100', '978-3-16-148410-9', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '101', '978-3-16-148410-8', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '102', '978-3-16-148410-7', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '103', '978-3-16-148410-6', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '104', '978-3-16-148410-5', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '105', '978-3-16-148410-4', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '106', '978-3-16-148410-3', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '107', '978-3-16-148410-2', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '108', '978-3-16-148410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '109', '978-3-16-018410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '110', '978-3-16-028410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '111', '978-3-16-038410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '112', '978-3-16-048410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '113', '978-3-16-058410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '114', '978-3-16-068410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '115', '978-3-16-078410-1', '3');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '116', '978-3-16-088410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '117', '978-3-16-098410-1', '5');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '100', '978-3-16-108410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '101', '978-3-16-101410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '102', '978-3-16-102410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '103', '978-3-16-103410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '104', '978-3-16-112410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '105', '978-3-16-101423-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '106', '978-3-16-103490-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '107', '901-3-16-101410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '108', '902-3-16-101410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '109', '903-3-16-101410-1', '1');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '110', '904-3-16-101410-1', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '111', '905-3-16-101410-1', '3');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '112', '906-3-16-101410-1', '4');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
VALUES (seq_recenzii.NEXTVAL, '113', '978-3-16-148410-0', '2');

INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)
```

```
VALUES (seq_recenzii.NEXTVAL, '114', '978-3-16-148410-0', '3');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '115', '978-3-16-148411-9', '1');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '116', '978-3-16-148411-8', '3');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '117', '978-3-16-148411-7', '2');
```

```
INSERT INTO rebede(ID_RECENZIE, ID_CLIENT, ISBN, NOTA)  
VALUES (seq_recenzii.NEXTVAL, '100', '978-3-16-148411-6', '1');
```

	ID_RECENZIE	ID_CLIENT	ISBN	NOTA	TEXT
1	1	107	978-3-16-148410-0	5	foarte fain
2	2	107	978-3-16-112410-1	4	oarecum fain
3	3	107	978-3-16-108410-1	3	merge
4	4	101	978-3-16-108410-1	3	(null)
5	5	101	978-3-16-148411-5	1	nu prea mi-a placut, baieti
6	6	115	978-3-16-102410-1	5	decent
7	7	107	978-3-16-148410-0	5	(null)
8	8	100	978-3-16-148410-0	4	(null)
9	9	101	978-3-16-148411-9	4	(null)
10	10	102	978-3-16-148411-8	2	(null)
11	11	103	978-3-16-148411-7	5	(null)
12	12	104	978-3-16-148411-6	3	(null)
13	13	105	978-3-16-148411-5	1	(null)
14	14	106	978-3-16-148411-4	4	(null)
15	15	107	978-3-16-148411-3	5	(null)
16	16	108	978-3-16-148411-2	5	(null)
17	17	109	978-3-16-148411-1	3	(null)
18	18	110	978-3-16-148411-0	2	(null)
19	19	111	978-3-16-148410-9	1	(null)
20	20	112	978-3-16-148410-8	4	(null)
21	21	113	978-3-16-148410-7	1	(null)
22	22	114	978-3-16-148410-6	2	(null)
23	23	115	978-3-16-148410-5	5	(null)
24	24	116	978-3-16-148410-4	3	(null)
25	25	117	978-3-16-148410-3	2	(null)
26	26	100	978-3-16-148410-2	1	(null)
27	27	101	978-3-16-148410-1	4	(null)
28	28	102	978-3-16-018410-1	4	(null)
29	29	103	978-3-16-028410-1	1	(null)
30	30	104	978-3-16-038410-1	3	(null)
31	31	105	978-3-16-048410-1	2	(null)

--9. Sa se afiseze nota medie la care au fost evaluate alaturi de editura de la care provin

```

SELECT b.isbn, e.ume, ROUND(SUM(r.nota)/COUNT(r.nota), 2) nota_medie
FROM rebede r, bede b, edbede e
WHERE r.isbn = b.isbn AND b.id_editura = e.id_editura
GROUP BY b.isbn, e.ume
ORDER BY SUM(r.nota)/COUNT(r.nota);

```

Query Result x			
All Rows Fetched: 42 in 0.008 seconds			
	ISBN	NUME	NOTA_MEDIE
1	978-3-16-148410-7	DC_COMICS	1
2	978-3-16-148410-9	DC_COMICS	1
3	978-3-16-028410-1	MARVEL	1
4	978-3-16-148411-5	DC_COMICS	1.33
5	904-3-16-101410-1	ARTHUR	1.5
6	978-3-16-148410-3	DC_COMICS	1.5
7	978-3-16-148411-6	DC_COMICS	1.67
8	978-3-16-148410-6	DC_COMICS	2
9	978-3-16-148410-2	DC_COMICS	2
10	978-3-16-148411-0	DC_COMICS	2
11	978-3-16-148411-9	DC_COMICS	2.33
12	978-3-16-102410-1	CRISAN	2.33
13	978-3-16-068410-1	MARVEL	2.5
14	978-3-16-103410-1	CRISAN	2.5
15	978-3-16-078410-1	MARVEL	2.5
16	978-3-16-101423-1	KOOB	2.5
17	978-3-16-058410-1	MARVEL	2.5
18	978-3-16-038410-1	MARVEL	2.5
19	903-3-16-101410-1	ARTHUR	2.5
20	978-3-16-048410-1	MARVEL	2.5
21	978-3-16-148411-8	DC_COMICS	2.67
22	901-3-16-101410-1	ARTHUR	3
23	978-3-16-108410-1	MARVEL	3
24	978-3-16-098410-1	MARVEL	3
25	905-3-16-101410-1	ARTHUR	3
26	978-3-16-103490-1	KOOB	3
27	978-3-16-101410-1	CRISAN	3
28	978-3-16-148410-0	DC_COMICS	3.14
29	978-3-16-148410-1	DC_COMICS	3.5
30	978-3-16-148411-1	DC_COMICS	3.5
31	978-3-16-088410-1	MARVEL	3.5
32	902-3-16-101410-1	ARTHUR	3.5
33	978-3-16-148410-4	DC_COMICS	3.5

--10. Sa se afiseze clientii si numarul de review-uri pe care le-au oferit

```
SELECT c.id_client, c.num_client, COUNT(r.nota) numar_note
```

```
FROM clbede c, rebede r
```

```
WHERE c.id_client = r.id_client
```

```
GROUP BY c.id_client, c.num_client
```

Query Result 4 x			
All Rows Fetched: 18 in 0.009 seconds			
	ID_CLIENT	NUME_CLIENT	NUMAR_NOTE
1	100	Dobrescu	6
2	101	Dorian	7
3	102	Matei	5
4	103	Popescu	5
5	104	Filipescu	5
6	105	Mardare	5
7	106	Vladescu	5
8	107	Coriolian	9
9	108	Caldare	5
10	109	Cristian	5
11	110	Onetiu	5
12	111	Casin	5
13	112	Mircea	5
14	113	Zlatan	5
15	114	Lorin	5
16	115	Dobrescu	6
17	116	Georgescu	5
18	117	Mancu	5

SEMESTRUL 2

--TEMA1

--Utilizând schema BD de la proiect (cel din semestrul 1, care va fi continuat în acest semestru), construiți cel puțin 5 blocuri PL/SQL care să conțină:

--

--structuri de control

--cursori implicați

--cursori expliți

--

Notă: se punctează complexitatea blocurilor create; acestea ar trebui să includă simultan cât mai multe dintre mecanismele învățate: structuri de control, cursori...

--1. Sa se afiseze editiile (cantitate + detalii) care pot fi returnate editurii. CONDITII: sa nu se fi vandut niciun exemplar, stocul sa fie mai mare de 10 unitati

--cursori expliți

SET SERVEROUTPUT ON

DECLARE

CURSOR c

IS

SELECT *

FROM benzi_desenate

WHERE isbn NOT IN

(SELECT b.isbn

FROM rand_comanda_benzi_desenate r, benzi_desenate b

WHERE b.isbn = r.isbn

GROUP BY b.isbn)

AND cantitate > 10;

BEGIN

FOR var in c LOOP

DBMS_OUTPUT.PUT_LINE(var.isbn || ' Serie: ' || RPAD(var.serie, 11) || ' Episod: ' || RPAD(var.nr_episod, 2) || ' Cantitate: ' || var.cantitate);

END LOOP;

END;

```
Script Output x
Task completed in 0.036 seconds

978-3-16-148410-7 Serie: SPIDER-MAN Episod: 14 Cantitate: 21
978-3-16-148411-4 Serie: SPIDER-MAN Episod: 7 Cantitate: 13
978-3-16-018410-1 Serie: IRON-MAN Episod: 1 Cantitate: 21
978-3-16-148410-5 Serie: SPIDER-MAN Episod: 16 Cantitate: 40
978-3-16-148410-1 Serie: SPIDER-MAN Episod: 20 Cantitate: 21
978-3-16-058410-1 Serie: IRON-MAN Episod: 5 Cantitate: 16
905-3-16-101410-1 Serie: MACHINE-MAN Episod: 5 Cantitate: 20
978-3-16-148410-2 Serie: SPIDER-MAN Episod: 19 Cantitate: 30
978-3-16-088410-1 Serie: IRON-MAN Episod: 8 Cantitate: 20
978-3-16-148410-8 Serie: SPIDER-MAN Episod: 13 Cantitate: 11
978-3-16-028410-1 Serie: IRON-MAN Episod: 2 Cantitate: 20
901-3-16-101410-1 Serie: MACHINE-MAN Episod: 1 Cantitate: 11
978-3-16-148410-3 Serie: SPIDER-MAN Episod: 18 Cantitate: 45

PL/SQL procedure successfully completed.
```

--2. Sa se afiseze toate informatiile despre editiile care au pretul mai mic decat pretul mediu

--cursori expliciti

DECLARE

CURSOR c IS

SELECT *

FROM benzi_desenate

WHERE pret < (SELECT ROUND(AVG(pret), 2) FROM benzi_desenate)

ORDER BY pret;

pret_mediu benzi_desenate.pret%type;

var c%rowtype;

BEGIN

SELECT ROUND(AVG(pret), 2) INTO pret_mediu FROM benzi_desenate;

DBMS_OUTPUT.PUT_LINE('Pret mediu: ' || pret_mediu);

OPEN c;

LOOP

FETCH c INTO var;

EXIT WHEN c%notfound;

DBMS_OUTPUT.PUT_LINE(var.isbn || ' Serie: ' || RPAD(var.serie, 11) || ' Episod: ' || RPAD(var.nr_episod, 2) || ' Cantitate: ' || RPAD(var.cantitate, 2) || ' Pret: ' || var.pret);

END LOOP;

END;

```
Script Output x
Task completed in 0.02 seconds

Pret mediu: 16
978-3-16-101423-1 Serie: HILLBILLIES Episod: 65 Cantitate: 1 Pret: 9
978-3-16-112410-1 Serie: HILLBILLIES Episod: 67 Cantitate: 1 Pret: 10
978-3-16-103490-1 Serie: HILLBILLIES Episod: 23 Cantitate: 10 Pret: 10
978-3-16-148411-7 Serie: SPIDER-MAN Episod: 4 Cantitate: 6 Pret: 15
978-3-16-148411-6 Serie: SPIDER-MAN Episod: 5 Cantitate: 3 Pret: 15
978-3-16-148411-5 Serie: SPIDER-MAN Episod: 6 Cantitate: 16 Pret: 15
978-3-16-148411-3 Serie: SPIDER-MAN Episod: 8 Cantitate: 1 Pret: 15
978-3-16-148411-2 Serie: SPIDER-MAN Episod: 9 Cantitate: 29 Pret: 15
978-3-16-148411-1 Serie: SPIDER-MAN Episod: 10 Cantitate: 7 Pret: 15
978-3-16-148411-8 Serie: SPIDER-MAN Episod: 3 Cantitate: 8 Pret: 15
978-3-16-148411-9 Serie: SPIDER-MAN Episod: 2 Cantitate: 9 Pret: 15
978-3-16-148410-0 Serie: SPIDER-MAN Episod: 1 Cantitate: 5 Pret: 15
978-3-16-148411-4 Serie: SPIDER-MAN Episod: 7 Cantitate: 13 Pret: 15

PL/SQL procedure successfully completed.
```

--3. Sa se mareasca discountul fiecarui client nascut in luna x cu 20%
--cursori impliciti

BEGIN

```
UPDATE clienti_benzi_desenate
SET discount = 1.2 * discount
WHERE EXTRACT(MONTH FROM data_nastere) = 3;
```

```
IF (SQL%NOTFOUND) THEN
DBMS_OUTPUT.PUT_LINE('Nu exista clienti nascuti in luna respectiva');
ELSE DBMS_OUTPUT.PUT_LINE('A fost actualizat discount-
ul pentru ' || SQL%ROWCOUNT || ' clienti ');
END IF;
```

END;

```
Script Output x
Task completed in 0.017 seconds

A fost actualizat discount-ul pentru 1 clienti

PL/SQL procedure successfully completed.
```

--4.Diminuează cu 10% stocul de benzi de desenate care au fost comandate în ultimele 15 luni
de către clienți din Bisericii
--cursori impliciti

BEGIN

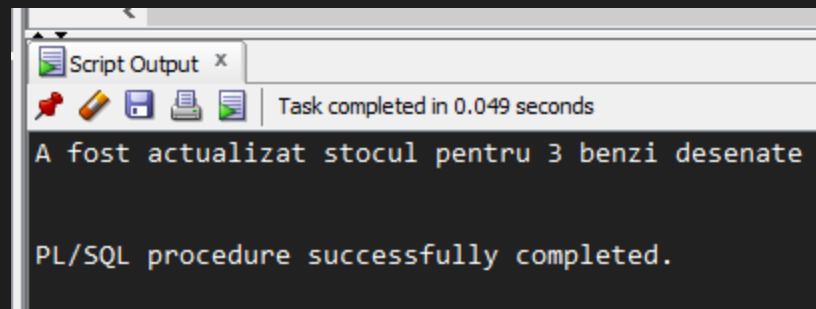
```

UPDATE benzi_desenate
SET cantitate = cantitate - 0.1*cantitate
WHERE isbn IN (
SELECT DISTINCT r.isbn
FROM comanda_benzi_desenate co, clienti_benzi_desenate cl, rand_comanda_benzi_desenate r
WHERE co.id_client = cl.id_client AND co.nr_comanda = r.nr_comanda AND ROUND(MONTHS_BETWEEN(s
ysdate, data)) <= 15 AND cl.adresa_client LIKE 'Bisericii%');

IF (SQL%NOTFOUND) THEN
DBMS_OUTPUT.PUT_LINE('Nu exista astfel de comenzi');
ELSE DBMS_OUTPUT.PUT_LINE('A fost actualizat stocul pentru ' || SQL%ROWCOUNT || ' benzi desen
ate');
END IF;

END;

```



--5. Sa se afiseze venitul real rezultat in urma tuturor comenzilor si sa se estimeze venitul pentru urmatorul trimestru

```

DECLARE

```

```

v_venit number;

```

```

BEGIN

```

```

SELECT SUM(r.cantitate * b.pret) - (SUM(r.cantitate * b.pret) -
0.1 * SUM(r.cantitate * b.pret) - 0.5 * (SUM(r.cantitate * b.pret) -
0.1 * SUM(r.cantitate * b.pret))) venit_real
INTO v_venit
FROM rand_comanda_benzi_desenate r, benzi_desenate b, edituri_benzi_desenate e
WHERE b.isbn = r.isbn AND b.id_editura = e.id_editura
ORDER BY b.id_editura ASC;

```

```

CASE

```

```

WHEN v_venit > 1000 THEN v_venit := 1.2*v_venit;
WHEN v_venit < 1000 THEN v_venit := 1.1*v_venit;
WHEN v_venit = 1000 THEN v_venit := 1.15*v_venit;
END CASE;

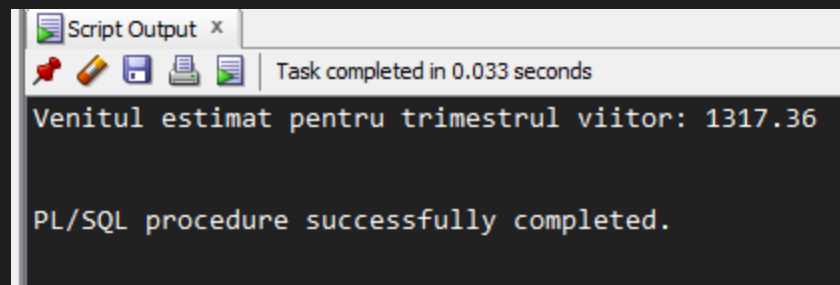
```

```

DBMS_OUTPUT.PUT_LINE('Venitul estimat pentru trimestrul viitor: ' || v_venit);

```

END;



The screenshot shows a 'Script Output' window with a title bar containing a close button and a status bar indicating 'Task completed in 0.033 seconds'. The main text area displays the output of a PL/SQL script: 'Venitul estimat pentru trimestrul viitor: 1317.36' followed by 'PL/SQL procedure successfully completed.' on a new line.

--TEMA2

--Utilizând schema BD de la proiect (cel din semestrul 1, care va fi continuat în acest semestru), construiți cel puțin 4 blocuri PL/SQL care să conțină:

--

--excepții implicite (de sistem)

--excepții explicite (definite de utilizator)

--Nota: se punctează complexitatea blocurilor create; acestea ar trebui să includă simultan cât mai multe dintre mecanismele învățate: structuri de control, cursori, excepții...

--1. Creați un bloc PL/SQL pentru a selecta codul și data de încheiere a comenzilor încheiate într-un an introdus de la tastatură

--

--dacă interogarea returnează mai mult de o valoare pentru codul comenzii, tratați excepția -
--cu o rutină de tratare corespunzătoare și afișați mesajul Atentie! În anul yyyy s-au încheiat mai multe comenzi!;

--

dacă interogarea nu returnează nicio valoare pentru codul comenzii, tratați excepția cu o rutină de tratare corespunzătoare și afișați mesajul Atentie! În anul yyyy nu s-au încheiat comenzi!;

--dacă se returnează o singură linie, afișați codul și data comenzii;

--

tratați orice altă excepție cu o rutină de tratare corespunzătoare și afișați mesajul A apărut o altă excepție!.

SET SERVEROUTPUT ON

DECLARE

 v_an NUMBER := &an;

 v_luna NUMBER := &luna;

 v_cod comanda_benzi_desenate.nr_comanda%TYPE;

 v_data comanda_benzi_desenate.data%TYPE;

BEGIN

 SELECT nr_comanda, data

 INTO v_cod, v_data

 FROM comanda_benzi_desenate

 WHERE EXTRACT(YEAR FROM data) = v_an AND EXTRACT(MONTH FROM data) = v_luna;

 DBMS_OUTPUT.PUT_LINE(v_cod || ' ' || v_data);

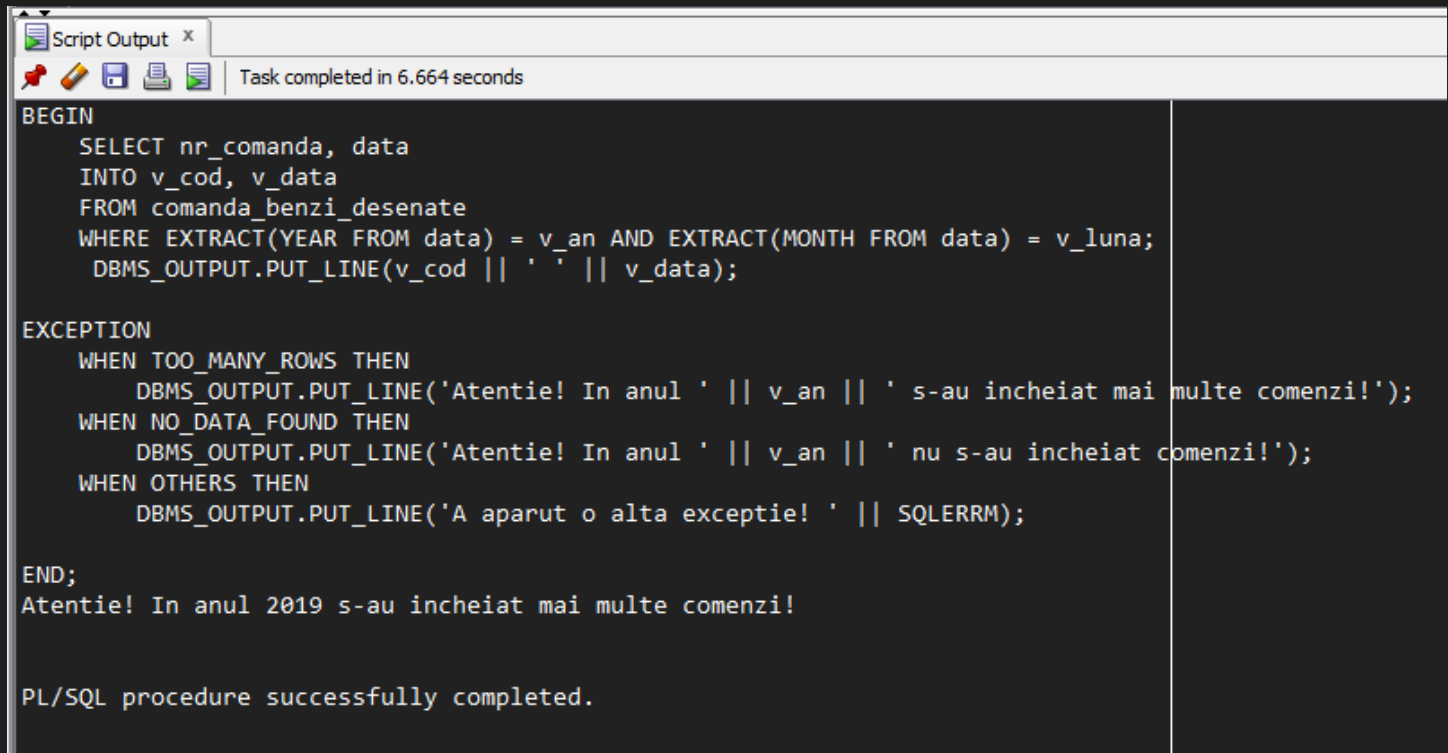
EXCEPTION

```

    WHEN TOO_MANY_ROWS THEN
        DBMS_OUTPUT.PUT_LINE('Atentie! In anul ' || v_an || ' s-
au incheiat mai multe comenzi!');
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Atentie! In anul ' || v_an || ' nu s-au incheiat comenzi!');
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('A aparut o alta exceptie! ' || SQLERRM);

END;
/

```



The screenshot shows a 'Script Output' window with a toolbar and a status bar indicating 'Task completed in 6.664 seconds'. The main area displays the execution of a PL/SQL script. The script begins with a SELECT statement to retrieve 'nr_comanda' and 'data' from 'comanda_benzi_desenate' for a specific year and month. It then uses DBMS_OUTPUT.PUT_LINE to display the results. An EXCEPTION block handles three cases: TOO_MANY_ROWS, NO_DATA_FOUND, and OTHERS, each with a corresponding message. The script ends with END; and the output shows the message 'Atentie! In anul 2019 s-au incheiat mai multe comenzi!'. Finally, a message 'PL/SQL procedure successfully completed.' is displayed.

```

BEGIN
    SELECT nr_comanda, data
    INTO v_cod, v_data
    FROM comanda_benzi_desenate
    WHERE EXTRACT(YEAR FROM data) = v_an AND EXTRACT(MONTH FROM data) = v_luna;
    DBMS_OUTPUT.PUT_LINE(v_cod || ' ' || v_data);

EXCEPTION
    WHEN TOO_MANY_ROWS THEN
        DBMS_OUTPUT.PUT_LINE('Atentie! In anul ' || v_an || ' s-au incheiat mai multe comenzi!');
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Atentie! In anul ' || v_an || ' nu s-au incheiat comenzi!');
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('A aparut o alta exceptie! ' || SQLERRM);

END;
Atentie! In anul 2019 s-au incheiat mai multe comenzi!

PL/SQL procedure successfully completed.

```

--
2. Creati un bloc PL/SQL prin care se dubleaza pretul benzii desenate al carei isbn este citit de la tastatura.
--

In cazul in care acesta nu exista (comanda UPDATE nu realizeaza nicio modificare) se va invoca o exceptie. Tratatati exceptia prin afisarea unui mesaj.

--

testati cu id produs 3061 pentru a obtine modificari prin UPDATE => se produce modificarea si se face afisarea din zona executabila

testati cu id produs 100 pentru a nu obtine modificari prin UPDATE => se invoca exceptia definita de utilizator => se afiseaza mesajul din Exception

```

DECLARE
    v_isbn benzi_desenate.isbn%type := &isbn;
    nu_exista_benzi_desenate EXCEPTION;

```

```

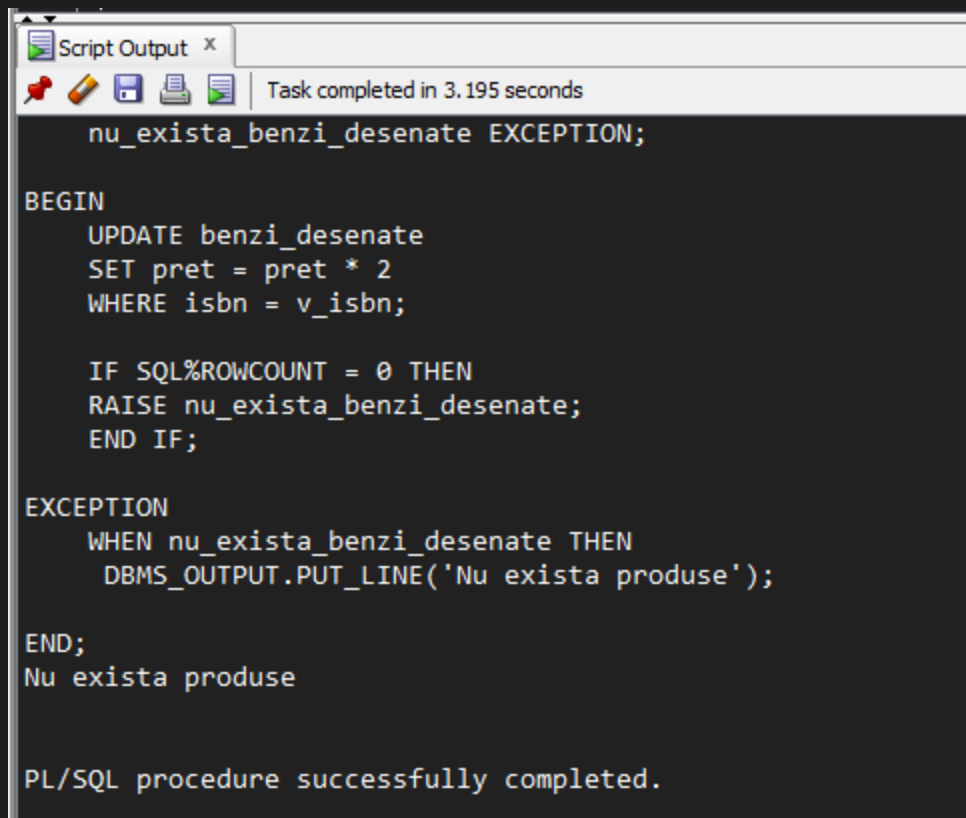
BEGIN
    UPDATE benzi_desenate
    SET pret = pret * 2
    WHERE isbn = v_isbn;

    IF SQL%ROWCOUNT = 0 THEN
        RAISE nu_exista_benzi_desenate;
    END IF;

EXCEPTION
    WHEN nu_exista_benzi_desenate THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista produse');

END;
/

```



The screenshot shows a 'Script Output' window with a title bar. Below the title bar, there is a status bar indicating 'Task completed in 3.195 seconds'. The main area of the window displays the following text:

```

nu_exista_benzi_desenate EXCEPTION;

BEGIN
    UPDATE benzi_desenate
    SET pret = pret * 2
    WHERE isbn = v_isbn;

    IF SQL%ROWCOUNT = 0 THEN
        RAISE nu_exista_benzi_desenate;
    END IF;

EXCEPTION
    WHEN nu_exista_benzi_desenate THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista produse');

END;
Nu exista produse

PL/SQL procedure successfully completed.

```

--3. Sa se mareasca discountul fiecarui client nascut in luna x cu 20%

```

DECLARE
    nu_exista EXCEPTION;

BEGIN

    UPDATE clienti_benzi_desenate
    SET discount = 1.2 * discount
    WHERE EXTRACT(MONTH FROM data_nastere) = 100;

    IF (SQL%NOTFOUND) THEN

```



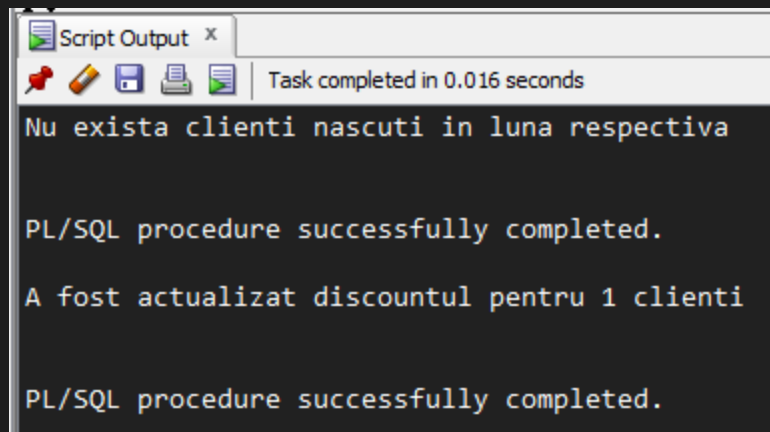
```

        RAISE nu_exista;
    ELSE DBMS_OUTPUT.PUT_LINE('A fost actualizat discountul pentru ' || SQL%ROWCOUNT || ' cli
enti ');
    END IF;

EXCEPTION

    WHEN nu_exista THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista clienti nascuti in luna respectiva');
END;

```



```

Script Output x
Task completed in 0.016 seconds

Nu exista clienti nascuti in luna respectiva

PL/SQL procedure successfully completed.

A fost actualizat discountul pentru 1 clienti

PL/SQL procedure successfully completed.

```

--

4.Diminuează cu 10% stocul de benzi de desenate care au fost comandate în ultimele 15 luni de către clienți din Bisericii

```

DECLARE
    nu_exista EXCEPTION;

BEGIN

    UPDATE benzi_desenate
    SET cantitate = cantitate - 0.1*cantitate
    WHERE isbn IN (
        SELECT DISTINCT r.isbn
        FROM comanda_benzi_desenate co, clienti_benzi_desenate cl, rand_comanda_benzi_desenate r
        WHERE co.id_client = cl.id_client AND co.nr_comanda = r.nr_comanda AND ROUND(MONTHS_BETWE
EN(sysdate, data)) <= 15 AND cl.adresa_client LIKE 'Bisericii%');

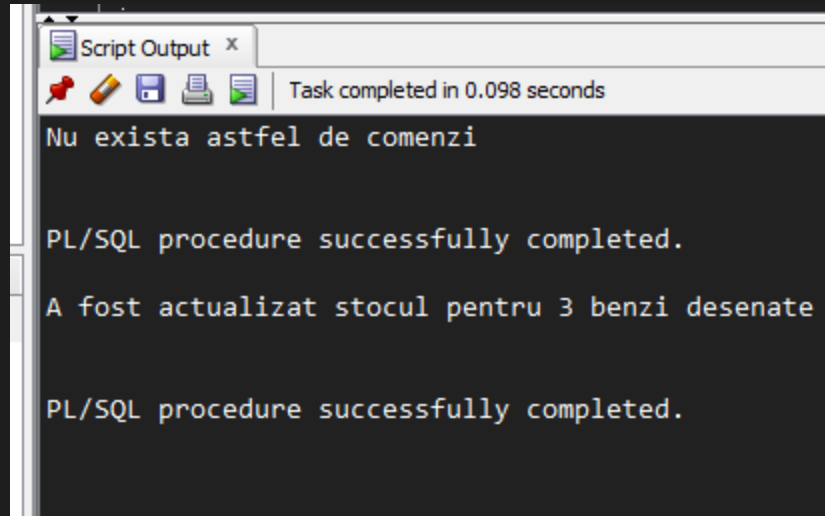
    IF (SQL%NOTFOUND) THEN
        RAISE nu_exista;
    ELSE DBMS_OUTPUT.PUT_LINE('A fost actualizat stocul pentru ' || SQL%ROWCOUNT || ' benzi d
esenate');
    END IF;

EXCEPTION

    WHEN nu_exista THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista astfel de comenzi');

```

END;



--TEMA3

--

Utilizând schema BD de la proiect (cel din semestrul 1, care va fi continuat în acest semestru), construiți cel puțin:

--

--3 proceduri

--4 funcții

--(+ apelurile aferente)

--proceduri

--1. afisare top N editii dupa nota primita

SET SERVEROUTPUT ON;

```
CREATE OR REPLACE PROCEDURE top_benzi (p_n NUMBER)
IS
```

```
    CURSOR c_benzi
    IS
    SELECT isbn, calcul_nota_medie(isbn) as nota
    FROM benzi_desenate
    ORDER BY calcul_nota_medie(isbn) DESC
    FETCH FIRST p_n ROWS ONLY;
```

```
    v_numar_produce NUMBER;
```

```
    numar_aberant_produce EXCEPTION;
```

```
BEGIN
```

```
    SELECT COUNT(isbn)
```

```

    INTO v_numar_produce
  FROM benzi_desenate;

  IF (p_n > v_numar_produce) THEN
    RAISE numar_aberant_produce;
  END IF;

  FOR var IN c_benzi LOOP
    DBMS_OUTPUT.PUT_LINE(afisare(var.isbn) || ' ' || var.nota);
  END LOOP;

EXCEPTION

  WHEN numar_aberant_produce THEN
    DBMS_OUTPUT.PUT_LINE('Numarul de produse cerut este mult mai mare decat numarul total
    de produse aflate in sistem!');

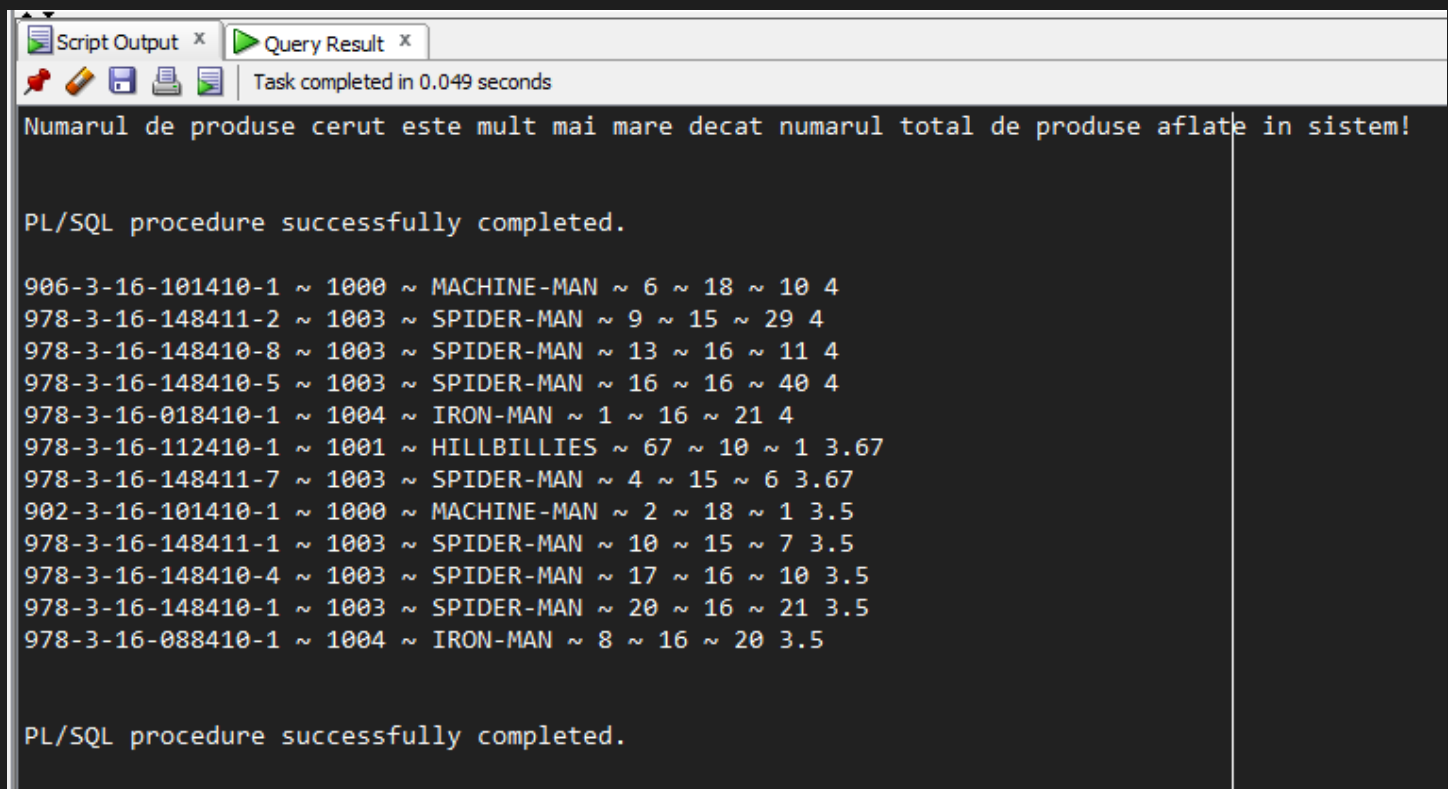
END;

--apel

EXECUTE top_benzi(19009999);

EXECUTE top_benzi(12);

```



The screenshot shows a SQL script execution window with two tabs: "Script Output" and "Query Result". The "Script Output" tab is active, displaying the results of the script execution. The output includes a message indicating that the number of requested products is greater than the total number of products in the system, followed by a list of products and their details, and a confirmation that the PL/SQL procedure completed successfully.

```

Task completed in 0.049 seconds

Numarul de produse cerut este mult mai mare decat numarul total de produse aflate in sistem!

PL/SQL procedure successfully completed.

906-3-16-101410-1 ~ 1000 ~ MACHINE-MAN ~ 6 ~ 18 ~ 10 4
978-3-16-148411-2 ~ 1003 ~ SPIDER-MAN ~ 9 ~ 15 ~ 29 4
978-3-16-148410-8 ~ 1003 ~ SPIDER-MAN ~ 13 ~ 16 ~ 11 4
978-3-16-148410-5 ~ 1003 ~ SPIDER-MAN ~ 16 ~ 16 ~ 40 4
978-3-16-018410-1 ~ 1004 ~ IRON-MAN ~ 1 ~ 16 ~ 21 4
978-3-16-112410-1 ~ 1001 ~ HILLBILLIES ~ 67 ~ 10 ~ 1 3.67
978-3-16-148411-7 ~ 1003 ~ SPIDER-MAN ~ 4 ~ 15 ~ 6 3.67
902-3-16-101410-1 ~ 1000 ~ MACHINE-MAN ~ 2 ~ 18 ~ 1 3.5
978-3-16-148411-1 ~ 1003 ~ SPIDER-MAN ~ 10 ~ 15 ~ 7 3.5
978-3-16-148410-4 ~ 1003 ~ SPIDER-MAN ~ 17 ~ 16 ~ 10 3.5
978-3-16-148410-1 ~ 1003 ~ SPIDER-MAN ~ 20 ~ 16 ~ 21 3.5
978-3-16-088410-1 ~ 1004 ~ IRON-MAN ~ 8 ~ 16 ~ 20 3.5

PL/SQL procedure successfully completed.

```

--2. afisare top N clienti dupa valoarea comenzii efectuate

```
CREATE OR REPLACE PROCEDURE top_clienti (p_n NUMBER)
```

IS

```
CURSOR c_comenzi
IS
SELECT id_client, valoare_comanda(nr_comanda) as valoare
FROM comanda_benzi_desenate
ORDER BY valoare_comanda(nr_comanda) DESC
FETCH FIRST p_n ROWS ONLY;
```

```
v_numar_clienti NUMBER;
```

```
numar_aberant_clienti EXCEPTION;
```

BEGIN

```
SELECT COUNT(id_client)
INTO v_numar_clienti
FROM clienti_benzi_desenate;
```

```
IF (p_n > v_numar_clienti) THEN
    RAISE numar_aberant_clienti;
END IF;
```

```
FOR var IN c_comenzi LOOP
    DBMS_OUTPUT.PUT_LINE(afisare_client(var.id_client) || ' ' || var.valoare);
END LOOP;
```

EXCEPTION

```
WHEN numar_aberant_clienti THEN
    DBMS_OUTPUT.PUT_LINE('Numarul de clienti cerut este mult mai mare decat numarul total
de clienti aflati in sistem!');
```

END;

--apel

```
EXECUTE top_clienti(10);
```

```
EXECUTE top_clienti(500);
```

```
Script Output x Query Result x
Task completed in 0.036 seconds

117 ~ Matei ~ Mancu ~ mancu@gmail.com ~ 0724491710 ~ 13-JUL-90 ~ M ~ .2 ~ Razbunarii 5 384
103 ~ Ioana ~ Popescu ~ popescu@gmail.com ~ 0784456710 ~ 16-MAY-03 ~ F ~ .2 ~ Zambilelor 5 340
116 ~ Mariana ~ Georgescu ~ mgeorgescu@gmail.com ~ 0726675710 ~ 09-OCT-87 ~ F ~ .2 ~ Dorobanti 1 224
104 ~ Vlad ~ Filipescu ~ filipescu@gmail.com ~ 0724456900 ~ 01-JAN-97 ~ M ~ .2 ~ Infratirii 4 192
108 ~ Iulian ~ Caldare ~ caldare@gmail.com ~ 0721246903 ~ 06-OCT-05 ~ M ~ .2 ~ Uneltelor 2 171
107 ~ Ion ~ Coriolian ~ coriolian@gmail.com ~ 0724246710 ~ 20-AUG-04 ~ M ~ .1 ~ Bisericii 666 155
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 150
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 124
110 ~ Valentin ~ Onetiu ~ onetiu@gmail.com ~ 0739956710 ~ 30-NOV-83 ~ M ~ .05 ~ Revolutiei 89 90
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 54

PL/SQL procedure successfully completed.

Numarul de clienti cerut este mult mai mare decat numarul total de clienti aflati in sistem!

PL/SQL procedure successfully completed.
```

--3. afisare clienti care au efectuat comenzi cu o valoare mai mare de o suma data

```
CREATE OR REPLACE PROCEDURE top_clienti_profit (p_suma NUMBER)
IS
    CURSOR c_comenzi
    IS
        SELECT id_client, valoare_comanda(nr_comanda) as valoare
        FROM comanda_benzi_desenate
        WHERE valoare_comanda(nr_comanda) > p_suma
        ORDER BY valoare_comanda(nr_comanda) DESC;

    v_suma_maxima NUMBER;

    suma_aberanta EXCEPTION;

BEGIN

    SELECT valoare_comanda(nr_comanda)
    INTO v_suma_maxima
    FROM comanda_benzi_desenate
    ORDER BY valoare_comanda(nr_comanda) DESC
    FETCH FIRST ROW ONLY;

    IF (p_suma > v_suma_maxima) THEN
        RAISE suma_aberanta;
    END IF;

    FOR var IN c_comenzi LOOP
        DBMS_OUTPUT.PUT_LINE(afisare_client(var.id_client) || ' ' || var.valoare);
```

```

END LOOP;

EXCEPTION
    WHEN suma_aberanta THEN
        DBMS_OUTPUT.PUT_LINE('Ati introdus o suma mult prea mare!');

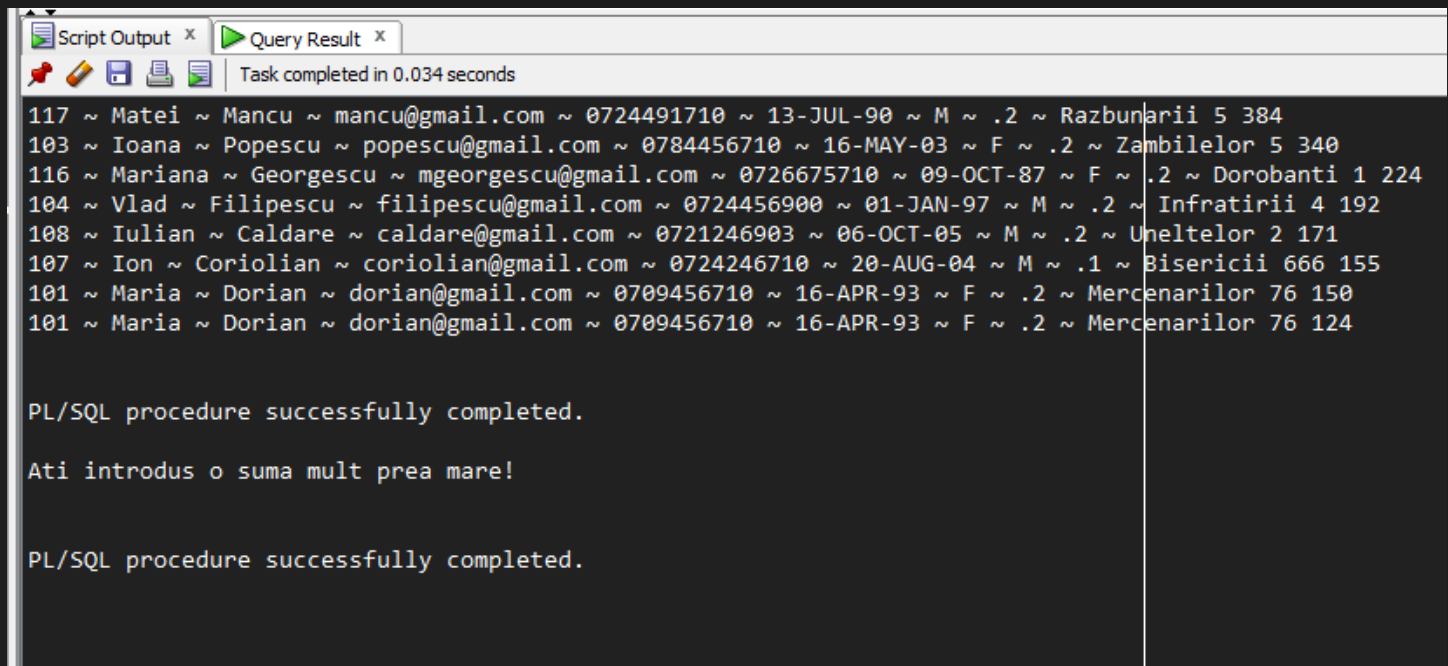
END;

--apel

EXECUTE top_clienti_profit(100);

EXECUTE top_clienti_profit(10000);

```



```

Script Output x Query Result x
Task completed in 0.034 seconds

117 ~ Matei ~ Mancu ~ mancu@gmail.com ~ 0724491710 ~ 13-JUL-90 ~ M ~ .2 ~ Razbunarii 5 384
103 ~ Ioana ~ Popescu ~ popescu@gmail.com ~ 0784456710 ~ 16-MAY-03 ~ F ~ .2 ~ Zambilelor 5 340
116 ~ Mariana ~ Georgescu ~ mgeorgescu@gmail.com ~ 0726675710 ~ 09-OCT-87 ~ F ~ .2 ~ Dorobanti 1 224
104 ~ Vlad ~ Filipescu ~ filipescu@gmail.com ~ 0724456900 ~ 01-JAN-97 ~ M ~ .2 ~ Infratirii 4 192
108 ~ Iulian ~ Caldare ~ caldare@gmail.com ~ 0721246903 ~ 06-OCT-05 ~ M ~ .2 ~ Uneltelor 2 171
107 ~ Ion ~ Coriolian ~ coriolian@gmail.com ~ 0724246710 ~ 20-AUG-04 ~ M ~ .1 ~ Bisericii 666 155
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 150
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 124

PL/SQL procedure successfully completed.

Ati introdus o suma mult prea mare!

PL/SQL procedure successfully completed.

```

--4. afisare venit real rezultat in urma comenzii primite ca parametru

```

CREATE OR REPLACE PROCEDURE venit_real (p_nr comanda_benzi_desenate.nr_comanda%type)
IS
    v_valoare NUMBER;

BEGIN
    v_valoare := valoare_comanda(p_nr);

    IF (v_valoare IS NULL) THEN
        RAISE NO_DATA_FOUND;
    END IF;

    v_valoare := v_valoare - (v_valoare - 0.1 * v_valoare - 0.5 * (v_valoare -
0.1 * v_valoare ));

```

```

DBMS_OUTPUT.PUT_LINE('Valoarea comenzii ' || p_nr || ': ' || v_valoare || ' lei');

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Comanda pentru care doriti sa aflati venitul nu exista!');

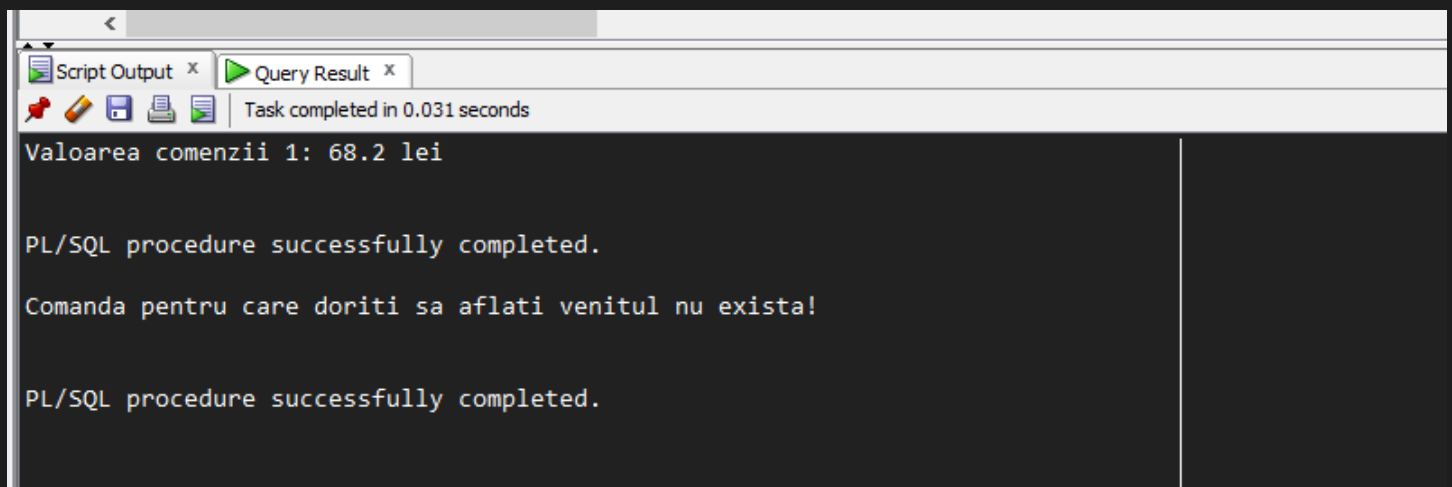
END;

--apel

EXECUTE venit_real(1);

EXECUTE venit_real(101001001010101);

```



Script Output x Query Result x

Task completed in 0.031 seconds

```

Valoarea comenzii 1: 68.2 lei

PL/SQL procedure successfully completed.

Comanda pentru care doriti sa aflati venitul nu exista!

PL/SQL procedure successfully completed.

```

```

--functii

--1. afisare detalii produs

CREATE OR REPLACE FUNCTION afisare (p_isbn benzi_desenate.isbn%type)
RETURN VARCHAR2
IS

    v_string VARCHAR(100);

    CURSOR c_benzi
    IS
        SELECT *
        FROM benzi_desenate
        WHERE isbn = p_isbn;

BEGIN

    FOR var in c_benzi LOOP

```

```

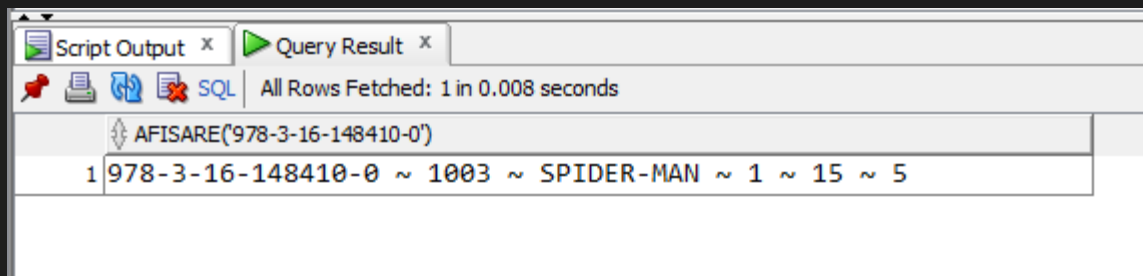
        v_string := var.isbn || ' ~ ' || var.id_editura || ' ~ ' || var.serie || ' ~ ' || var.nr_
episod || ' ~ ' || var.pret || ' ~ ' || var.cantitate;
    END LOOP;

    RETURN v_string;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN NULL;
END;

--apel
SELECT afisare('978-3-16-148410-0') FROM DUAL;

```



The screenshot shows a database query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying the result of the query 'AFISARE('978-3-16-148410-0')'. The result is a single row with the following data: '1 978-3-16-148410-0 ~ 1003 ~ SPIDER-MAN ~ 1 ~ 15 ~ 5'. The window also shows a status bar indicating 'All Rows Fetched: 1 in 0.008 seconds'.

	AFISARE('978-3-16-148410-0')
1	978-3-16-148410-0 ~ 1003 ~ SPIDER-MAN ~ 1 ~ 15 ~ 5

--2. calculare nota medie rezultata in urma review-urilor

```

CREATE OR REPLACE FUNCTION calcul_nota_medie (p_isbn benzi_desenate.isbn%type)
RETURN NUMBER
IS
    v_nota NUMBER;

BEGIN

    SELECT ROUND(SUM(r.nota)/COUNT(r.nota), 2)
    INTO v_nota
    FROM rebede r, bede b, edbede e
    WHERE r.isbn = b.isbn AND b.id_editura = e.id_editura AND b.isbn = p_isbn
    GROUP BY b.isbn
    ORDER BY SUM(r.nota)/COUNT(r.nota);

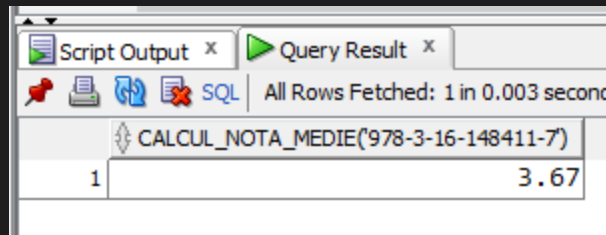
    RETURN v_nota;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN NULL;

END;

--apel
SELECT calcul_nota_medie('978-3-16-148411-7') FROM DUAL;

```

CALCUL_NOTA_MEDIE('978-3-16-148411-7')	
1	3.67

--3. afisare client

```
CREATE OR REPLACE FUNCTION afisare_client (p_id clienti_benzi_desenate.id_client%type)
RETURN VARCHAR2
IS
    v_string VARCHAR(100);

    CURSOR c_client
    IS
    SELECT *
    FROM clienti_benzi_desenate
    WHERE id_client = p_id;

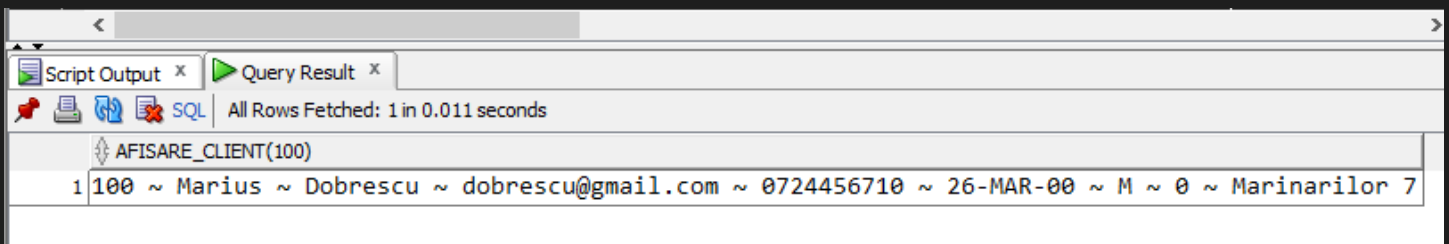
BEGIN
    FOR var IN c_client LOOP
        v_string := var.id_client || ' ~ ' || var.prenume_client || ' ~ ' || var.numa_client || '
~ ' || var.email_client || ' ~ ' || var.numar_telefon_client || ' ~ ' || var.data_nastere ||
' ~ ' || var.sex || ' ~ ' || NVL(var.discount, 0) || ' ~ ' || var.adresa_client;
    END LOOP;

    RETURN v_string;

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

--apel

```
SELECT afisare_client(100) FROM DUAL;
```



AFISARE_CLIENT(100)	
1	100 ~ Marius ~ Dobrescu ~ dobrescu@gmail.com ~ 0724456710 ~ 26-MAR-00 ~ M ~ 0 ~ Marinariilor 7

--4. calcul valoare comanda

```
CREATE OR REPLACE FUNCTION valoare_comanda (p_nr comanda_benzi_desenate.nr_comanda%type)
RETURN NUMBER
```

IS

```
v_valoare NUMBER;
```

BEGIN

```
SELECT SUM(b.pret * r.cantitate) as valoare  
INTO v_valoare  
FROM rand_comanda_benzi_desenate r, benzi_desenate b  
WHERE nr_comanda = p_nr AND r.isbn = b.isbn;
```

```
RETURN v_valoare;
```

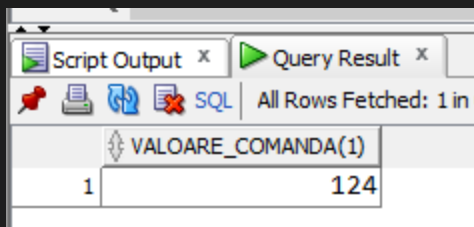
EXCEPTION

```
WHEN NO_DATA_FOUND THEN  
    RETURN NULL;
```

END;

--apel

```
SELECT valoare_comanda(1) FROM DUAL;
```



The screenshot shows a SQL query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying a table with one column named 'VALOARE_COMANDA(1)' and one row containing the value '124'. The status bar indicates 'All Rows Fetched: 1 in 0'.

VALOARE_COMANDA(1)
124

```
--pachet top care grupeaza toate procedurile ce afiseaza topuri
```

```
CREATE OR REPLACE PACKAGE top  
AS
```

```
    PROCEDURE top_benzi (p_n NUMBER);  
    PROCEDURE top_clienti (p_n NUMBER);  
    PROCEDURE top_clienti_profit (p_suma NUMBER);
```

```
END;
```

```
CREATE OR REPLACE PACKAGE BODY top  
AS
```

```
    PROCEDURE top_benzi (p_n NUMBER)  
    IS  
  
        CURSOR c_benzi  
        IS  
        SELECT isbn, calcul_nota_medie(isbn) as nota  
        FROM benzi_desenate  
        ORDER BY calcul_nota_medie(isbn) DESC  
        FETCH FIRST p_n ROWS ONLY;
```

```
        v_numar_produce NUMBER;
```

```
        numar_aberant_produce EXCEPTION;
```

```
BEGIN
```

```
    SELECT COUNT(isbn)  
    INTO v_numar_produce FROM benzi_desenate;
```

```
    IF (p_n > v_numar_produce) THEN
```

```
        RAISE numar_aberant_produce;
```

```
    END IF;
```

```
    FOR var in c_benzi LOOP
```

```
        DBMS_OUTPUT.PUT_LINE(afisare(var.isbn) || ' ' || var.nota);
```

```
    END LOOP;
```

```
EXCEPTION
```

```
    WHEN numar_aberant_produce THEN
```

```
        DBMS_OUTPUT.PUT_LINE('Numarul de produse cerut este mult mai mare decat numarul t  
otal de produse aflate in sistem!');
```

```
END;
```

```

PROCEDURE top_clienti (p_n NUMBER)
IS

    CURSOR c_comenzi
    IS
    SELECT id_client, valoare_comanda(nr_comanda) as valoare
    FROM comanda_benzi_desenate
    ORDER BY valoare_comanda(nr_comanda) DESC
    FETCH FIRST p_n ROWS ONLY;
    v_numar_clienti NUMBER;
    numar_aberant_clienti EXCEPTION;

BEGIN

    SELECT COUNT(id_client)
    INTO v_numar_clienti
    FROM clienti_benzi_desenate;

    IF (p_n > v_numar_clienti) THEN
        RAISE numar_aberant_clienti;
    END IF;

    FOR var IN c_comenzi LOOP
        DBMS_OUTPUT.PUT_LINE(afisare_client(var.id_client) || ' ' || var.valoare);
    END LOOP;

EXCEPTION

    WHEN numar_aberant_clienti THEN
        DBMS_OUTPUT.PUT_LINE('Numarul de clienti cerut este mult mai mare decat numarul
total de clienti aflati in sistem!');

END;

PROCEDURE top_clienti_profit (p_suma NUMBER)
IS

    CURSOR c_comenzi
    IS
    SELECT id_client, valoare_comanda(nr_comanda) as valoare
    FROM comanda_benzi_desenate
    WHERE valoare_comanda(nr_comanda) > p_suma
    ORDER BY valoare_comanda(nr_comanda) DESC;

    v_suma_maxima NUMBER;

    suma_aberanta EXCEPTION;

BEGIN

```

```

SELECT valoare_comanda(nr_comanda)
INTO v_suma_maxima
FROM comanda_benzi_desenate
ORDER BY valoare_comanda(nr_comanda) DESC
FETCH FIRST ROW ONLY;

IF (p_suma > v_suma_maxima) THEN
    RAISE suma_aberanta;
END IF;

FOR var IN c_comenzi LOOP
    DBMS_OUTPUT.PUT_LINE(afisare_client(var.id_client) || ' ' || var.valoare); END L
OOP;

EXCEPTION

    WHEN suma_aberanta THEN
        DBMS_OUTPUT.PUT_LINE('Ati introdus o suma mult prea mare!');
END;

END;

EXECUTE top.top_benzi(19009999);
EXECUTE top.top_benzi(12);

EXECUTE top.top_clienti(10);
EXECUTE top.top_clienti(500);

EXECUTE top.top_clienti_profit(100);
EXECUTE top.top_clienti_profit(10000);

```

Numarul de produse cerut este mult mai mare decat numarul total de produse aflate in sistem!

PL/SQL procedure successfully completed.

906-3-16-101410-1 ~ 1000 ~ MACHINE-MAN ~ 6 ~ 18 ~ 100 4
978-3-16-148411-2 ~ 1003 ~ SPIDER-MAN ~ 9 ~ 15 ~ 100 4
978-3-16-148410-8 ~ 1003 ~ SPIDER-MAN ~ 13 ~ 16 ~ 100 4
978-3-16-148410-5 ~ 1003 ~ SPIDER-MAN ~ 16 ~ 16 ~ 100 4
978-3-16-018410-1 ~ 1004 ~ IRON-MAN ~ 1 ~ 16 ~ 100 4
978-3-16-112410-1 ~ 1001 ~ HILLBILLIES ~ 67 ~ 10 ~ 7 3.67
978-3-16-148411-7 ~ 1003 ~ SPIDER-MAN ~ 4 ~ 15 ~ 100 3.67
902-3-16-101410-1 ~ 1000 ~ MACHINE-MAN ~ 2 ~ 18 ~ 100 3.5
978-3-16-148411-1 ~ 1003 ~ SPIDER-MAN ~ 10 ~ 15 ~ 100 3.5
978-3-16-148410-4 ~ 1003 ~ SPIDER-MAN ~ 17 ~ 16 ~ 100 3.5
978-3-16-148410-1 ~ 1003 ~ SPIDER-MAN ~ 20 ~ 16 ~ 100 3.5
978-3-16-088410-1 ~ 1004 ~ IRON-MAN ~ 8 ~ 16 ~ 100 3.5

PL/SQL procedure successfully completed.

117 ~ Matei ~ Mancu ~ mancu@gmail.com ~ 0724491710 ~ 13-JUL-90 ~ M ~ .2 ~ Razbunarii 5 384
103 ~ Ioana ~ Popescu ~ popescu@gmail.com ~ 0784456710 ~ 16-MAY-03 ~ F ~ .2 ~ Zambilelor 5 340
116 ~ Mariana ~ Georgescu ~ mgeorgescu@gmail.com ~ 0726675710 ~ 09-OCT-87 ~ F ~ .2 ~ Dorobanti 1 224
104 ~ Vlad ~ Filipescu ~ filipescu@gmail.com ~ 0724456900 ~ 01-JAN-97 ~ M ~ .2 ~ Infratirii 4 192
108 ~ Iulian ~ Caldare ~ caldare@gmail.com ~ 0721246903 ~ 06-OCT-05 ~ M ~ .2 ~ Uneltelor 2 171
107 ~ Ion ~ Coriolian ~ coriolian@gmail.com ~ 0724246710 ~ 20-AUG-04 ~ M ~ .1 ~ Bucuresti 155
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 150
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 124
110 ~ Valentin ~ Onetiu ~ onetiu@gmail.com ~ 0739956710 ~ 30-NOV-83 ~ M ~ .05 ~ Revolutiei 89 90
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 54

PL/SQL procedure successfully completed.

Numarul de clienti cerut este mult mai mare decat numarul total de clienti aflati in sistem!

PL/SQL procedure successfully completed.

117 ~ Matei ~ Mancu ~ mancu@gmail.com ~ 0724491710 ~ 13-JUL-90 ~ M ~ .2 ~ Razbunarii 5 384
103 ~ Ioana ~ Popescu ~ popescu@gmail.com ~ 0784456710 ~ 16-MAY-03 ~ F ~ .2 ~ Zambilelor 5 340
116 ~ Mariana ~ Georgescu ~ mgeorgescu@gmail.com ~ 0726675710 ~ 09-OCT-87 ~ F ~ .2 ~ Dorobanti 1 224
104 ~ Vlad ~ Filipescu ~ filipescu@gmail.com ~ 0724456900 ~ 01-JAN-97 ~ M ~ .2 ~ Infratirii 4 192
108 ~ Iulian ~ Caldare ~ caldare@gmail.com ~ 0721246903 ~ 06-OCT-05 ~ M ~ .2 ~ Uneltelor 2 171
107 ~ Ion ~ Coriolian ~ coriolian@gmail.com ~ 0724246710 ~ 20-AUG-04 ~ M ~ .1 ~ Bucuresti 155
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 150
101 ~ Maria ~ Dorian ~ dorian@gmail.com ~ 0709456710 ~ 16-APR-93 ~ F ~ .2 ~ Mercenarilor 76 124

PL/SQL procedure successfully completed.

Ati introdus o suma mult prea mare!

PL/SQL procedure successfully completed.

```
--pachet care grupeaza functiile de afisare
```

```
CREATE OR REPLACE PACKAGE afisarea  
AS
```

```
    FUNCTION afisare_produs (p_isbn benzi_desenate.isbn%type) RETURN VARCHAR2;  
    FUNCTION afisare_client (p_id clienti_benzi_desenate.id_client%type) RETURN VARCHAR2;  
    FUNCTION afisare_editura (p_id edituri_benzi_desenate.id_editura%type) RETURN VARCHAR2;  
    FUNCTION afisare_comanda (p_id comanda_benzi_desenate.nr_comanda%type) RETURN VARCHAR2;
```

```
END;
```

```
CREATE OR REPLACE PACKAGE BODY afisarea  
AS
```

```
    FUNCTION afisare_produs (p_isbn benzi_desenate.isbn%type)  
    RETURN VARCHAR2  
    AS
```

```
        v_string VARCHAR(100);  
        CURSOR c_benzi  
        IS  
        SELECT *  
        FROM benzi_desenate  
        WHERE isbn = p_isbn;
```

```
BEGIN
```

```
    FOR var IN c_benzi LOOP  
        v_string := var.isbn || ' ~ ' || var.id_editura || ' ~ ' || var.serie || ' ~ ' || va  
r.nr_episod || ' ~ ' || var.pret || ' ~ ' || var.cantitate;  
    END LOOP;  
  
    RETURN v_string;
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN  
        RETURN NULL;
```

```
END;
```

```
FUNCTION afisare_client (p_id clienti_benzi_desenate.id_client%type)  
RETURN VARCHAR2  
AS
```

```
    v_string VARCHAR(100);  
    CURSOR c_client  
    IS  
    SELECT *
```

```
FROM clienti_benzi_desenate
WHERE id_client = p_id;
```

```
BEGIN
```

```
    FOR var IN c_client LOOP
        v_string := var.id_client || ' ~ ' || var.prenume_client || ' ~ ' || var.num_client
        || ' ~ ' || var.email_client || ' ~ ' || var.numar_telefon_client || ' ~ ' || var.data_naste
re || ' ~ ' || var.sex || ' ~ ' || NVL(var.discount, 0) || ' ~ ' || var.adresa_client;
    END LOOP;
```

```
    RETURN v_string;
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN
        RETURN NULL;
```

```
END;
```

```
FUNCTION afisare_editura (p_id edituri_benzi_desenate.id_editura%type)
RETURN VARCHAR2
AS
```

```
    v_string VARCHAR(100);
    CURSOR c_editura
    IS
    SELECT *
    FROM edituri_benzi_desenate
    WHERE id_editura = p_id;
```

```
BEGIN
```

```
    FOR var IN c_editura LOOP
        v_string := var.id_editura || ' ~ ' || var.num_client || ' ~ ' || var.numar_telefon || ' ~
' || var.url || ' ~ ' || var.email_editura || ' ~ ' || var.adresa;
    END LOOP;
```

```
    RETURN v_string;
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN
        RETURN NULL;
```

```
END;
```

```
FUNCTION afisare_comanda (p_id comanda_benzi_desenate.nr_comanda%type)
RETURN VARCHAR2
AS
```



```

        v_string VARCHAR(100);

        CURSOR c_comanda
        IS
        SELECT *
        FROM rand_comanda_benzi_desenate
        WHERE nr_comanda = p_id;

BEGIN

        FOR var IN c_comanda LOOP
            v_string := v_string || (var.nr_comanda || ' ~ ' || var.isbn || ' ~ ' || var.cantita
te || '\n| ');
        END LOOP;

        RETURN v_string;

EXCEPTION

        WHEN NO_DATA_FOUND THEN
            RETURN NULL;

END;

END;

BEGIN
    DBMS_OUTPUT.PUT_LINE(afisarea.afisare_comanda(1));
    DBMS_OUTPUT.PUT_LINE(afisarea.afisare_client(100));
    DBMS_OUTPUT.PUT_LINE(afisarea.afisare_editura(1000));
    DBMS_OUTPUT.PUT_LINE(afisarea.afisare_produs('978-3-16-108410-1'));
END;

```

Script Output x

Task completed in 0.023 seconds

```

1 ~ 978-3-16-108410-1 ~ 4\n| 1 ~ 978-3-16-148411-5 ~ 4\n|
100 ~ Marius ~ Dobrescu ~ dobrescu@gmail.com ~ 0724456710 ~ 26-MAR-00 ~ M ~ 0 ~ Marinariilor 7
1000 ~ ARTHUR ~ 0212456731 ~ www.arthur.ro ~ arthur@gmail.com ~ Tipografiei 1
978-3-16-108410-1 ~ 1004 ~ IRON-MAN ~ 10 ~ 16 ~ 7

PL/SQL procedure successfully completed.

```

--trigger care actualizeaza stocul de produse

```
CREATE OR REPLACE TRIGGER stoc_check
BEFORE INSERT OR UPDATE OF cantitate ON rand_comanda_benzi_desenate
FOR EACH ROW

DECLARE
    v_stoc NUMBER;
BEGIN

    SELECT cantitate
    INTO v_stoc
    FROM benzi_desenate
    WHERE isbn = :NEW.isbn;

    IF :NEW.cantitate > v_stoc THEN
        RAISE_APPLICATION_ERROR(-
20000, 'Atentie! Nu poti comanda o cantitate asa de mare. ');
    END IF;

    IF INSERTING THEN
        UPDATE benzi_desenate SET cantitate = cantitate - :NEW.cantitate;
    ELSE
        IF (:NEW.cantitate > :OLD.cantitate) THEN
            UPDATE benzi_desenate set cantitate = cantitate - (:NEW.cantitate - :OLD.cantitate);
        END IF;

        IF (:NEW.cantitate < :OLD.cantitate) THEN
            UPDATE benzi_desenate set cantitate = cantitate + (:OLD.cantitate - :NEW.cantitate);
        END IF;

    END IF;

END;

SELECT cantitate
FROM benzi_desenate
WHERE isbn = '978-3-16-018410-1';

INSERT INTO rand_comanda_benzi_desenate (nr_comanda, isbn, cantitate)
VALUES ('1', '978-3-16-018410-1', '5');

SELECT cantitate
FROM benzi_desenate
WHERE isbn = '978-3-16-018410-1';
```

CANTITATE	
1	95

1 row inserted.

CANTITATE	
1	90

--trigger care sterge inregistrarile din rand_comanda atunci cand comanda este stearsa

```
CREATE OR REPLACE TRIGGER delete_comanda
BEFORE DELETE ON comanda_benzi_desenate
FOR EACH ROW
```

```
BEGIN
```

```
    IF DELETING THEN
        DELETE FROM rand_comanda_benzi_desenate
        WHERE nr_comanda = :OLD.nr_comanda;
    END IF;
```

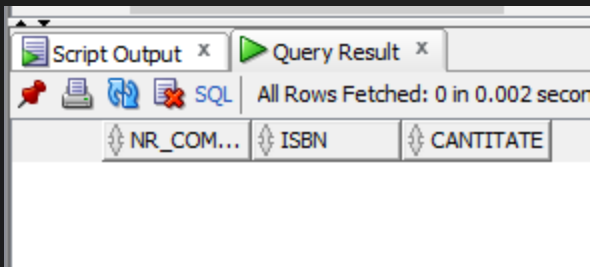
```
END;
```

```
DELETE FROM comanda_benzi_desenate
WHERE nr_comanda = '1';
```

```
SELECT *
FROM rand_comanda_benzi_desenate
WHERE nr_comanda = '1';
```

Trigger DELETE_COMANDA compiled

1 row deleted.



```
--trigger care genereaza id-ul clientilor
```

```
CREATE SEQUENCE clienti_secv
START WITH 100
INCREMENT BY 1
MAXVALUE 1000
NOCYCLE;
```

```
CREATE OR REPLACE TRIGGER genereare_idclienti
BEFORE INSERT ON clienti_benzi_desenate
FOR EACH ROW
BEGIN
    SELECT clienti_secv.nextval INTO :new.id_client FROM dual;
END;
```

```
INSERT INTO clienti_benzi_desenate (ID_CLIENT, NUME_CLIENT, PRENUME_CLIENT, DATA_NASTERE, ADR
ESA_CLIENT, EMAIL_CLIENT, SEX, NUMAR_TELEFON_CLIENT)
VALUES ('200', 'Dodescu', 'Marin', TO_DATE('26.03.2001', 'DD.MM.YYYY'), 'Marinarilor 9', 'dod
escu@gmail.com', 'M', '0724326710');
```

```
Trigger GENERARE_IDCLIENTI compiled
```

```
1 row inserted.
```

19	120	Dodescu	Marin	26-MAR-01	Marinarilor 9	
----	-----	---------	-------	-----------	---------------	--