

Proiect SGBD

Anul II, Semestrul I

1. Prezentați pe scurt baza de date (utilitatea ei).

Anticariatele reprezinta pentru unii oameni o poarta catre trecut. Astfel, multi pasionati de literatura cauta ore in sir prin rafturi si dulapuri cartea ce le va umple sufletul de bucurie. Din pacate, cateodata, in ciuda faptului ca vanzatorul isi aminteste perfect ca detine anumite exemplare in stoc, exista posibilitatea ca acestea sa nu fie gasite nici in magazine, nici in depozit fie din cauza dezordinii, fie din cauza omiterii trecerii in registru.

Proiectul meu propune o baza de date cu un design simplist, ce este usor de folosit de catre oricine si usor de inteles pentru a rezolva problemele legate de stoc sau valoare. De asemenea, cu ajutorul acestei baze de date, clientul poate afla pareri si impresii de la alti cumparatori in sectiunea de recenzii. Astfel, atat cumparatorii, cat si anticariatele pot gestiona mai bine vanzarile si stocul produselor, limitandu-se considerabil atat timpul de cautare, cat si pierderile survenite din greselile vanzatorilor si personalului administrativ. Aceasta baza de date va scadea semnificativ pierderile si va maximiza profitul.

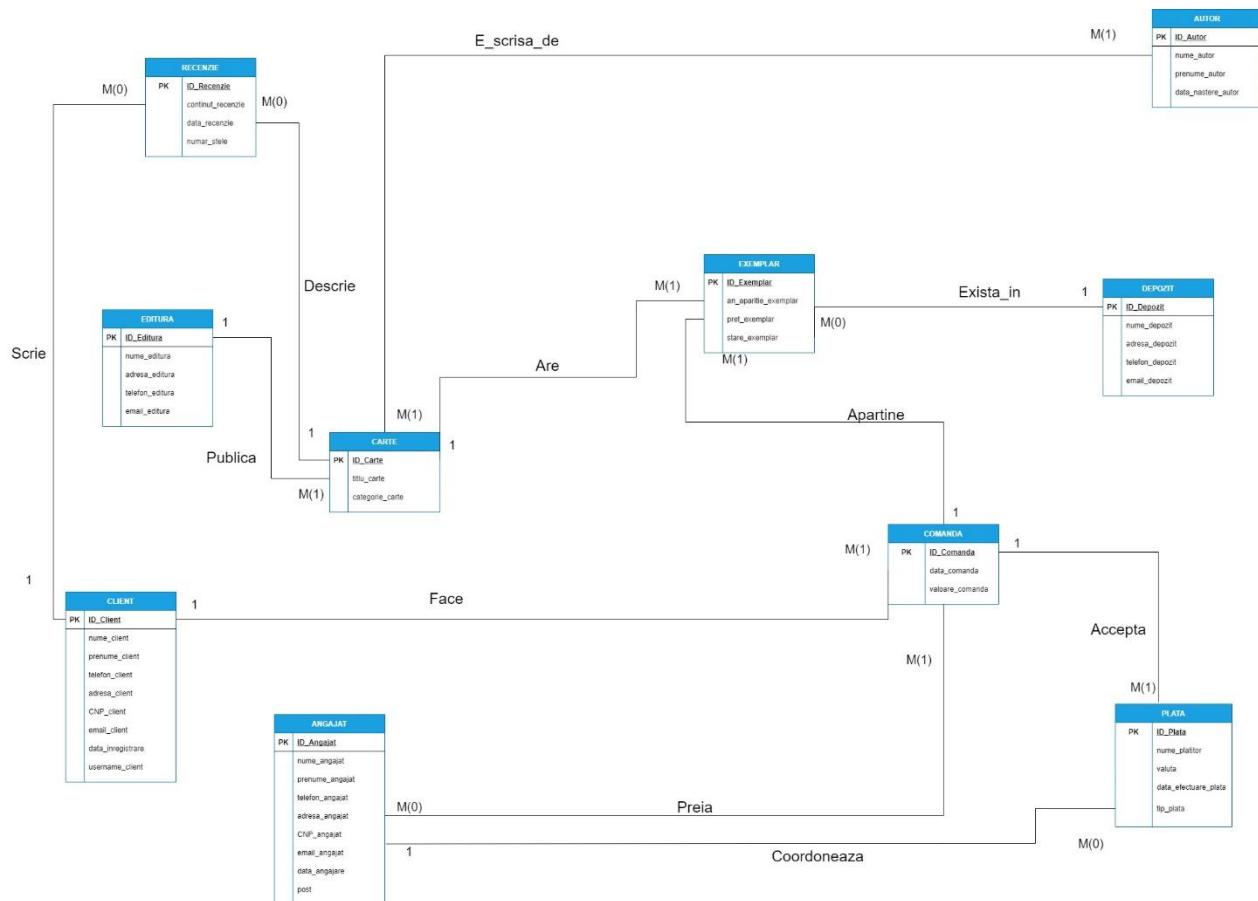
Proiectul are 10 entitati independente (CLIENT, ANGAJAT, PLATA, COMANDA, CARTE, EDITURA, RECENZIE, EXEMPLAR, DEPOZIT, AUTOR), dar si 2 tabele asociative (PRELUARE, CREATIE). Acestea rezolva legaturile Many to Many dintre tabelele ANGAJAT si COMANDA, respectiv dintre AUTOR si CARTE.

Reguli de functionare:

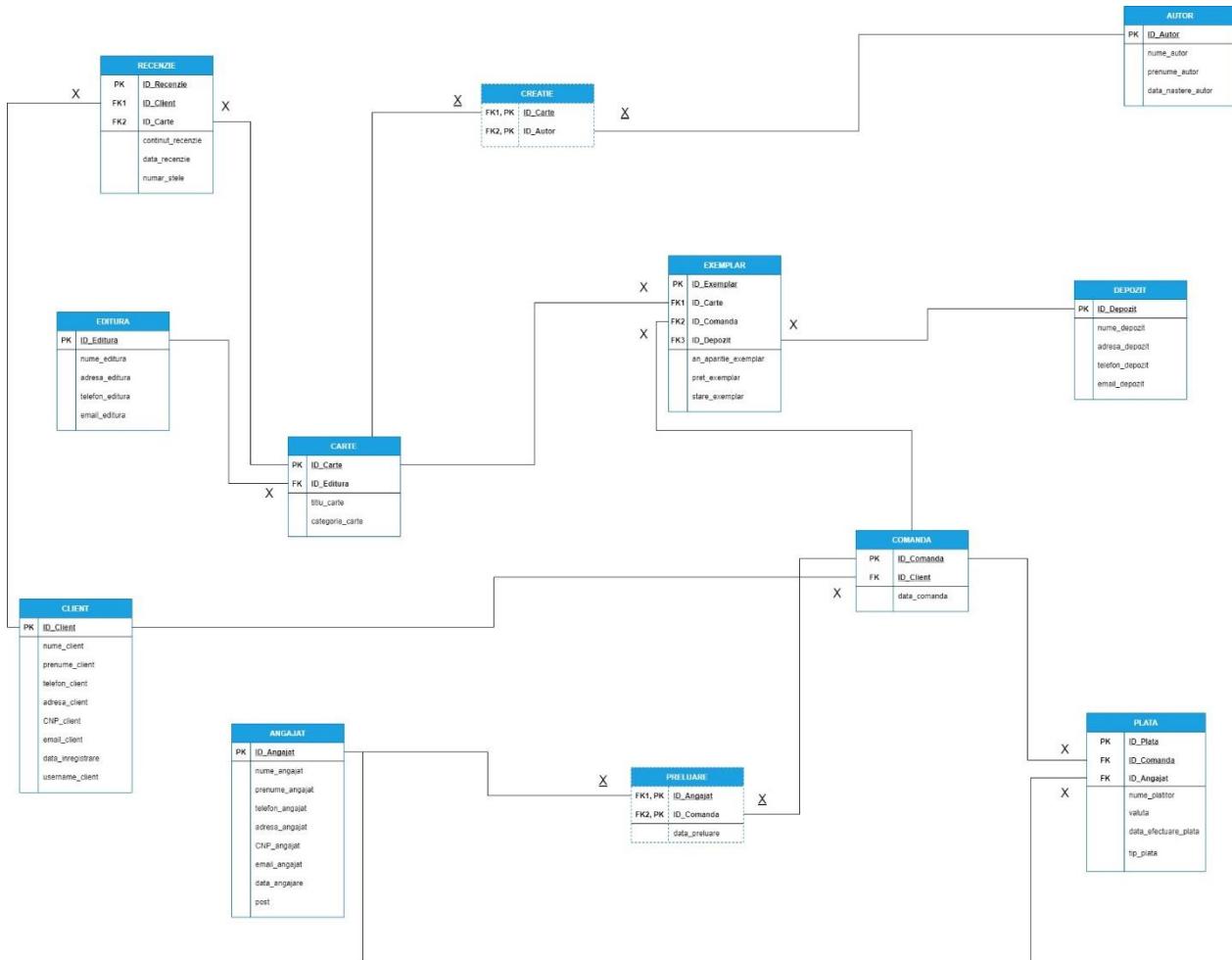
- Un anagajat poate sa nu aiba un post, acesta fiind managerul

- Orice client trebuie sa aiba un username
- Un client poate face mai multe comenzi, dar poate sa fie client inregistrat si daca nu a facut nici macar o comanda
- O editura poate avea mai multe carti publicate, dar poate sa nu aiba nicio carte publicata.
- Un exemplar trebuie neaparat sa fie vandut ca sa se afle in tabel, deci sa apartina unei comenzi, acesta neputand fi in mai multe comenzi.
- Un depozit poate gazdui mai multe exemplare, dar poate fi si gol (sa nu aiba niciun exemplar in el).
- Plata este mereu efectuata de catre casier, dar alti angajati pot prelua comenzi.
- Recenzia trebuie sa contine numar de stele, dar descrierea poate lipsi.
- O comanda poate include mai multe exemplare.

2. Realizați diagrama entitate-relație (ERD).



3. Pornind de la diagrama entitate-relație realizată **diagrama conceptuală** a modelului propus, integrând toate attributele necesare.



4. Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, implementând toate constrângerile de integritate necesare (chei primare, cheile externe etc).

```
create table client(
    id_client number(5) primary key,
    nume_client varchar2(55) not null,
    prenume_client varchar2(55) not null,
    telefon_client varchar2(12),
    adresa_client varchar2(100),
```

```
cnp_client varchar2(15),
email_client varchar2(55),
data_inregistrare date not null,
username_client varchar2(55) not null
);
```

```
create table angajat(
id_angajat number(5) primary key,
nume_angajat varchar2(55) not null,
prenume_angajat varchar2(55) not null,
telefon_angajat varchar2(12),
adresa_angajat varchar2(100),
cnp_angajat varchar2(15),
email_angajat varchar2(55),
data_angajare date not null,
post varchar2(55)
);
```

```
create table editura(
id_editura number(5) primary key,
nume_editura varchar2(55) not null,
adresa_editura varchar2(100),
telefon_editura varchar2(12),
email_editura varchar2(55)
);
```

```
create table autor(
id_autor number(5) primary key,
nume_autor varchar2(55) not null,
prenume_autor varchar2(55) not null,
data_nasterii date
);
```

```
create table depozit(
id_depozit number(5) primary key,
nume_depozit varchar2(55) not null,
adresa_depozit varchar2(100),
telefon_depozit varchar2(12),
email_depozit varchar2(55)
);
```

```
create table carte(
    id_carte number(5) primary key,
    id_editura number(5) not null,
    titlu_carte varchar2(55) not null,
    categorie_carte varchar2(55),
    foreign key (id_editura) references editura (id_editura) on delete cascade
);
```

```
create table recenzie(
    id_recenzie number(5) primary key,
    id_client number(5) not null,
    id_carte number(5) not null,
    continut_recenzie varchar2(200),
    data_recenzie date not null,
    numar_stele number(1) not null,
    foreign key (id_client) references client (id_client) on delete cascade,
    foreign key (id_carte) references carte (id_carte) on delete cascade
);
```

```
create table comanda(
    id_comanda number(5) primary key,
    id_client number(5) not null,
    data_comanda date not null,
    foreign key (id_client) references client(id_client) on delete cascade
);
```

```
create table preluare(
    id_angajat number(5) not null,
    id_comanda number(5) not null,
    data_preluare date not null,
    constraint PK_preluare primary key (id_angajat, id_comanda),
    foreign key(id_angajat) references angajat(id_angajat) on delete cascade,
    foreign key(id_comanda) references comanda(id_comanda) on delete cascade
);
```

```
create table creatie(
    id_carte number(5) not null,
    id_autor number(5) not null,
    constraint PK_creatie primary key (id_carte, id_autor),
```

```
foreign key(id_carte) references carte(id_carte) on delete cascade,  
foreign key(id_autor) references autor(id_autor) on delete cascade  
);
```

```
create table plata(  
    id_plata number(5) primary key,  
    id_comanda number(5) not null,  
    id_angajat number(5) not null,  
    nume_platitor varchar2(55),  
    valuta varchar2(10) not null,  
    data_efectuare_plata date not null,  
    tip_plata varchar2(55),  
    foreign key(id_comanda) references comanda(id_comanda) on delete cascade,  
    foreign key(id_angajat) references angajat(id_angajat) on delete cascade  
);
```

```
create table exemplar(  
    id_exemplar number(5) primary key,  
    id_carte number(5) not null,  
    id_comanda number(5) not null,  
    id_depozit number(5) not null,  
    an_aparitie_exemplar number(4),  
    pret_exemplar numeric(4,2) not null,  
    stare_exemplar varchar2(55),  
    foreign key(id_carte) references carte(id_carte) on delete cascade,  
    foreign key(id_comanda) references comanda(id_comanda) on delete cascade,  
    foreign key(id_depozit) references depozit(id_depozit) on delete cascade  
);
```

The screenshot shows two instances of Oracle SQL Developer running side-by-side. Both instances are connected to a project named 'SGBD_Project'. The left instance displays the creation of the following tables:

- Table CLIENT created.
- Table ANGAJAT created.
- Table EDITURA created.
- Table AUTOR created.
- Table DEPOZIT created.
- Table CARTE created.
- Table RECENZIE created.
- Table COMANDA created.

The right instance displays the creation of the following tables:

- Table DEPOZIT created.
- Table CARTE created.
- Table RECENZIE created.
- Table COMANDA created.
- Table PRELUARE created.
- Table CREATIE created.
- Table PLATA created.
- Table EXEMPLAR created.

Both instances show a 'Script Output' window indicating that tasks completed in 0.583 seconds. The system tray at the bottom of the screen shows the date as 31-Dec-21, the time as 17:45, and the weather as 3°C Partly cloudy.

5. Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

-----INSERT-----

```
insert into client values(1, 'Popescu', 'Ioan', '0723595269','Str. Lalelor 108, Tg. Jiu', '5010304678823','ipop@gmail.com', to_date('23/06/2020', 'DD/MM/YYYY'), 'ipopescu');
insert into client values(2, 'Radu', 'Ioana', '0723293170','Str. Florilor 2, Petrosani', '6010611672723','ioanar@gmail.com', to_date('24/06/2020', 'DD/MM/YYYY'), 'ioanair');
insert into client values(3, 'Mihai', 'Alexandra', '0723464287','Str. Paris 8, Bucuresti', '611020372710','am@gmail.com', to_date('30/08/2020', 'DD/MM/YYYY'), 'alm');
insert into client values(4, 'Banul', 'Radu', '0789124766','Bvd. Libertatii 32, Craiova', '514010777777','radub@gmail.com', to_date('25/02/2020', 'DD/MM/YYYY'), 'radub');
insert into client values(5, 'Aionese', 'Andrei', '0723666666','Str. Diamantelor 8, Botosani', '502030402012','aa@gmail.com', to_date('06/06/2019', 'DD/MM/YYYY'), 'aanda');
insert into client values(6, 'Alexandrescu', 'Paula', '0723868646','Str. Bibliotecii 3, Tg. Jiu', '600030744150','paulaalexandrescu2000@yahoo.com', to_date('12/12/2019', 'DD/MM/YYYY'), 'paulaalexandrescu');
insert into client values(7, 'Fulea', 'Andreea', '0723996996','Str. Turda 44, Urziceni', '605020149374','andre123@gmail.com', to_date('16/08/2020', 'DD/MM/YYYY'), 'andre1234');
insert into client values(8, 'Manea', 'Paul', '0723116256','Str. Sisesti 24, Bucuresti', '511020375780','pman@gmail.com', to_date('11/09/2018', 'DD/MM/YYYY'), 'paman');
insert into client values(9, 'Sorin', 'Maria', '0723099056','Str. Soarelui 88, Galati', null, null, to_date('30/08/2020', 'DD/MM/YYYY'), 'mssms');
insert into client values(10, 'Andrei', 'Adriana', '0723887723',null, '6130302737214','aand@gmail.com', to_date('04/02/2020', 'DD/MM/YYYY'), 'adand');

insert into angajat values(1, 'Mihailescu', 'Carina', '0723237924',null, null,'mcarina@gmail.com', to_date('04/02/2020', 'DD/MM/YYYY'), 'Casier');
```

```
insert into angajat values(2, 'Mihăilescu', 'Dorina', '0723337844', null,  
'6010302967214', 'mdorina@gmail.com', to_date('04/02/2020', 'DD/MM/YYYY'),  
'Vanzator');  
insert into angajat values(3, 'Georgescu', 'Daria', '0723852467', 'Str. Sisesti 222,  
Bucuresti', null, 'dariageo@gmail.com', to_date('05/02/2020', 'DD/MM/YYYY'),  
'Contabil');  
insert into angajat values(4, 'Vasilescu', 'Simona', '0723295844', 'Str. Garii 22,  
Buftea', '601030234433', 'vasimona@gmail.com', to_date('08/03/2020',  
'DD/MM/YYYY'), 'Responsabil Impachetare');  
insert into angajat values(5, 'Popescu', 'Mihaela', '0723196222', 'Str. Zorilor 11,  
Buftea', null, 'popmiha@gmail.com', to_date('09/04/2020', 'DD/MM/YYYY'),  
'Responsabil Impachetare');  
insert into angajat values(6, 'Pavaloiu', 'Anca', '0723894370', 'Str. Sbl. Mihai  
Cristian Oancea, Bl. 6, Ap.11, Tg. Jiu',  
'6000308967312', 'pavaloiuanca@gmail.com', to_date('20/12/2019',  
'DD/MM/YYYY'), 'Asistent Vanzari');  
insert into angajat values(7, 'Dumitrascu', 'Iulia', '0723742196', 'Str. Stadionului 123,  
Petrosani', null, 'iuliad@gmail.com', to_date('01/12/2019', 'DD/MM/YYYY'), null);  
insert into angajat values(8, 'Mosoi', 'Andrei', '0724592756', 'Sos. Grivitei 128C,  
Bucuresti', null, 'andreimosoi@gmail.com', to_date('09/12/2020',  
'DD/MM/YYYY'), 'Responsabil Inventar');  
insert into angajat values(9, 'Gherghe', 'Vladimir', '0723237345', 'Str. Garii 17,  
Brasov', '5010203050401', 'vlag@gmail.com', to_date('03/02/2021',  
'DD/MM/YYYY'), 'Serviciu Tehnic');  
insert into angajat values(10, 'Ceparu', 'Andrei', '0745063364', null,  
null, 'cepi2001@gmail.com', to_date('05/05/2021', 'DD/MM/YYYY'), 'Om de  
Serviciu');
```

```
insert into editura values(1, 'Editura Univers', 'Str. Dionisie Lupu Nr. 65, Bucuresti',  
'0758770777', null);  
insert into editura values(2, 'Editura Adevarul Holding', null, '0214077650',  
'carte@adevarul.ro');  
insert into editura values(3, 'Polirom', 'Bvd. Carol I nr. 4, Iasi', '0232214100',  
'office@polirom.ro');  
insert into editura values(4, 'Editura ART', 'Splaiul Independentei nr. 315-319,  
Bucuresti', '0744300870', 'comenzi@editura-art.ro');  
insert into editura values(5, 'LEDA', 'Calea Plevnei nr. 145, Bucuresti',  
'0213198822', 'vanzari@edituracorint.ro');
```

```
insert into editura values(6, 'Humanitas', 'Pta. Presei Libere nr. 1, Bucuresti',  
'0214088350', 'vanzari@libhumanitas.ro');  
insert into editura values(7, 'Bantam Books', '61-63 Uxbridge Road, London, UK',  
null, 'bdpublicity@randomhouse.com');  
insert into editura values(8, 'Editura Trei', 'O.P. 16, Ghiseul 1, Bucuresti',  
'0213006090', 'comenzi@editura3.ro');  
insert into editura values(9, 'Editura Litera', 'O.P. 53, Bucuresti', '0213196390',  
'comenzi@litera.ro');  
insert into editura values(10, 'Editura RAO', 'Str. Bargaului, nr. 9-11, Bucuresti',  
null, 'office@rao.ro');  
insert into editura values(11, 'Nemira', 'Str. Iani Buzoianu 14, Bucuresti',  
'0721747464', 'office@nemira.ro');  
insert into editura values(12, 'Weidenfeld and Nicolson', 'Carmelite House 50,  
London, UK', null, null);
```

```
insert into autor values(1, 'Brown', 'Dan', to_date('22/06/1964', 'DD/MM/YYYY'));  
insert into autor values(2, 'Bergler', 'Igor', to_date('21/09/1970', 'DD/MM/YYYY'));  
insert into autor values(3, 'Vancu', 'Radu', to_date('13/07/1978', 'DD/MM/YYYY'));  
insert into autor values(4, 'Zafon', 'Carlos Ruiz', to_date('25/09/1964',  
'DD/MM/YYYY'));  
insert into autor values(5, 'Sullivan', 'Mark', to_date('28/06/1958',  
'DD/MM/YYYY'));  
insert into autor values(6, 'Burnet', 'Graeme Macrae', to_date('01/01/1967',  
'DD/MM/YYYY'));  
insert into autor values(7, 'Orwell', 'George', to_date('25/06/1903',  
'DD/MM/YYYY'));  
insert into autor values(8, 'Stocker', 'Bram', to_date('08/11/1947',  
'DD/MM/YYYY'));  
insert into autor values(9, 'Wilde', 'Oscar', to_date('16/10/1854',  
'DD/MM/YYYY'));  
insert into autor values(10, 'Gerrard', 'Steven', to_date('30/05/1980',  
'DD/MM/YYYY'));  
insert into autor values(11, 'Jones', 'Lesley-Ann', to_date(null));
```

```
insert into depozit values(1, 'Depozitul de Carte', 'Str. Fantanii Nr. 65, Chiajna',  
'0758777877', 'depozituldecarte@yahoo.com');  
insert into depozit values(2, 'Depozitul de Carte Straina', 'Pta. Romana Nr. 9,  
Bucuresti', '021379877', 'sitka@sitka.ro');
```

```
insert into depozit values(3, 'Depozitul de Carti Vechi', 'Str. Cluj Nr. 88, Bucuresti',  
'0723667877', 'carteaveche@gmail.com');  
insert into depozit values(4, 'Depozitul Cartea Muntelui', 'Str. Postei Nr. 4, Brasov',  
'0758774513', 'cartteamunteluibv@yahoo.com');  
insert into depozit values(5, 'Depozitul Ursuletul Cititor', 'Str. Ursului Nr. 1, Ranca',  
'0758789922', 'ursuletulcititor@gmail.com');  
insert into depozit values(6, 'La Raftul cu Carti', 'Str. Stadionului Nr. 4, Buzau',  
'0238850714', 'lraft@gmail.com');  
insert into depozit values(7, 'Depozitul de Carti Horror', 'Str. Castelului Nr. 14,  
Bran', '0766677666', 'horrorbook@gmail.com');  
insert into depozit values(8, 'Books4ALL', 'Str. Turda Nr. 18, Bucuresti',  
'0217775678', 'books4all@yahoo.com');  
insert into depozit values(9, 'RoBooksBuc', 'Bvd. Magheru Nr. 284, Bl. 6, Et. 3,  
Bucuresti', '0758812877', 'robb@yahoo.com');  
insert into depozit values(10, 'Dream and Read', 'Str. Parcului Nr. 2, Tg. Jiu',  
'0758777899', 'dreamandread2000@yahoo.com');  
  
insert into carte values(1,2,'Dracula','Horror');  
insert into carte values(2,1,'Portretul lui Dorian Grey',null);  
insert into carte values(3,4,'Un Proiect Sangeros','Horror');  
insert into carte values(4,3,'Ferma Animalelor', null);  
insert into carte values(5,3,'Umbra Vantului','Thriller');  
insert into carte values(6,10,'Biblia Pierduta','Thriller');  
insert into carte values(7,8,'Sub Un Cer Sangeriu','Fictiune Istorica');  
insert into carte values(8,6,'Transparenta','Fictiune');  
insert into carte values(9,9,'6 Povesti cu Draci','Povestiri');  
insert into carte values(10,10,'Inferno','Thriller');  
insert into carte values(11,11,'Bohemian Rhapsody','Biografie');  
insert into carte values(12,7,'Gerrard: My Autobiography','Biografie');  
  
insert into recenzie values(1,2,7,'Recomand',to_date('20/05/2021',  
'DD/MM/YYYY'),4);  
insert into recenzie values(2,2,8,'Recomand!',to_date('20/05/2021',  
'DD/MM/YYYY'),4);  
insert into recenzie values(3,2,11,'Foarte buna cartea!',to_date('20/05/2021',  
'DD/MM/YYYY'),5);  
insert into recenzie values(4,8,12,'Cea mai buna biografie citita  
vreodata!',to_date('16/05/2021', 'DD/MM/YYYY'),5);  
insert into recenzie values(5,3,5,'Inca nu pot dormi din cauza  
cartii...',to_date('05/05/2021', 'DD/MM/YYYY'),3);
```

```
insert into recenzie values(6,8,10,'Nu este mai buna decat Codul lui DaVinci, dar este decenta.',to_date('20/05/2021', 'DD/MM/YYYY'),4);
```

```
insert into recenzie values(7,4,2,'Nu recomand sa o cititi!',to_date('28/04/2021', 'DD/MM/YYYY'),2);
```

```
insert into recenzie values(8,1,10,null,to_date('05/02/2021', 'DD/MM/YYYY'),3);
```

```
insert into recenzie values(9,10,1,null,to_date('22/05/2021', 'DD/MM/YYYY'),5);
```

```
insert into recenzie values(10,5,6,'Recomand',to_date('07/03/2021', 'DD/MM/YYYY'),4);
```

```
insert into comanda values(1,1,to_date('07/03/2021', 'DD/MM/YYYY'));
```

```
insert into comanda values(2,7,to_date('07/02/2021', 'DD/MM/YYYY'));
```

```
insert into comanda values(3,4,to_date('05/03/2020', 'DD/MM/YYYY'));
```

```
insert into comanda values(4,6,to_date('18/04/2019', 'DD/MM/YYYY'));
```

```
insert into comanda values(5,10,to_date('11/05/2021', 'DD/MM/YYYY'));
```

```
insert into comanda values(6,8,to_date('11/05/2021', 'DD/MM/YYYY'));
```

```
insert into comanda values(7,2,to_date('29/06/2020', 'DD/MM/YYYY'));
```

```
insert into comanda values(8,6,to_date('18/02/2021', 'DD/MM/YYYY'));
```

```
insert into comanda values(9,2,to_date('19/02/2021', 'DD/MM/YYYY'));
```

```
insert into comanda values(10,5,to_date('15/01/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(4,1,to_date('08/03/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(4,2,to_date('07/02/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(4,3,to_date('05/03/2020', 'DD/MM/YYYY'));
```

```
insert into preluare values(6,4,to_date('18/04/2019', 'DD/MM/YYYY'));
```

```
insert into preluare values(5,5,to_date('12/05/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(5,6,to_date('12/05/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(4,7,to_date('29/06/2020', 'DD/MM/YYYY'));
```

```
insert into preluare values(5,8,to_date('19/02/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(6,9,to_date('19/02/2021', 'DD/MM/YYYY'));
```

```
insert into preluare values(6,10,to_date('16/01/2021', 'DD/MM/YYYY'));
```

```
insert into creatie values(1,8);
```

```
insert into creatie values(2,9);
```

```
insert into creatie values(3,6);
```

```
insert into creatie values(4,7);
```

```
insert into creatie values(5,4);
```

```
insert into creatie values(6,2);
```

```
insert into creatie values(7,5);
```

```
insert into creatie values(8,3);
```

```
insert into creatie values(9,2);
```

```
insert into creatie values(10,1);
insert into creatie values(11,11);
insert into creatie values(12,10);

insert into plata values(1,1,1,null,'RON',to_date('08/03/2021',
'DD/MM/YYYY'),'cash');
insert into plata values(2,2,1,null,'RON',to_date('07/02/2021',
'DD/MM/YYYY'),'card');
insert into plata values(3,3,1,'Alexandrescu','RON',to_date('05/03/2020',
'DD/MM/YYYY'),'cash');
insert into plata values(4,4,1,null,'RON',to_date('18/04/2019',
'DD/MM/YYYY'),'cash');
insert into plata values(5,5,1,null,'RON',to_date('12/05/2021',
'DD/MM/YYYY'),'card');
insert into plata values(6,6,1,null,'RON',to_date('12/05/2021',
'DD/MM/YYYY'),'cec');
insert into plata values(7,7,1,null,'RON',to_date('29/06/2020',
'DD/MM/YYYY'),'cash');
insert into plata values(8,8,1,'Mihai','USD',to_date('20/02/2021',
'DD/MM/YYYY'),'card');
insert into plata values(9,9,1,null,'RON',to_date('19/02/2021',
'DD/MM/YYYY'),'cash');
insert into plata values(10,10,1,'Ghitu','EUR',to_date('18/01/2021',
'DD/MM/YYYY'),'cash');

insert into exemplar values(1,2,1,5,2007,8.50,'Buna');
insert into exemplar values(2,12,2,1,2007,80.70,'Buna');
insert into exemplar values(3,4,3,7,2008,4.50,'Proasta');
insert into exemplar values(4,4,4,4,2008,10.99,'Buna');
insert into exemplar values(5,8,5,9,2005,24.99,'Proasta');
insert into exemplar values(6,5,6,1,2001,45.00,'Foarte Buna');
insert into exemplar values(7,11,7,6,2019,14.00,'Proasta');
insert into exemplar values(8,10,8,9,2015,25.00,null);
insert into exemplar values(9,11,9,10,2020,30.00,null);
insert into exemplar values(10,6,10,1,2015,25.00,'Proasta');
insert into exemplar values(11,5,4,7,2006,30.00,'Buna');
insert into exemplar values(12,5,8,5,2005,22.50,null);
insert into exemplar values(13,5,6,5,2008,17.08,'Proasta');
```

Oracle SQL Developer : SGBD_Project

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
 - Laborator ID
 - Laborator ID 2
 - Radi-Ioan M&I
 - SGBD_Project
 - Tables Filtered
 - Views
 - Indexes
 - Sequences
 - Functions
 - Operators
 - Queues
 - Triggers
 - Types
 - Sequences
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Database Links
 - Public Database Links
 - Directories
- Reports

Welcome Page : SGBD_Project

Worksheet : Query Builder

Script Output : Query Result : Task completed in 0.932 seconds

```
1 row inserted.

1 row inserted.
```

Messages - Log

Type here to search

4°C Partly cloudy 1753 ENG 31-Dec-21

Oracle SQL Developer : SGBD_Project

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
 - Laborator ID
 - Laborator ID 2
 - Radi-Ioan M&I
 - SGBD_Project
 - Tables Filtered
 - Views
 - Indexes
 - Sequences
 - Functions
 - Operators
 - Queues
 - Triggers
 - Types
 - Sequences
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Database Links
 - Public Database Links
 - Directories
- Reports

Welcome Page : SGBD_Project

Worksheet : Query Builder

```
insert into exemplar values(12,5,8,5,2005,22.50,null);
insert into exemplar values(13,5,6,5,2008,17.08,'Proasta');

-----VERIFICARE TABELLE-----

select *
from client;

select *
from angajat;

select*
from editura;
```

Script Output : All Rows Petched: 10 in 0.038 seconds

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	TELEFON_CLIENT	ADRESA_CLIENT	CNP_CLIENT	EMAIL_CLIENT	DATA_INREGISTR
1	I Popescu	Ioan	0723595269	Str. Lalelelor 108, Ta. Jiu	5010304678823	iopop@gmail.com	23-JUN-2
2	Radu	Ioana	0723293170	Str. Florilor 2, Petrosani	601061672723	ioanarl@gmail.com	24-JUN-2
3	Andrei	Alexandra	0723293170	Str. Florilor 2, Petrosani	611061672710	andreea123@gmail.com	30-JUN-2
4	dBanul	Radu	0769124766	Bvd. Libertatii 32, Craiova	514010777717	adubu@gmail.com	25-FEB-2
5	5Aionesei	Andrei	0723666666	Str. Diamantelor 8, Botosani	502030402012	aa@gmail.com	06-JUN-1
6	6Alexandrescu	Paula	0723868646	Str. Bibliotecii 3, Ta. Jiu	600303744150	paulaalexandrescu2000@yahoo.com	12-DEC-1
7	7Fulea	Andreea	0723996996	Str. Turda 44, Urziceni	605020149374	andre123@gmail.com	16-AUG-2
8	8Manea	Paul	0723116256	Str. Sisesti 24, Bucuresti	511020375780	rmane@gmail.com	11-SEP-1
9	9Sorin	Maria	0723099056	Str. Soarelui 88, Galati	(null)	(null)	30-AUG-2
10	10 Andrei	Adriana	0723887723	(null)	6130302737214	aand@gmail.com	04-FEB-2

Messages - Log

Type here to search

4°C Partly cloudy 1753 ENG 31-Dec-21

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator BD
 - Radiu-loan BD 2
 - Radiu-loan MHAL
 - SGBD Project
 - Tables (Filtered)
 - Views
 - Indexes
 - Packages
 - Procedures
 - Functions
 - Triggers
 - Types
 - Synonyms
 - Materialized Views
 - Materialized View Logs
 - Public Synonyms
 - Public Column Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimexTen Reports
- User Defined Reports

Worksheet | Query Builder

```
insert into exemplar values(12,5,8,5,2005,22.50,null);
insert into exemplar values(13,5,6,5,2008,17.00,'Proasta');

-----VERIFICARE TEBELE-----

select *
from client;

select *
from angajat;

select*
from editura;
```

Script Output | SQL | All Rows Fetched: 10 in 0.014 seconds

ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	TELEFON_ANGAJAT	ADRESA_ANGAJAT	CNP_ANGAJAT	EMAIL_ANGAJAT
1	Mihai Ionescu	Carina	0723237924	(null)	(null)	mcarina@gmail.com
2	Mihai Ionescu	Dorina	0723337844	(null)	6010302967214	mdorina@gmail.com
3	Georgescu	Daria	0723852467	Str. Sisesti 222, Bucuresti	(null)	dariageo@gmail.com
4	Vasilescu	Simona	0723295844	Str. Garii 22, Buftea	601030234433	vasimona@gmail.com
5	Popescu	Mihaila	0723196222	Str. Zorilor 11, Buftea	(null)	pomnaha@gmail.com
6	Pavaloiu	Anca	0723894370	Sbl. Mihai Cristian Oancea, Bl. 6, Ap.11, Tg. Jiu	6000308967312	pavaloiuanca@gmail.com
7	Dumitrascu	Iulia	0723742196	Str. Stadionului 123, Petrosani	(null)	iuliasi@gmail.com
8	Mosoi	Andrei	0724592756	Sos. Grivitei 128C, Bucuresti	(null)	andreimosoi@gmail.com
9	Gherche	Vladimir	0723237345	Str. Garii 17, Brasov	5010203050401	vlag@gmail.com
10	Ceparu	Andrei	0745063364	(null)	(null)	cepi2001@gmail.com

Messages - Log

Type here to search

4°C Partly cloudy 17:54 ENG 31-Dec-21

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator BD
 - Radiu-loan BD 2
 - Radiu-loan MHAL
 - SGBD Project
 - Tables (Filtered)
 - Views
 - Indexes
 - Packages
 - Procedures
 - Functions
 - Triggers
 - Types
 - Synonyms
 - Materialized Views
 - Materialized View Logs
 - Public Synonyms
 - Public Column Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimexTen Reports
- User Defined Reports

Worksheet | Query Builder

```
select *
from angajat;

select*
from editura;

select*
from autor;

select*
from depozit;

select*
from carte;
```

Script Output | SQL | All Rows Fetched: 12 in 0.007 seconds

ID_EDITURA	NUME_EDITURA	ADRESA_EDITURA	TELEFON_EDITURA	EMAIL_EDITURA
1	Editura Univers	Str. Dionisie Lupu Nr. 65, Bucuresti	0758770777	(null)
2	2Editura Adavarul Holding	(null)	0214077650	cartea@devarul.ro
3	3Polirom	Bvd. Carol I nr. 4, Iasi	0232214100	office@polirom.ro
4	4Editura ART	Splaiul Independentei nr. 315-319, Bucuresti	0744300870	comenzi@editura-art.ro
5	5LEDA	Calea Plevnei nr. 145, Bucuresti	0213198822	vanzari@edituracorint.ro
6	6Humanitas	Pta. Presei Libere nr. 1, Bucuresti	0214088350	vanzari@lphumanitas.ro
7	7Bantant Books	61-63 Uxbridge Road, London, UK	(null)	bdpublicity@randomhouse.com
8	8Editura Trei	O.P. 16, Ghiseul 1, Bucuresti	0213006090	comenzi@editura3.ro
9	9Editura Litera	O.P. 53, Bucuresti	0213196390	comenzi@litera.ro
10	10Editura RAO	Str. Baragaului, nr. 9-11, Bucuresti	(null)	office@rao.ro
11	11Nemira	Str. Iani Buzoianu 14, Bucuresti	0721747464	office@nemira.ro
12	Weidenfeld and Nicolson	Carmelite House 50, London, UK	(null)	(null)

Messages - Log

Type here to search

4°C Partly cloudy 17:54 ENG 31-Dec-21

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Radu-Ioan BD 2
 - Radu-Ioan MSQL
 - SGBD Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadrops
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - External Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimexTen Reports
- User Defined Reports

Worksheet - Query Builder

```
select *
from angajat;

select*
from editura;

select*
from autor;

select*
from depozit;

select*
from carte;
```

Script Output | Query Result | All Row Fetches: 11 in 0.007 seconds

ID_AUTOR	NAME_AUTOR	PRAEFALME_AUTOR	DATA_NASTERII
1	I Brown	Dan	22-JUN-64
2	B Bergler	Igor	21-SEP-70
3	Vancu Radu		13-JUL-78
4	Zafon Carlos Ruiz		25-SEP-64
5	Sullivan Mark		28-JUN-58
6	Burnet Graeme Macrae	O1-JAN-67	
7	Orwell George		25-JUN-03
8	Stocker Bram		08-NOV-47
9	Wilde Oscar		16-OCT-54
10	Gerrard Steven		30-MAY-80
11	Jones Lesley-Ann	(null)	

Messages - Log messages Logging mode: automatic

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

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Radu-Ioan BD 2
 - Radu-Ioan MSQL
 - SGBD Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadrops
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - External Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimexTen Reports
- User Defined Reports

Worksheet - Query Builder

```
select *
from angajat;

select*
from editura;

select*
from autor;

select*
from depozit;

select*
from carte;
```

Script Output | Query Result | All Row Fetches: 10 in 0.008 seconds

ID_DEPOZIT	NAME_DEPOZIT	ADRESA_DEPOZIT	TELEFON_DEPOZIT	EMAIL_DEPOZIT
1	1Depozitul de Carte	Str. Fantanii Nr. 65, Chiajna	0756777077	depozituldecarte@yahoo.com
2	2Depozitul de Carte Straină	Ptr. Romană Nr. 9, Bucuresti	021370977	sitka@sitka.ro
3	3Depozitul de Carti Vechi	Str. Cluj Nr. 68, Bucuresti	0723667877	carteaveche@gmail.com
4	4Depozitul Cartea Muntelui	Str. Postei Nr. 4, Brasov	0758774513	carteanumteluibv@yahoo.com
5	5Depozitul Ursuletul Cititor	Str. Ursului Nr. 1, Rancă	0758789922	ursuletulcititor@gmail.com
6	6La Raftul cu Carti	Str. Stadionului Nr. 4, Buzau	0238850714	larraft@gmail.com
7	7Depozitul de Carti Horror	Str. Castelului Nr. 14, Bran	0766677666	horrorbook@gmail.com
8	8Books4ALL	Str. Turda Nr. 18, Bucuresti	0217775678	books4all@yahoo.com
9	9RoBooksBuc	Bvd. Magheru Nr. 284, Bl. 6, Et. 3, Bucuresti	0758812877	robb@yahoo.com
10	10Dream and Read	Str. Parcului Nr. 2, Tg. Jiu	0758777899	dramandread2000@yahoo.com

Messages - Log messages Logging mode: automatic

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

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Radu-loan BD 2
 - Radu-loan MySQL
 - SGBD Project
 - Tables (Filtered)
 - Views
 - Materialized Views
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Sequences
 - Materialized Views Logs
 - Synonyms
 - Public Synonyms
 - External Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analyze View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimetTen Reports
- User Defined Reports

Worksheet | Query Builder | Welcome Page | SGBD_Project | Script Output | Query Result | SQL | All Rows Fetched: 12 in 0.007 seconds

```

select*
from depozit;

select*
from carte;

select*
from recenzie;

select*
from comanda;

select*
from preluare;
  
```

Messages - Log messages Logging major - Database

ID_CARTE	ID_JEZUITA	TITLU_CARTE	CATEGORIE_CARTE
1	1	2 Dracula	Horror
2	2	1Portretul lui Dorian Grey	(null)
3	3	4Un Proiect Sanderos	Horror
4	4	3Ferma Animalelor	(null)
5	5	3Umbra Vantului	Thriller
6	6	10Biblia Pierduta	Thriller
7	7	8Sub Un Cer Sangeriu	Fictiune Istorica
8	8	6Transparenta	Fictiune
9	9	96 Povesti cu Draci	Povestiri
10	10	10Inferno	Thriller
11	11	11Bohemian Rhapsody	Biografie
12	12	7Gerrard: My Autobiography	Biografie

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

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Radu-loan BD 2
 - Radu-loan MySQL
 - SGBD Project
 - Tables (Filtered)
 - Views
 - Materialized Views
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Sequences
 - Materialized Views Logs
 - Synonyms
 - Public Synonyms
 - External Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analyze View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimetTen Reports
- User Defined Reports

Worksheet | Query Builder | Welcome Page | SGBD_Project | Script Output | Query Result | SQL | All Rows Fetched: 10 in 0.008 seconds

```

select*
from depozit;

select*
from carte;

select*
from recenzie;

select*
from comanda;

select*
from preluare;
  
```

Messages - Log messages Logging major - Database

ID_RECENZIE	ID_CLIENT	ID_CARTE	CONTINUT_RECENZIE	DATA_RECENZIE	NUMAR_STELA
1	1	2	7Recomand!	20-MAY-21	4
2	2	2	8Recomand!	20-MAY-21	4
3	3	2	11Foarte buna carte!	20-MAY-21	5
4	4	8	12Cea mai buna biografie citita vreodata!	16-MAY-21	5
5	5	3	15Inca nu pot dormi din cauza cartii.	05-MAY-21	3
6	6	8	10Nu este mai bună decat Codul lui DaVinci, dar este decentă.	20-MAY-21	4
7	7	4	2Nu recomand sa o cititi!	28-APR-21	2
8	8	1	10 (null)	05-FEB-21	3
9	9	10	1 (null)	22-MAY-21	5
10	10	5	6Recomand	07-MAR-21	4

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

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Radu-Ioan BD 2
 - Radu-Ioan MRel
 - SGBD_Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadecors
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Public Links
 - Public Database Links
 - Directories
- Reports

Worksheet | Query Builder

```

select*
from depozit;

select*
from carte;

select*
from recenzie;

select*
from comanda;

select*
from preluare;
  
```

Script Output | SQL | All Rows Fetched: 10 in 0.006 seconds

ID_COMMANDA	ID_CLIENT	DATA_COMMANDA
1	1	107-MAR-21
2	2	107-FEB-21
3	3	405-MAR-20
4	4	618-APR-19
5	5	1011-MAY-21
6	6	811-MAY-21
7	7	229-JUN-20
8	8	618-FEB-21
9	9	219-FEB-21
10	10	515-JAN-21

Messages - Log

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

Type here to search

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Radu-Ioan BD 2
 - Radu-Ioan MRel
 - SGBD_Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadecors
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Public Links
 - Public Database Links
 - Directories
- Reports

Worksheet | Query Builder

```

select*
from preluare;

select*
from creatie;

select*
from plata;

select*
from exemplar;
  
```

Script Output | SQL | All Rows Fetched: 10 in 0.006 seconds

ID_ANGAGIAT	ID_COMMANDA	DATA_PLAGARE
1	4	109-MAR-21
2	4	207-FBD-21
3	4	305-MAR-20
4	6	418-APR-19
5	5	512-MAY-21
6	5	612-MAY-21
7	4	729-JUN-20
8	5	819-FEB-21
9	6	919-FEB-21
10	6	1016-JAN-21

Messages - Log

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

Type here to search

Oracle SQL Developer : SGBD_Project

```

select*
from prelure;

select*
from creatie;

select*
from plata;

select*
from exemplar;

```

Script Output x Query Result x

ID_CREATIE	ID_AUTOR
1	8
2	9
3	6
4	7
5	4
6	2
7	5
8	3
9	2
10	1
11	11
12	10

Messages - Log

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

Type here to search

1 Line 299 Column 14 | Insert | Modified | Windows: 0

4°C Partly cloudy 17:56 ENG 31-Dec-21

Oracle SQL Developer : SGBD_Project

```

select*
from prelure;

select*
from creatie;

select*
from plata;

select*
from exemplar;

```

Script Output x Query Result x

ID_PLATA	ID_COMANDA	ID_ANGAJAT	NUME_PLATOR	VALUTA	DATA_EFFECTUARE_PLATA	TIP_PLATA
1	1	1	1(null)	RON	08-MAR-21	cash
2	2	2	1(null)	RON	07-FEB-21	card
3	3	3	1Alexandrescu	RON	05-MAR-20	cash
4	4	4	1(null)	RON	18-APR-19	cash
5	5	5	1(null)	RON	12-MAY-21	card
6	6	6	1(null)	RON	12-MAY-21	cec
7	7	7	1(null)	RON	29-JUN-20	cash
8	8	8	1Mihai	USD	20-FEB-21	card
9	9	9	1(null)	RON	19-FEB-21	cash
10	10	10	1Ghitu	EUR	18-JAN-21	cash

Messages - Log

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

Type here to search

1 Line 302 Column 12 | Insert | Modified | Windows: 0

4°C Partly cloudy 17:56 ENG 31-Dec-21

```

select*
from preluate;

select*
from creatie;

select*
from plata;

select*
from exemplar;

```

ID_EXEMPLAR	ID_CARTE	ID_COMANDA	ID_DEPOZIT	AN_APARITIE_EXEMPLAR	PRET_EXEMPLAR	STARE_EXEMPLAR
1	1	2	1	5	2007	8,5 Buna
2	2	12	2	1	2007	80,7 Buna
3	3	4	3	7	2008	4,5 Proasta
4	4	4	4	4	2008	10,99 Buna
5	5	8	5	9	2005	24,99 Proasta
6	6	5	6	1	2001	45 Foarte Buna
7	7	11	7	6	2019	14 Proasta
8	8	10	8	9	2015	25 (null)
9	9	11	9	10	2020	30 (null)
10	10	6	10	1	2015	25 Proasta
11	11	5	4	7	2006	30 Buna
12	12	5	8	5	2005	22,5 (null)
13	13	5	6	5	2008	17,08 Proasta

6. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat** care să utilizeze două tipuri de colecție studiate. Apelați subprogramul.

Enunt: Folosind un subprogram stocat cu un an primit ca parametru, afișați cu ajutorul unui parametru de tip colectie numele, prenumele și e-mail-ul clientilor care s-au înregistrat înainte de acel an și numele, prenumele și e-mail-ul personalului care a fost angajat înainte de acel an (inclusiv acel an). În cazul în care nu există clienti și angajați, se va trata cu o eroare.

Rezolvare:

```
CREATE OR REPLACE PROCEDURE detalii_clienti_angajati
(an VARCHAR2)
```

AS

```
    TYPE tablou_indexat IS TABLE OF client%ROWTYPE INDEX BY
    PLS_INTEGER;
```

```
    detalii_client tablou_indexat;
```

```
    TYPE tablou_imbricat IS TABLE OF angajat%ROWTYPE;
```

```
    detalii_angajat tablou_imbricat:= tablou_imbricat();
```

```
    nr1 NUMBER;
```

```
nr2 NUMBER;

BEGIN

    SELECT null INTO nr1
    FROM client
    WHERE TO_CHAR(data_inregistrare, 'YYYY') <= an AND ROWNUM=1;

    SELECT null INTO nr2
    FROM angajat
    WHERE TO_CHAR(data_angajare, 'YYYY') <= an AND ROWNUM=1;

    SELECT * BULK COLLECT INTO detalii_client
    FROM client
    WHERE TO_CHAR(data_inregistrare, 'YYYY') <= an;

    SELECT * BULK COLLECT INTO detalii_angajat
    FROM angajat
    WHERE TO_CHAR(data_angajare, 'YYYY') <= an;

    DBMS_OUTPUT.PUT_LINE('CLIENTI: ');

    FOR i IN detalii_client.FIRST..detalii_client.LAST LOOP
        DBMS_OUTPUT.PUT_LINE (detalii_client(i).nume_client || ' ' ||
        detalii_client(i).prenume_client || ' -> ' || NVL(detalii_client(i).email_client,'Adresa
        email inexistentă'));
    END LOOP;

    DBMS_OUTPUT.PUT_LINE('ANGAJATI: ');

    FOR i IN detalii_angajat.FIRST..detalii_angajat.LAST LOOP
        DBMS_OUTPUT.PUT_LINE (detalii_angajat(i).nume_angajat || ' ' ||
        detalii_angajat(i).prenume_angajat || ' -> ' ||
        NVL(detalii_angajat(i).email_angajat,'Adresa email inexistentă'));
    END LOOP;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
```

```

    RAISE_APPLICATION_ERROR(-20000, 'Nu exista clienti sau angajati
inregistrati inainte de ' || an);
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20001, 'Alta eroare!');
END detalii_clienti_angajati;
/

```

Testare:

```

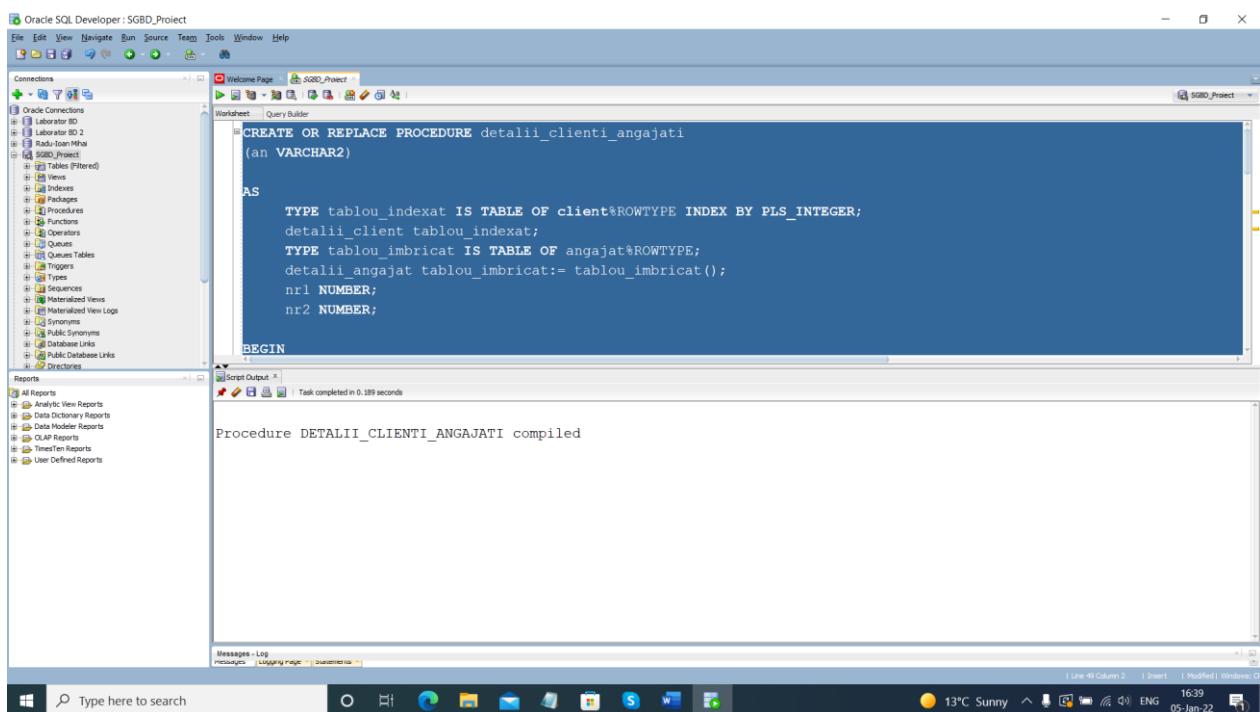
SET SERVEROUTPUT ON;
execute detalii_clienti_angajati(2021);
/

```

```

execute detalii_clienti_angajati(2001);
/

```



Session 1 (CLIENTI):

```

WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20001,'Alta eroare!');
END detalii_clienti_angajati;
/
SET SERVEROUTPUT ON;
execute detalii_clienti_angajati(2021);
/
execute detalii_clienti_angajati(2001);
/

```

Output:

```

CLIENTI:
Popescu Ioan -> ipop@gmail.com
Radu Ioana -> ioanar@gmail.com
Mihai Alexandra -> am@gmail.com
Banu Radu -> radub@gmail.com
Aionese Andrei -> aa@gmail.com
Alexandrescu Paula -> paulalexandrescu2000@yahoo.com
Fulea Andreea -> andreil23@gmail.com
Manea Paul -> pman@gmail.com
Sorin Maria -> Adresa email inexistentă
Andrei Adriana -> aand@gmail.com
Andrei B -> aand@gmail.com

```

Session 2 (ANGAJATI):

```

WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20001,'Alta eroare!');
END detalii_clienti_angajati;
/
SET SERVEROUTPUT ON;
execute detalii_clienti_angajati(2021);
/
execute detalii_clienti_angajati(2001);
/

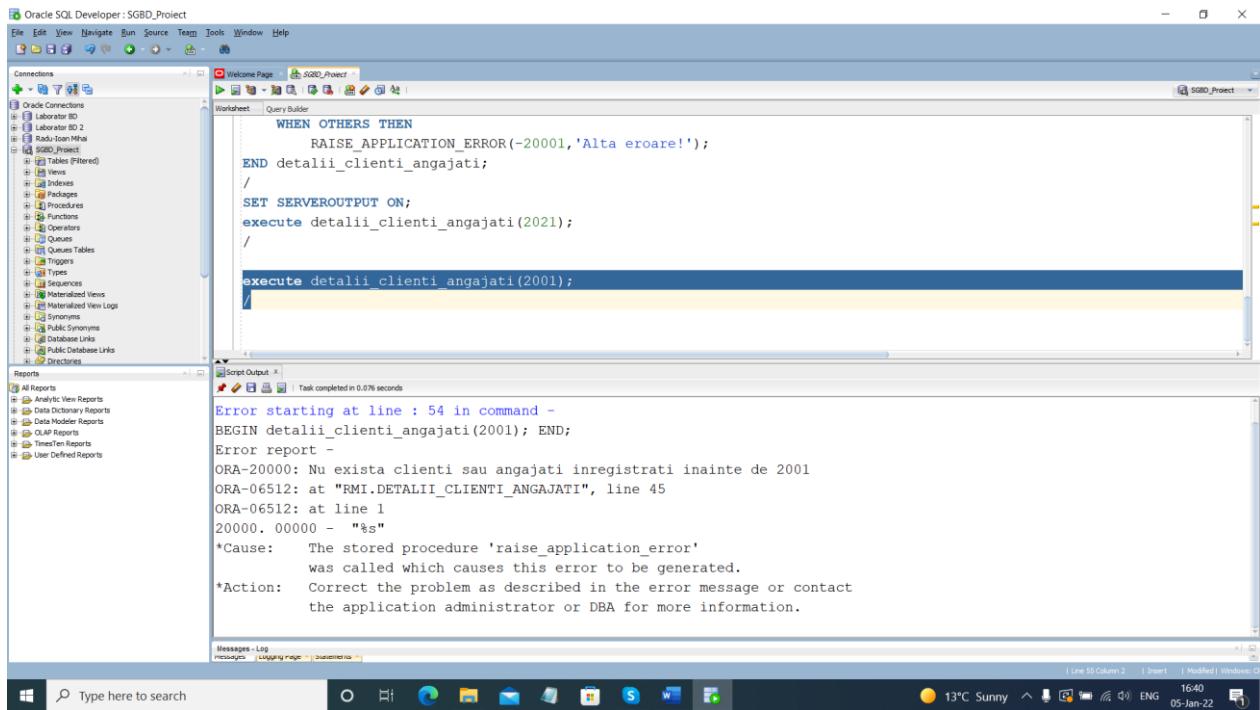
```

Output:

```

ANGAJATI:
Mihailescu Carina -> mcarina@gmail.com
Mihailescu Dorina -> mdorina@gmail.com
Georgescu Daria -> dariageo@gmail.com
Vasilescu Simona -> vasimona@gmail.com
Popescu Mihaela -> popmiha@gmail.com
Pavaloiu Anca -> pavaloiuanca@gmail.com
Dumitrascu Iulia -> iuliad@gmail.com
Mosoi Andrei -> andreimosoi@gmail.com
Gherghel Vladimir -> vlag@gmail.com
Ceparu Andrei -> cepi2001@gmail.com

```



7. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat care să utilizeze un tip de cursor studiat. Apelați subprogramul.

Enunt: Sa se afiseze id-ul celor mai cumparate 3 carti si de cate ori au fost achizitionate intr-un an dat ca parametru intr-un subprogram stocat. In caz de egalitate al numarului de exemplare, se va afisa cartea cu id-ul cel mai mic prima data.

Rezolvare:

```

CREATE OR REPLACE PROCEDURE
top3(an VARCHAR2) IS
CURSOR c IS
SELECT idc, numar FROM
(SELECT e.id_carte idc, COUNT(*) numar
FROM exemplar e
JOIN comanda co ON (co.id_comanda = e.id_comanda)
WHERE TO_CHAR(co.data_comanda,'YYYY') = an
GROUP BY id_carte
ORDER BY 2 DESC,1)

```

```

        WHERE ROWNUM<=3;
n1 NUMBER;
n2 NUMBER;
BEGIN
    SELECT idc, numar INTO n1, n2
    FROM
        (SELECT e.id_carte idc, COUNT(*) numar
        FROM exemplar e
        JOIN comanda co ON (co.id_comanda = e.id_comanda)
        WHERE TO_CHAR(co.data_comanda,'YYYY') = an
        GROUP BY id_carte
        ORDER BY 2 DESC,1)
        WHERE ROWNUM=1;
FOR i IN c LOOP
    DBMS_OUTPUT.PUT_LINE ('ID Carte: ' || i.idc || ' -> Numar comenzi in ' ||
an || ': '|| i.numar);
END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Nu exista carti vandute in ' || an);
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001,'Alta eroare!');
END top3;
/

```

Testare:

```

BEGIN
    top3(2021);
END;
/

```

```

BEGIN
    top3(2001);
END;
/

```

```

CREATE OR REPLACE PROCEDURE
    top3(an VARCHAR2) IS
    CURSOR c IS
        SELECT idc, numar FROM
            (SELECT e.id_carte idc, COUNT(*) numar
             FROM exemplar e
             JOIN comanda co ON (co.id_comanda = e.id_comanda)
             WHERE TO_CHAR(co.data_comanda,'YYYY') = an
             GROUP BY id_carte
             ORDER BY 2 DESC,1)
        WHERE ROWNUM<=3;
    nl NUMBER;

```

Procedure TOP3 compiled

```

WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20001, 'Alta eroare!');
END top3;
/

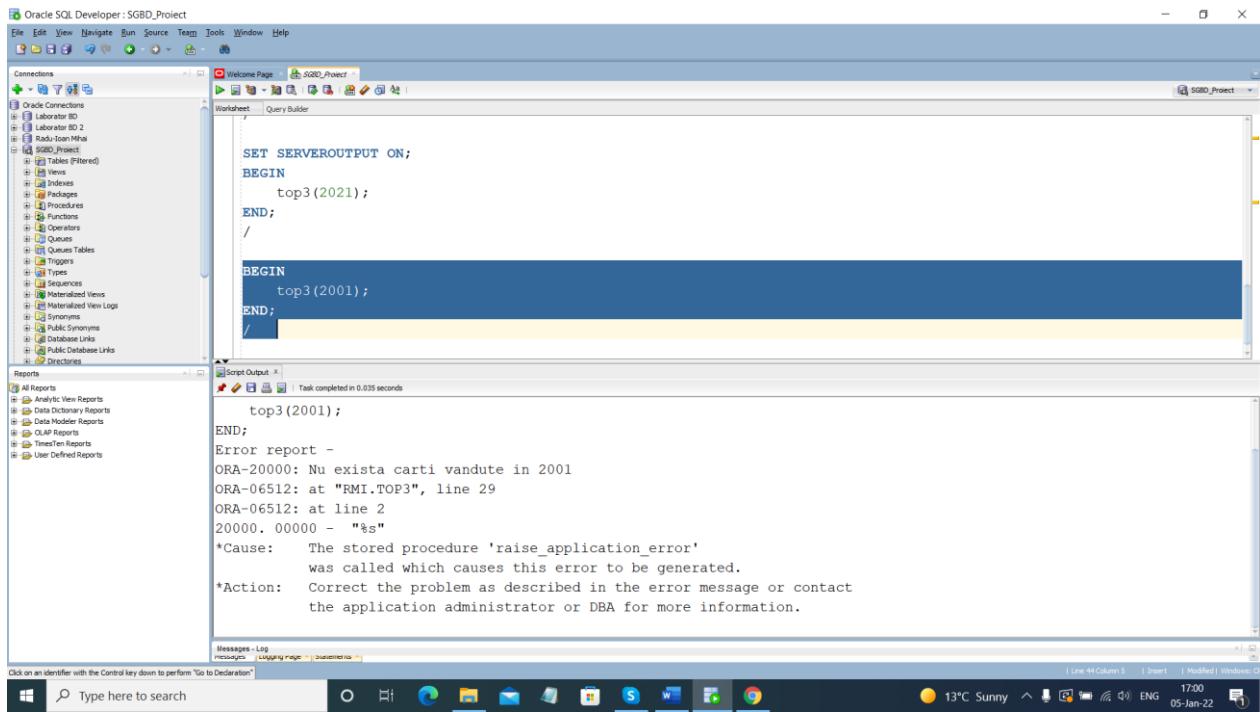

SET SERVEROUTPUT ON;
BEGIN
    top3(2021);
END;
/

```

Procedure TOP3 compiled

ID Carte: 5 -> Numar comenzi in 2021: 3
ID Carte: 2 -> Numar comenzi in 2021: 1
ID Carte: 6 -> Numar comenzi in 2021: 1

PL/SQL procedure successfully completed.



8. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite. Tratați toate excepțiile care pot apărea. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

Enunt: De Black Friday, anticariatul are o ofertă specială: se aplică o reducere **a TVA-ului (5%)** pentru fiecare carte dacă cumpără toate cartile existente de la o editură. **De asemenea, dacă prețul după reducere depășește 100 RON, se va mai face o reducere de 10%.** Dat fiind numele editurii, determinați prețul final ce trebuie platit.

Rezolvare:

```

CREATE OR REPLACE FUNCTION reducere_editura
(v_numă editura.numă_editura%TYPE DEFAULT 'Nemira')
RETURN NUMBER IS
    pret_total NUMBER(10,2);
    id_ed editura.id_editura%TYPE;
BEGIN
    SELECT id_editura
    INTO id_ed
    FROM editura
    WHERE numă_editura = v_numă;
    pret_total := sum(cantitate * pret) / count(cantitate);
    IF pret_total > 100 THEN
        pret_total := pret_total * 0.9;
    END IF;
    RETURN pret_total;
END;

```

```

FROM editura e
WHERE upper(nume_editura)=upper(v_nume);

SELECT SUM(pret_exemplar-(5/100)*pret_exemplar)
INTO pret_total
FROM editura e
JOIN carte c ON (c.id_editura=e.id_editura)
JOIN exemplar ex ON (ex.id_carte=c.id_carte)
WHERE upper(nume_editura)=upper(v_nume);

IF pret_total >= 100 THEN
    pret_total := pret_total - (10/100)*pret_total;
END IF;

RETURN pret_total;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Editura cautata nu exista.');
    WHEN TOO_MANY_ROWS THEN
        RAISE_APPLICATION_ERROR(-20001, 'Exista mai multe edituri cu acest
nume!');
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20002,'Alta eroare!');
END reducere_editura;
/

```

Testare:

```

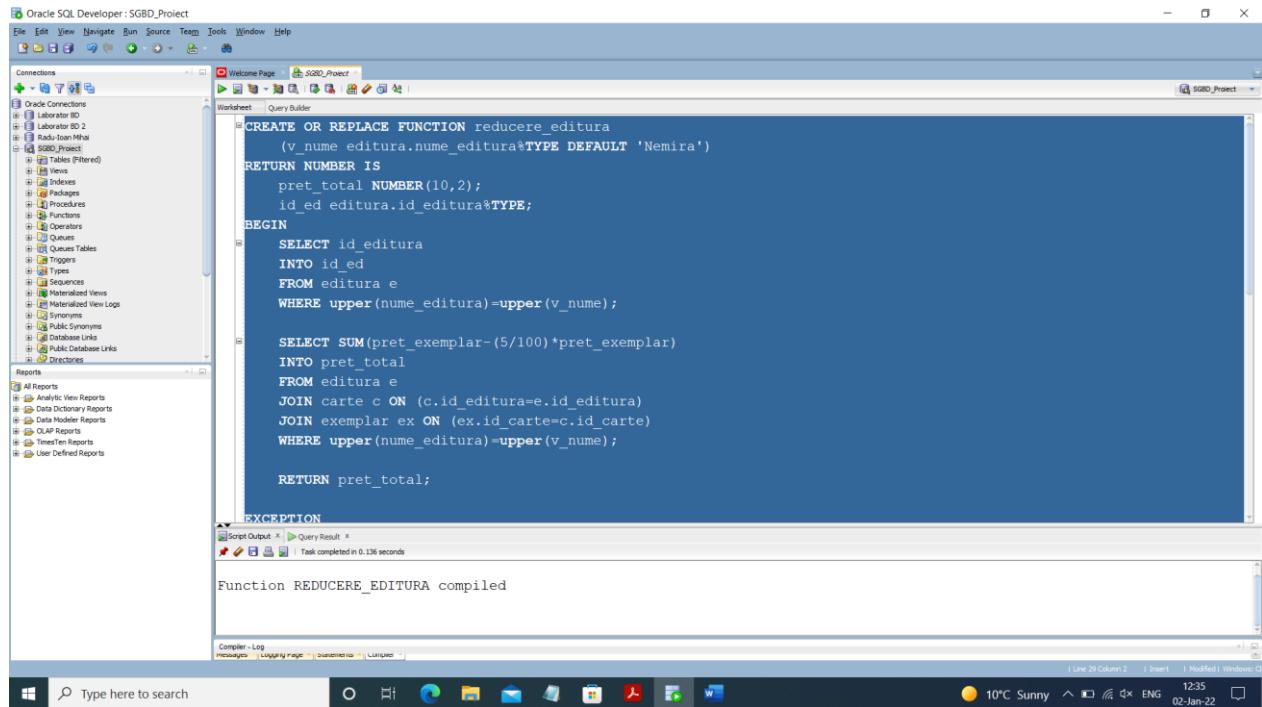
SET SERVEROUTPUT ON;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Rao'));
END;
/

BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura);
END;
/

```

```
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Editura
RAO'));
END;
/
```

```
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Polirom'));
END;
/
```



The screenshot shows the Oracle SQL Developer interface with the title bar "Oracle SQL Developer : SGBD_Project". The "Connections" sidebar lists several database connections, including "Laborator ID", "Laborator Bd 2", "Radu-Ioan Bd 2", and "SGBD Project". The "Reports" sidebar contains various report types. The "Worksheet" tab is active, displaying the following PL/SQL code:

```
--TESTARE
SET SERVEROUTPUT ON;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Rao'));
END;
/

```

Below the code, the "Script Output" and "Query Result" panes are visible. The "Script Output" pane shows the error message:

```
Error starting at line : 34 in command -
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Rao'));
END;
Error report -
ORA-20000: Editura cautata nu exista.
ORA-06512: at "RMI.REDUCE>EditURA", line 23
ORA-06512: at line 2
20000. 00000 - "%s"
*Cause: The stored procedure 'raise_application_error'
was called which causes this error to be generated.
*Action: Correct the problem as described in the error message or contact
the application administrator or DBA for more information.
```

The status bar at the bottom right indicates "10°C Sunny 12:36 02-Jan-22".

This screenshot shows the same Oracle SQL Developer environment. The "Worksheet" tab is active, displaying the same PL/SQL code as the previous screenshot:

```
--TESTARE
SET SERVEROUTPUT ON;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Rao'));
END;
/

```

Below the code, the "Script Output" and "Query Result" panes are visible. The "Query Result" pane shows the output of the procedure:

```
Pretul final este 41.8

PL/SQL procedure successfully completed.
```

The status bar at the bottom right indicates "10°C Sunny 12:36 02-Jan-22".

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Labo_Rao ID 2
 - Radu-Ioan Rao
 - SGBD_Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadrops
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Public Database Links
 - Public Database Links
 - Directories
- Reports

Worksheet (Query Builder)

```

SET SERVEROUTPUT ON;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Rao'));
END;
/

BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura);
END;
/

```

Script Output x | Query Result x

Pretul final este 47.5

PL/SQL procedure successfully completed.

Compiler - Log

Type here to search

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Labo_Rao ID 2
 - Radu-Ioan Rao
 - SGBD_Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadrops
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Public Database Links
 - Public Database Links
 - Directories
- Reports

Worksheet

```

END reducere_editura;
/


insert into editura values(52, 'LEDA', 'Calea Plevnei nr. 145, Bucuresti', '0213198822', 'vanzari@edituracorint.ro');

select * from editura;

BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('LEDA'));
END;
/

```

Error starting at line : 36 in command -

```

BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('LEDA'));
END;
Error report -
ORA-20001: Există mai multe edituri cu acest nume!
ORA-06512: at "RMI.REDUCERE_EDITURA", line 25
ORA-06512: at line 2

```

Messages - Log

Type here to search

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' sidebar lists various database objects under 'SGBD_Project'. In the center, the 'Worksheet' tab contains the following PL/SQL code:

```

BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Editura RAO'));
END;
/

```

Below this, another block of code is partially visible:

```

BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este'|| reducere_editura('Polirrom'));
END;
/

```

To the right of the code, the 'Script Output' window displays the results of the execution:

```

Task completed in 0.068 seconds
Pretul final este 111.21
PL/SQL procedure successfully completed.

```

The status bar at the bottom shows the file path 'Saved: C:\Users\Radu Mihai\SGBD_Project.sql' and the system date/time '28-Aug-22 00:53'.

9. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat de tip procedură care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO_DATA_FOUND și TOO_MANY_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

Enunt: Pentru numele de familie al unui client transmis ca parametru unei proceduri afisati numele complet al clientului, data fiecarei comenzi, angajatul care a preluat comanda si cum a fost platita aceasta. **Daca exista mai multe comenzi efectuate de acel client, se vor afisa in ordinea descrescatoare a datei comenzi. In cazul in care comenzile sunt efectuate in aceeasi zi, se vor folosi ca criteriu de departajare, in aceasta ordine, data descrescatoare a preluarii si numele angajatului care a preluat comanda.**

Rezolvare:

CREATE OR REPLACE PROCEDURE

```

p9(num_c client.nume_client%TYPE)
IS ccod client.id_client%TYPE;
CURSOR c1 IS
  SELECT c.nume_client ||' '|| c.prenume_client as NumeClient, data_comanda,
nume_angajat ||' '|| prenume_angajat as NumeAngajat, tip_plata
  FROM client c
  JOIN comanda co ON (c.id_client=co.id_client)
  JOIN preluare p ON (p.id_comanda=co.id_comanda)
  JOIN angajat a ON (p.id_angajat=a.id_angajat)
  JOIN plata pl ON (pl.id_comanda=co.id_comanda)
  WHERE UPPER(c.nume_client)=UPPER(num_c)
  ORDER BY co.data_comanda DESC, p.data_preluare DESC, a.nume_angajat;
BEGIN
  SELECT id_client
  INTO ccod
  FROM client c
  WHERE UPPER(c.nume_client)=UPPER(num_c);
  DBMS_OUTPUT.PUT_LINE('Clientul cu id '||ccod||': ');
  DBMS_OUTPUT.NEW_LINE();
  DBMS_OUTPUT.PUT_LINE('-----');
  FOR i in c1 LOOP
    DBMS_OUTPUT.PUT_LINE('Nume complet: '||i.NumeClient);
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('Data comanda: '||i.data_comanda);
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('Angajat: '||i.NumeAngajat);
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('Tip Plata: '||i.tip_plata);
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('-----');
  END LOOP;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    RAISE_APPLICATION_ERROR(-20000, 'Clientul cu numele respectiv nu
exista.');
  WHEN TOO_MANY_ROWS THEN

```

```

    RAISE_APPLICATION_ERROR(-20001, 'Mai multi clienti cu acelasi
nume');
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20002, 'Alta eroare!');
END p9;
/

```

Testare:

```

SET SERVEROUTPUT ON;
BEGIN
p9('xcv');
END;

```

```

BEGIN
p9('Andrei');
END;

```

```

BEGIN
p9('Radu');
END;

```

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays the project structure under 'SGBD_Project' with various database objects like tables, views, packages, and procedures. The central workspace contains the following PL/SQL code:

```

CREATE OR REPLACE PROCEDURE
comenzi(numec_c client.numec_client%TYPE)
IS ccod client.id_client%TYPE;
CURSOR cl IS
    SELECT c.numec_client ||' '|| c.prenume_client AS NumecClient, data_comanda, nume_angajat ||' '|| prenume_angajat
    FROM client c
    JOIN comanda co ON (c.id_client=co.id_client)
    JOIN preluare p ON (p.id_comanda=co.id_comanda)
    JOIN angajat a ON (p.id_angajat=a.id_angajat)
    JOIN plata pl ON (pl.id_comanda=co.id_comanda)
    WHERE UPPER(c.numec_client)=UPPER(numec_c);
BEGIN

```

Below the code, the 'Script Output' window shows the message: "Procedure COMENZI compiled". The bottom status bar indicates the system is 13°C, sunny, and the date is 05-Jan-22.

```

WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20002,'Alta eroare!');
END comenzi;
/


BEGIN
comenzi('xcv');
END;

BEGIN
comenzi('Andrei');
END;

```

Script Output x

```

comenzi('xcv');
END;
Error report -
ORA-20000: Clientul cu numele respectiv nu exista.
ORA-06512: at "RMI.COMENZI", line 35
ORA-06512: at line 2
20000. 00000 - "%s"
*Cause: The stored procedure 'raise_application_error'
        was called which causes this error to be generated.
*Action: Correct the problem as described in the error message or contact
        the application administrator or DBA for more information.

```

Messages - Log

Type here to search

13°C Sunny 18:00 05-Jan-22

```

SET SERVEROUTPUT ON;
BEGIN
p9('xcv');
END;

BEGIN
p9('Andrei');
END;

BEGIN
p9('Radu');
END;

```

Script Output x

```

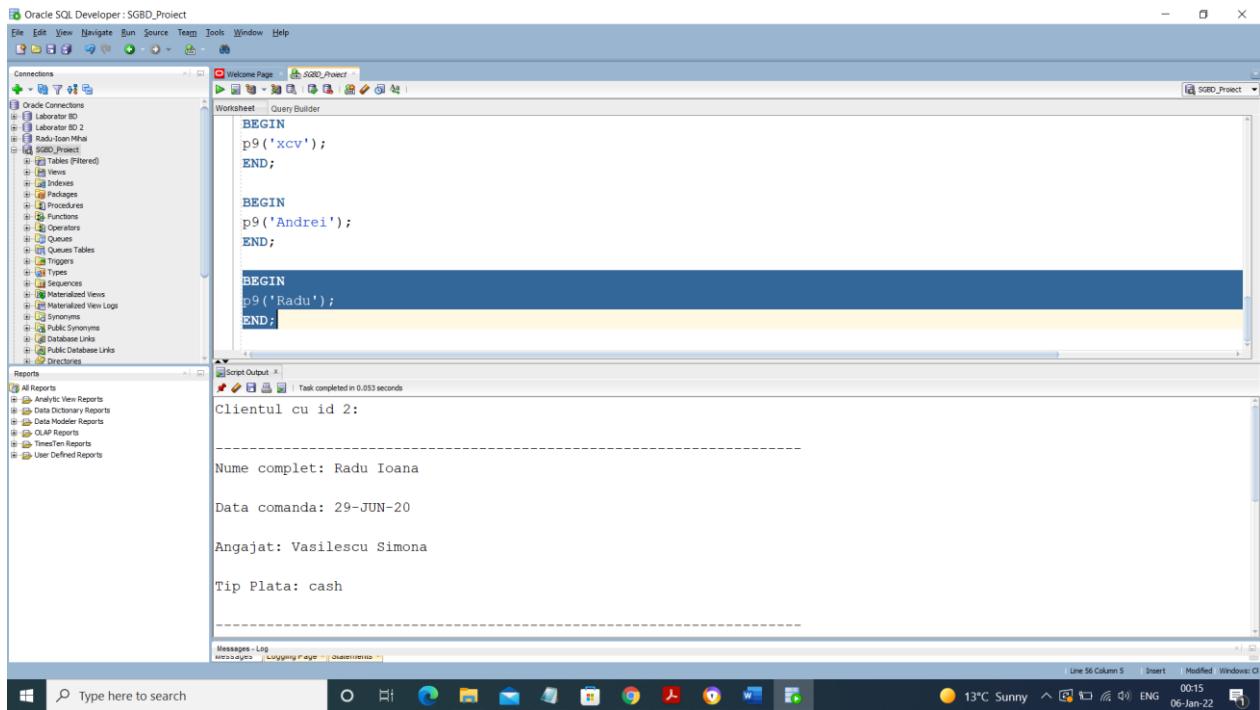
Error starting at line : 50 in command -
BEGIN
p9('Andrei');
END;
Error report -
ORA-20001: Mai multi clienti cu acelasi nume
ORA-06512: at "RMI.P9", line 37
ORA-06512: at line 2

```

Messages - Log

Type here to search

13°C 00:12 06-Jan-22



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

Enunt: Definiti un trigger pentru a permite inserarea in baza de data si updatearea acestieia doar in timpul programului de lucru: Luni – Sambata, de la ora 8:00 pana la ora 21:00.

Rezolvare:

```

CREATE OR REPLACE TRIGGER insert_invalid
BEFORE INSERT OR UPDATE ON client
BEGIN
  IF (TO_CHAR(SYSDATE,'D') = 1) THEN
    RAISE_APPLICATION_ERROR(-20000,'Tabelul nu poate fi actualizat.
Incercati din nou Luni!');
  ELSIF (TO_CHAR(SYSDATE,'HH24') NOT BETWEEN 8 and 20) THEN
    RAISE_APPLICATION_ERROR(-20001,'Tabelul nu poate fi actualizat decat
in timpul programului!');
  END IF;
END;

```

/

Testare:

```
insert into client values(500, 'Radu', 'Mihai', '0723293122','Str. Fabricii 2, Braila',
'6010611672883','rmi_m@gmail.com', sysdate+5, 'rmih');
```

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays database connections and objects. The central workspace contains a query builder window with the following code:

```
CREATE OR REPLACE TRIGGER insert_invalid
BEFORE INSERT OR UPDATE ON client
BEGIN
    IF (TO_CHAR(SYSDATE,'D') = 1) THEN
        RAISE_APPLICATION_ERROR(-20000,'Tabelul nu poate fi actualizat. Incercati din nou Luni!');
    ELSIF (TO_CHAR(SYSDATE,'HH24') NOT BETWEEN 8 and 20) THEN
        RAISE_APPLICATION_ERROR(-20000,'Tabelul nu poate fi actualizat decat in timpul programului!');
    END IF;
END;
```

Below this, there is an 'insert' statement:

```
insert into client values(500, 'Radu', 'Mihai', '0723293122','Str. Fabricii 2, Braila', '6010611672883','rmi_m@gmail.com', sysdate+5, 'rmih');
```

At the bottom of the workspace, the message "Trigger INSERT_INVALID compiled" is displayed. The status bar at the bottom right shows the date and time: 02-Jan-22, 01:55.

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' sidebar lists various database objects like Oracle Connections, Tables, Procedures, Functions, Triggers, Types, and Materialized Views. The central workspace contains a 'Query Builder' window with the following PL/SQL code:

```

CREATE OR REPLACE TRIGGER insert_invalid
BEFORE INSERT OR UPDATE ON client
BEGIN
    IF (TO_CHAR(SYSDATE,'D') = 1) THEN
        RAISE_APPLICATION_ERROR(-20000,'Tabelul nu poate fi actualizat. Incercati din nou Luni!');
    ELSIF (TO_CHAR(SYSDATE,'HH24') NOT BETWEEN 8 and 20) THEN
        RAISE_APPLICATION_ERROR(-20000,'Tabelul nu poate fi actualizat decat in timpul programului!');
    END IF;
END;
/

```

Below this, a 'Script Output' window shows the execution results:

```

insert into client values(500, 'Radu', 'Mihai', '0723293122','Str. Fabricii 2, Braila', '6010611672883','rmi_m@gmail.com');
drop trigger insert_invalid;

```

At the bottom, an 'Error report' window displays the following errors:

```

Error starting at line : 12 in command -
insert into client values(500, 'Radu', 'Mihai', '0723293122','Str. Fabricii 2, Braila', '6010611672883','rmi_m@gmail.com',
Error report -
ORA-20000: Tabelul nu poate fi actualizat. Incercati din nou Luni!
ORA-06512: at "RMI.INSERT_INVALID", line 3
ORA-04088: error during execution of trigger 'RMI.INSERT_INVALID'

```

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

Enunt: Definiti un trigger care sa impiedice cresterea pretului cartilor care nu au starea “Foarte Buna”.

Rezolvare:

```

CREATE OR REPLACE TRIGGER pret_stare
BEFORE INSERT OR UPDATE ON exemplar
FOR EACH ROW
WHEN (NEW.pret_exemplar > OLD.pret_exemplar AND OLD.stare_exemplar != 'Foarte Buna')
BEGIN
    RAISE_APPLICATION_ERROR(-20002, 'Pretul nu poate fi marit din cauza
starii cartii');
END;

```

/

Testare:**UPDATE exemplar**

SET pret_exemplar=pret_exemplar+1;

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab contains the following PL/SQL code:

```
CREATE OR REPLACE TRIGGER pret_stare
BEFORE INSERT OR UPDATE ON exemplar
FOR EACH ROW
WHEN (NEW.pret_exemplar > OLD.pret_exemplar AND OLD.stare_exemplar != 'Foarte Buna')
BEGIN
    RAISE_APPLICATION_ERROR(-20002, 'Pretul nu poate fi marit din cauza starii cartii');
END;

DROP TRIGGER pret_stare;

UPDATE exemplar
```

The 'Script Output' tab shows the message: "Trigger PRET_STARE compiled". The status bar at the bottom right indicates: "13°C Sunny 17:08 ENG 04-Jan-22".

The screenshot shows the Oracle SQL Developer interface. In the central workspace, a trigger named 'PRET_STARE' is defined:

```

BEGIN
  RAISE_APPLICATION_ERROR(-20002, 'Pretul nu poate fi marit din cauza starii cartii');
END;
/
DROP TRIGGER pret_stare;

UPDATE exemplar
SET pret_exemplar=pret_exemplar+1;

select * from exemplar;

```

The 'Script Output' tab at the bottom shows the compilation message:

Trigger PRET_STARE compiled

Below it, the 'Messages' tab displays the error output:

Error starting at line : 12 in command -
 UPDATE exemplar
 SET pret_exemplar=pret_exemplar+1
 Error report -
 ORA-20002: Pretul nu poate fi marit din cauza starii cartii
 ORA-06512: at "RMI.PRET_STARE", line 2
 ORA-04088: error during execution of trigger 'RMI.PRET_STARE'

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

Enunt: Definiti un trigger pe tabelul actiuni_baza_date care sa insereze date in tabel la apelarea comenziilor ALTER, CREATE si DROP in baza de date.

Rezolvare:

```

CREATE TABLE actiuni_baza_date
  (nume_baza_date VARCHAR2(100),
   user_activ VARCHAR2(100),
   actiune VARCHAR2(100),
   tip_object_actiune VARCHAR2(100),
   nume_object_actiune VARCHAR2(100),
   data_actiune DATE);

```

```

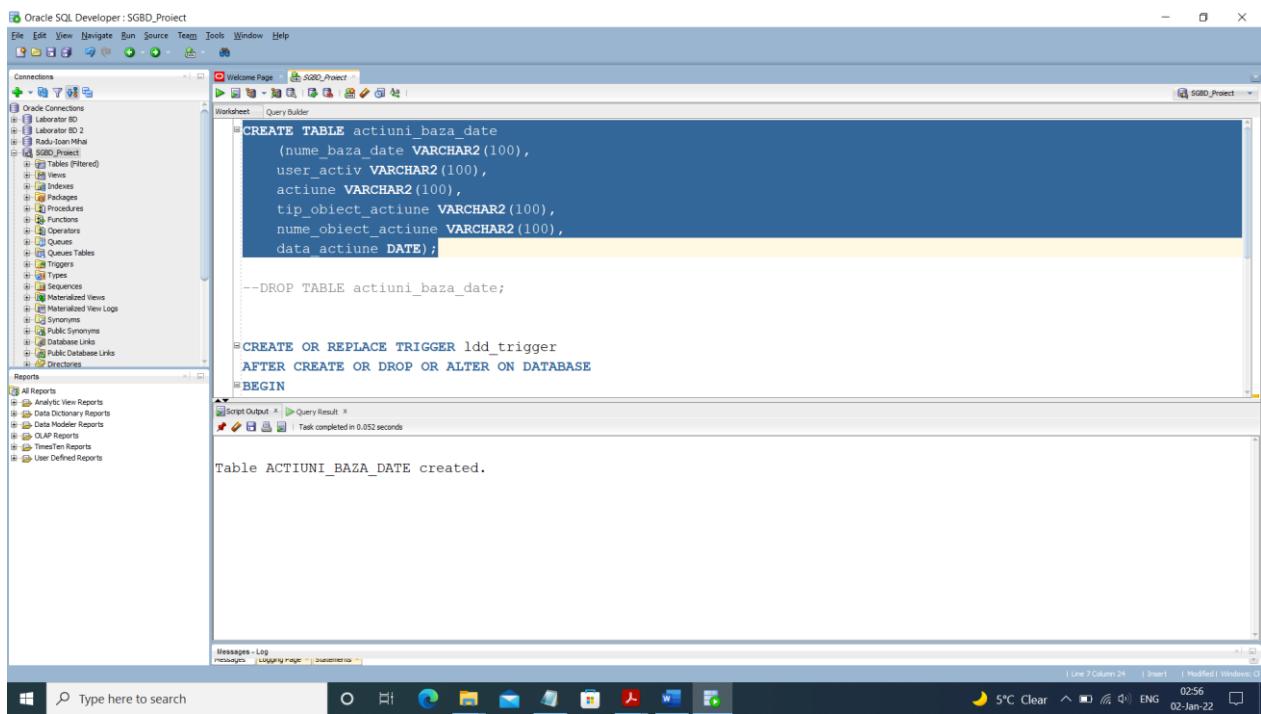
CREATE OR REPLACE TRIGGER ldd_trigger
AFTER CREATE OR DROP OR ALTER ON DATABASE
BEGIN
  INSERT INTO actiuni_baza_date VALUES

```

```
(SYS DATABASE_NAME, SYS LOGIN_USER, SYS SYSEVENT,  
SYS DICTIONARY_OBJ_TYPE, SYS DICTIONARY_OBJ_NAME,  
SYS DATE);  
END;  
/
```

Testare:

```
CREATE TABLE testare  
(testare VARCHAR2(100));  
  
DROP TABLE testare;  
  
SELECT * FROM actiuni_baza_date;
```



Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Laborator ID 2
 - Radu-Ioan Mihai
 - SGBD_Project
- Tables (Filtered)
- Views
- Indexes
- Sequences
- Procedures
- Functions
- Operators
- Triggers
- Types
- Sequences
- Materialized Views
- Materialized View Logs
- Synonyms
- Public Synonyms
- Database Links
- Public Database Links
- Directories

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Model Reports
- OLAP Reports
- TimetTen Reports
- User Defined Reports

Worksheet | Query Builder

```

data_actiune DATE);

--DROP TABLE actiuni_baza_date;

CREATE OR REPLACE TRIGGER ldd_trigger
AFTER CREATE OR DROP OR ALTER ON DATABASE
BEGIN
  INSERT INTO actiuni_baza_date VALUES
  (SYS.DATABASE_NAME, SYS.LOGIN_USER, SYS.SYSEVENT, SYS.DICTIONARY_OBJ_TYPE, SYS.DICTIONARY_OBJ_NAME, SYSDATE);
END;
/
--DROP TRIGGER ldd_trigger;

```

Script Output | Query Result

Table ACTIUNI_BAZA_DATE created.

Trigger LDD_TRIGGER compiled

Messages - Log

Type here to search

5°C Clear 02:56 ENG 02-Jan-22

Oracle SQL Developer : SGBD_Project

Connections

- Oracle Connections
 - Laborator ID
 - Laborator ID 2
 - Radu-Ioan Mihai
 - SGBD_Project
- Tables (Filtered)
- Views
- Indexes
- Sequences
- Procedures
- Functions
- Operators
- Triggers
- Types
- Sequences
- Materialized Views
- Materialized View Logs
- Synonyms
- Public Synonyms
- Database Links
- Public Database Links
- Directories

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Model Reports
- OLAP Reports
- TimetTen Reports
- User Defined Reports

Worksheet | Query Builder

```

INSERT INTO actiuni_baza_date VALUES
(SYS.DATABASE_NAME, SYS.LOGIN_USER, SYS.SYSEVENT, SYS.DICTIONARY_OBJ_TYPE, SYS.DICTIONARY_OBJ_NAME, SYSDATE);
END;
/

--DROP TRIGGER ldd_trigger;

CREATE TABLE testare
(testare VARCHAR2(100));

DROP TABLE testare;

SELECT * FROM actiuni_baza_date;

```

Script Output | Query Result

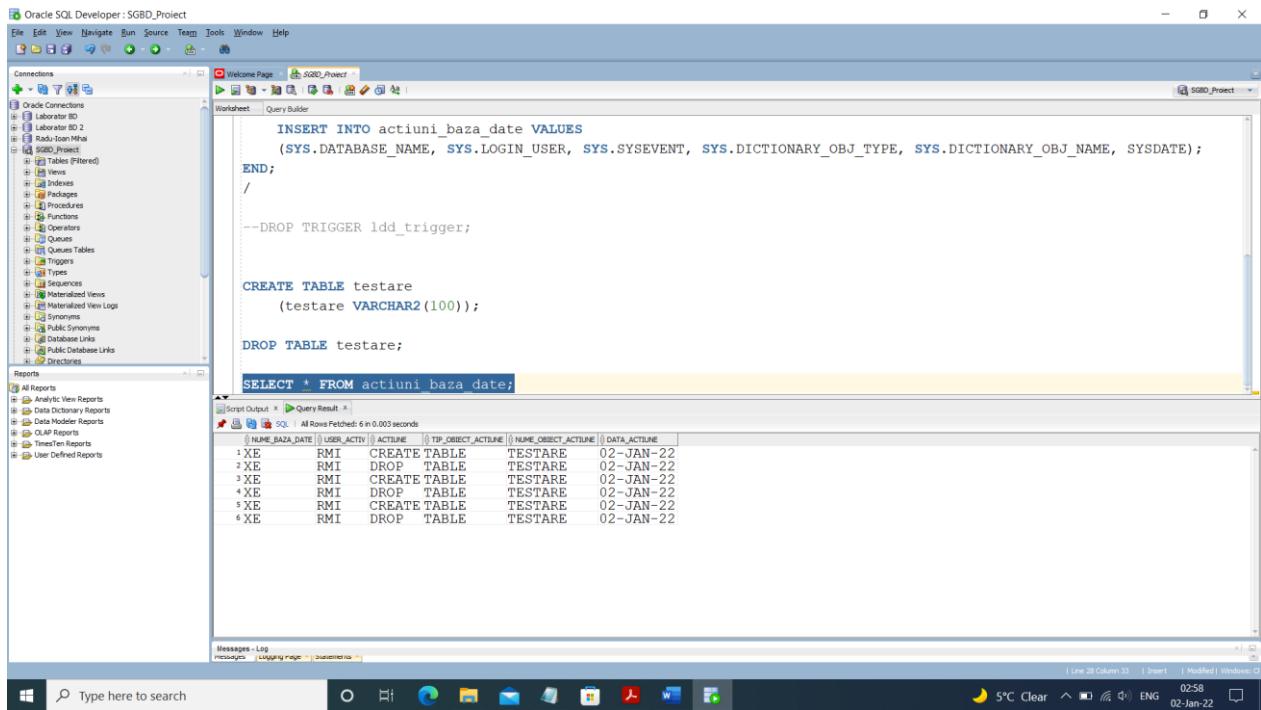
Table TESTARE created.

Table TESTARE dropped.

Messages - Log

Type here to search

5°C Clear 02:57 ENG 02-Jan-22



13. Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului

Rezolvare:

```

CREATE OR REPLACE PACKAGE pachet_ex13
IS
    PROCEDURE detalii_clienti_angajati
        (an VARCHAR2);
    PROCEDURE top3
        (an VARCHAR2);
    FUNCTION reducere_editura
        (v_nume_editura.nume_editura%TYPE DEFAULT 'Nemira')
        RETURN NUMBER;
    PROCEDURE
        p9(num_c client.nume_client%TYPE);
END pachet_ex13;
/

```

```
CREATE OR REPLACE PACKAGE BODY pachet_ex13
IS
    PROCEDURE detalii_clienti_angajati
    (an VARCHAR2)
    AS
        TYPE tablou_indexat IS TABLE OF client%ROWTYPE INDEX BY
        PLS_INTEGER;
        detalii_client tablou_indexat;
        TYPE tablou_imbricat IS TABLE OF angajat%ROWTYPE;
        detalii_angajat tablou_imbricat:= tablou_imbricat();
        nr1 NUMBER;
        nr2 NUMBER;

    BEGIN
        SELECT null INTO nr1
        FROM client
        WHERE TO_CHAR(data_inregistrare, 'YYYY') <= an AND ROWNUM=1;

        SELECT null INTO nr2
        FROM angajat
        WHERE TO_CHAR(data_angajare, 'YYYY') <= an AND ROWNUM=1;

        SELECT * BULK COLLECT INTO detalii_client
        FROM client
        WHERE TO_CHAR(data_inregistrare, 'YYYY') <= an;

        SELECT * BULK COLLECT INTO detalii_angajat
        FROM angajat
        WHERE TO_CHAR(data_angajare, 'YYYY') <= an;

        DBMS_OUTPUT.PUT_LINE('CLIENTI: ');

        FOR i IN detalii_client.FIRST..detalii_client.LAST LOOP
            DBMS_OUTPUT.PUT_LINE (detalii_client(i).nume_client || ' ' ||
            detalii_client(i).prenume_client || ' -> ' || NVL(detalii_client(i).email_client,'Adresa
            email inexistentă'));
        END LOOP;
```

```

DBMS_OUTPUT.PUT_LINE('ANGAJATI: ');

FOR i IN detalii_angajat.FIRST..detalii_angajat.LAST LOOP
    DBMS_OUTPUT.PUT_LINE (detalii_angajat(i).nume_angajat || ' ' ||
detalii_angajat(i).prenume_angajat || ' -> ' ||
NVL(detalii_angajat(i).email_angajat,'Adresa email inexistentă'));
END LOOP;

```

```

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Nu există clienti sau angajați
înregistrati înainte de ' || an);
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001, 'Alta eroare!');
END detalii_clienti_angajati;

```

```

PROCEDURE
    top3(an VARCHAR2) IS
CURSOR c IS
    SELECT idc, numar FROM
        (SELECT e.id_carte idc, COUNT(*) numar
        FROM exemplar e
        JOIN comanda co ON (co.id_comanda = e.id_comanda)
        WHERE TO_CHAR(co.data_comanda,'YYYY') = an
        GROUP BY id_carte
        ORDER BY 2 DESC,1)
        WHERE ROWNUM<=3;
n1 NUMBER;
n2 NUMBER;
BEGIN
    SELECT idc, numar INTO n1, n2
    FROM
        (SELECT e.id_carte idc, COUNT(*) numar
        FROM exemplar e
        JOIN comanda co ON (co.id_comanda = e.id_comanda)
        WHERE TO_CHAR(co.data_comanda,'YYYY') = an
        GROUP BY id_carte
        ORDER BY 2 DESC,1)
        WHERE ROWNUM=1;

```

```

FOR i IN c LOOP
    DBMS_OUTPUT.PUT_LINE ('ID Carte: ' || i.idc || ' -> Numar comenzi in ' ||
an || ':' || i.numar);
END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Nu exista carti vandute in ' ||
an);
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001,'Alta eroare!');
END top3;

FUNCTION reducere_editura
(v_numa editura.nume_editura%TYPE DEFAULT 'Nemira')
RETURN NUMBER IS
    pret_total NUMBER(10,2);
    id_ed editura.id_editura%TYPE;
BEGIN
    SELECT id_editura
    INTO id_ed
    FROM editura e
    WHERE upper(nume_editura)=upper(v_numa);

    SELECT SUM(pret_exemplar-(5/100)*pret_exemplar)
    INTO pret_total
    FROM editura e
    JOIN carte c ON (c.id_editura=e.id_editura)
    JOIN exemplar ex ON (ex.id_carte=c.id_carte)
    WHERE upper(nume_editura)=upper(v_numa);

    IF pret_total >= 100 THEN
        pret_total := pret_total - (10/100)*pret_total;
    END IF;

    RETURN pret_total;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Editura cautata nu exista.');

```

```

WHEN TOO_MANY_ROWS THEN
    RAISE_APPLICATION_ERROR(-20001, 'Exista mai multe edituri cu acest
nume!');
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20002, 'Alta eroare!');
END reducere_editura;

```

```

PROCEDURE
p9(ume_c client.ume_client%TYPE)
IS ccod client.id_client%TYPE;
CURSOR c1 IS
    SELECT c.ume_client ||' '|| c.prenume_client as NumeClient, data_comanda,
ume_angajat ||' '|| prenume_angajat as NumeAngajat, tip_plata
        FROM client c
        JOIN comanda co ON (c.id_client=co.id_client)
        JOIN preluare p ON (p.id_comanda=co.id_comanda)
        JOIN angajat a ON (p.id_angajat=a.id_angajat)
        JOIN plata pl ON (pl.id_comanda=co.id_comanda)
        WHERE UPPER(c.ume_client)=UPPER(ume_c)
        ORDER BY co.data_comanda DESC, p.data_preluare DESC, a.ume_angajat;
BEGIN
    SELECT id_client
    INTO ccod
    FROM client c
    WHERE UPPER(c.ume_client)=UPPER(ume_c);
    DBMS_OUTPUT.PUT_LINE('Clientul cu id '|| ccod ||': ');
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('-----');
    FOR i in c1 LOOP
        DBMS_OUTPUT.PUT_LINE('Nume complet: '|| i.NumeClient);
        DBMS_OUTPUT.NEW_LINE();
        DBMS_OUTPUT.PUT_LINE('Data comanda: '|| i.data_comanda);
        DBMS_OUTPUT.NEW_LINE();
        DBMS_OUTPUT.PUT_LINE('Angajat: '|| i.NumeAngajat);
        DBMS_OUTPUT.NEW_LINE();
        DBMS_OUTPUT.PUT_LINE('Tip Plata: '|| i.tip_plata);
        DBMS_OUTPUT.NEW_LINE();
        DBMS_OUTPUT.PUT_LINE('-----');
    END LOOP;

```

```

    END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Clientul cu numele respectiv nu
exista.');
    WHEN TOO_MANY_ROWS THEN
        RAISE_APPLICATION_ERROR(-20001, 'Mai multi clienti cu acelasi
nume');
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20002, 'Alta eroare!');
END p9;
END pachet_ex13;
/

```

Testare:

```
SET SERVEROUTPUT ON;
```

```
execute pachet_ex13.detalii_clienti_angajati(2021);
/
```

```
BEGIN
    pachet_ex13.top3(2021);
END;
/
```

```
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| 
pachet_ex13.reducere_editura('Editura Rao'));
END;
/
```

```
BEGIN
    pachet_ex13.p9('Radu');
END;
```

The screenshot shows the Oracle SQL Developer interface with the 'SGBD_Project' workspace selected. In the 'Worksheet' tab, a PL/SQL block is being compiled:

```

    DBMS_OUTPUT.PUT_LINE('Angajat: ' || i.NumeAngajat);
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('Tip Plata: ' || i.tip_plata);
    DBMS_OUTPUT.NEW_LINE();
    DBMS_OUTPUT.PUT_LINE('-----');
END LOOP;
DBMS_OUTPUT.PUT_LINE('-----');
DBMS_OUTPUT.NEW_LINE();
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Clientul cu numele respectiv nu exista.');
    WHEN TOO_MANY_ROWS THEN
        RAISE_APPLICATION_ERROR(-20001, 'Mai multi clienti cu acelasi nume');
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20002, 'Alta eroare!');
END p9;
END pachet_ex13;

```

The 'Script Output' window at the bottom shows the compilation results:

```

Package PACHET_EX13 compiled

Package Body PACHET_EX13 compiled

```

The status bar at the bottom right indicates: 13°C Sunny 21:10 ENG 05-Jan-22.

The screenshot shows the Oracle SQL Developer interface with the 'SGBD_Project' workspace selected. In the 'Worksheet' tab, a PL/SQL block is being executed:

```

execute pachet_ex13.detalii_clienti_angajati(2021);

BEGIN
    pachet_ex13.top3(2021);
END;
/

BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este '|| pachet_ex13.reducere_editura('Editura Rao'));
END;
/

```

The 'Script Output' window at the bottom shows the execution results:

```

PL/SQL procedure successfully completed.

Vasilescu Simona -> vasimona@gmail.com
Popescu Mihaela -> popmiha@gmail.com
Pavaloiu Anca -> pavaloianu@gmail.com
Dumitrascu Iulia -> iuliad@gmail.com
Mosoi Andrei -> andreimosoi@gmail.com
Gherghie Vladimir -> vlag@gmail.com
Ceparu Andrei -> cepi2001@gmail.com

```

The status bar at the bottom right indicates: 13°C Sunny 21:19 ENG 05-Jan-22.

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar lists several database connections, including 'Laborator ID', 'Laborator ID 2', and 'SGBD_Project'. The 'Worksheet' tab displays a PL/SQL block:

```

execute pachet_ex13.detalii_clienti_angajati(2021);
/
BEGIN
    pachet_ex13.top3(2021);
END;
/
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| pachet_ex13.reducere_editura('Editura Rao'));
END;
/

```

The 'Script Output' window shows the results of the execution:

```

ID Carte: 5 -> Numar comenzi in 2021: 3
ID Carte: 2 -> Numar comenzi in 2021: 1
ID Carte: 6 -> Numar comenzi in 2021: 1

PL/SQL procedure successfully completed.

```

The status bar at the bottom indicates the task completed in 0.108 seconds.

This screenshot shows the same Oracle SQL Developer environment after modifications. The 'Worksheet' tab now contains a different PL/SQL block:

```

BEGIN
    pachet_ex13.top3(2021);
END;
/
BEGIN
    DBMS_OUTPUT.PUT_LINE('Pretul final este'|| pachet_ex13.reducere_editura('Editura Rao'));
END;
/
BEGIN
    pachet_ex13.p9('Radu');
END;

```

The 'Script Output' window shows the output:

```

Pretul final este 47.5

PL/SQL procedure successfully completed.

```

The status bar at the bottom indicates the task completed in 0.054 seconds.

The screenshot shows the Oracle SQL Developer interface. In the top-left corner, there's a 'Connections' tree view showing various database objects like Oracle Connections, Schemas, and Tables. The main workspace has a 'Worksheet' tab open, displaying the following PL/SQL code:

```

BEGIN
  DBMS_OUTPUT.PUT_LINE('Pretul final este '|| pachet_ex13.reducere_editura('Editura Rao'));
END;
/

```

Below this, another block of code is shown:

```

BEGIN
  pachet_ex13.p9('Radu');
END;

```

In the bottom-right pane, the 'Script Output' window displays the results of the execution:

```

Clientul cu id 2:

-----
Nume complet: Radu Ioana
Data comanda: 29-JUN-20
Angajat: Vasilescu Simona
Tip Plata: cash

```

The status bar at the bottom right shows the date as 05-Jan-22, the time as 21:20, and the system temperature as 13°C.

14. Definiți un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri).

Enunt: Pentru a imbunatati calitatea serviciilor anticariatului, patronul a cerut construirea unui pachet care sa il ajute la:

- Afisarea, cu ajutorul unei proceduri, id-ului si numelui complet al angajatilor neperformanti, care nu au contribuit la nicio comanda de cand au fost angajati. Se exclud de la aceasta verificare posturile care nu au acest task in fisa posturilor: om de serviciu, contabil, serviciu tehnic si responsabil inventar.
- O functie care sa returneze procentul comenzilor dintr-un an primit ca parametru care nu au fost preluate in aceeasi zi cu data comenzii.
- O procedura ce afiseaza o lista stocata intr-o matrice cu depozitele si titlurile exemplarelor vandute. In cazul in care nu au fost exemplare vandute din acel depozit, se va afisa un mesaj corespunzator.
- O functie ce afiseaza review-urile suspecte (cu scopul de a denigra firma sau facute de catre un bot) si numarul acestora, considerate review-urile cu un numar mai mic de stele decat cel dat ca parametru sau textul recenziei este mai scurt de 10 caractere. Pentru fiecare se va afisa titlul cartii, username-ul persoanei care a facut review-ul si in care dintre cele 2 cazuri de mai sus se

incadreaza. Daca se incadreaza in ambele cazuri atunci se vor afisa ambele mesaje!

Rezolvare:

```

CREATE OR REPLACE PACKAGE pachet_ex14
AS
    TYPE v_id_dep IS VARRAY(100) OF depozit.id_depozit%TYPE;
    TYPE v_carti IS VARRAY(100) OF carte.titlu_carte%TYPE;
    TYPE matrice_titluri_dep IS VARRAY(100) OF v_carti;
    TYPE info_angajat IS RECORD (a_id angajat.id_angajat%TYPE,
                                  a_nume_angajat.nume_angajat%TYPE,
                                  a_prenume_angajat.prenume_angajat%TYPE);
    TYPE detalii_angajat IS TABLE OF info_angajat;

    TYPE info_comenzi IS RECORD (c_id comanda.id_comanda%TYPE,
                                 data_p preluare.data_preluare%TYPE,
                                 data_c comanda.data_comanda%TYPE);
    TYPE detalii_comenzi IS TABLE OF info_comenzi INDEX BY
PLS_INTEGER;

    TYPE info_rec IS RECORD (c_id client.id_client%TYPE,
                            c_nume_client.nume_client%TYPE,
                            c_prenume_client.prenume_client%TYPE,
                            c_email_client.email_client%TYPE,
                            c_uname_client.username_client%TYPE,
                            r_cont_recenzie.continut_recenzie%TYPE,
                            r_stele_recenzie.numar_stele%TYPE,
                            car_titlu_carte.titlu_carte%TYPE);
    TYPE detalii_rec IS TABLE OF info_rec;

    PROCEDURE afisare_angajati_neperformanti;
    PROCEDURE afisare_depozitare_carti;

    FUNCTION intarziere_comenzi
    (an_comenzi NUMBER DEFAULT 2021)
    RETURN NUMBER;

    FUNCTION avertizare_review
    (nr_star NUMBER DEFAULT 2.5)

```

```
    RETURN NUMBER;  
  
END pachet_ex14;  
  
CREATE OR REPLACE PACKAGE BODY pachet_ex14  
AS  
  
PROCEDURE afisare_angajati_neperformanti  
AS  
v_angj detalii_angajat := detalii_angajat();  
TYPE tablou_agj IS TABLE OF angajat.id_angajat%TYPE;  
tab_id_ang tablou_agj:=tablou_agj();  
indx1 NUMBER;  
indx2 NUMBER;  
BEGIN  
  
SELECT id_angajat, nume_angajat, prenume_angajat  
BULK COLLECT INTO v_angj  
FROM angajat  
WHERE LOWER(post)!='om de serviciu'  
AND LOWER(post)!='responsabil inventar'  
AND LOWER(post)!='serviciu tehnic'  
AND LOWER(post)!='contabil';  
  
SELECT DISTINCT * BULK COLLECT INTO tab_id_ang  
FROM  
(SELECT DISTINCT id_angajat  
FROM preluare  
UNION  
SELECT DISTINCT id_angajat  
FROM plata) ;  
  
FOR i IN v_angj.FIRST..v_angj.LAST LOOP  
    indx1:=0;  
    FOR j IN tab_id_ang.FIRST..tab_id_ang.LAST LOOP  
        IF v_angj(i).a_id = tab_id_ang(j) THEN  
            indx1:=1;  
        END IF;  
    END LOOP;  
    IF indx1=0 THEN
```

```

    DBMS_OUTPUT.PUT_LINE('-----');
    DBMS_OUTPUT.PUT_LINE(v_angj(i).a_id);
    DBMS_OUTPUT.PUT_LINE(v_angj(i).a_nu);
    DBMS_OUTPUT.PUT_LINE(v_angj(i).a_prnu);
    DBMS_OUTPUT.PUT_LINE('-----');

    END IF;
END LOOP;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Nu exista date pentru a fi
introduce in tabel.');
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001,'Alta eroare!');

END afisare_angajati_neperformanti;

```

```

FUNCTION intarziere_comenzi
(an_comenzi NUMBER DEFAULT 2021)
RETURN NUMBER IS
    int_procent NUMBER(10,2);
    v_com detaliu_comenzi;
    nr_com NUMBER;
    nr_com_intarziate NUMBER;
    nrtest NUMBER;
BEGIN
    nr_com := 0;
    nr_com_intarziate := 0;

    SELECT null INTO nrtest
    FROM preluare p, comanda c
    WHERE p.id_comanda = c.id_comanda
    AND TO_CHAR(c.data_comanda, 'YYYY') = an_comenzi;

    SELECT p.id_comanda, p.data_preluare, c.data_comanda
    BULK COLLECT INTO v_com
    FROM preluare p, comanda c
    WHERE p.id_comanda = c.id_comanda

```

```

AND TO_CHAR(c.data_comanda, 'YYYY') = an_comenzi;

FOR i IN v_com.FIRST..v_com.LAST LOOP
    nr_com := nr_com+1;

    IF (v_com(i).data_p - v_com(i).data_c)>0 THEN
        nr_com_intarziate := nr_com_intarziate + 1;
    END IF;

END LOOP;

int_procent := nr_com_intarziate / nr_com * 100;

RETURN int_procent;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20000, 'Nu exista comenzi in anul
cautat.');
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001,'Alta eroare!');
END intarzire_comenzi;

```

```

PROCEDURE afisare_depozitare_carti
AS
    vector_id_depozit v_id_dep := v_id_dep();
    m_depozitare matrice_titluri_dep := matrice_titluri_dep();
BEGIN
    SELECT id_depozit
    BULK COLLECT INTO vector_id_depozit
    FROM depozit;

    FOR i IN vector_id_depozit.FIRST..vector_id_depozit.LAST LOOP
        m_depozitare.extend();
        SELECT c.titlu_carte
        BULK COLLECT INTO m_depozitare(i)
        FROM carte c
    
```

```

    JOIN exemplar e ON (c.id_carte = e.id_carte)
    JOIN depozit d ON (e.id_depozit = d.id_depozit)
    WHERE d.id_depozit = i;
END LOOP;

DBMS_OUTPUT.PUT_LINE('-----');
FOR i IN m_depozitare.FIRST..m_depozitare.LAST LOOP
    DBMS_OUTPUT.PUT('Depozitul numarul ' || i || ':');
    IF m_depozitare(i).COUNT > 0 THEN
        FOR j IN 1..m_depozitare(i).COUNT LOOP
            DBMS_OUTPUT.PUT(m_depozitare(i)(j) || '; ');
        END LOOP;
    ELSE
        DBMS_OUTPUT.PUT('Nu au fost vandute exemplare din acest
depozit');
    END IF;
    DBMS_OUTPUT.PUT_LINE('');
    DBMS_OUTPUT.PUT_LINE('-----');
END LOOP;

EXCEPTION
WHEN NO_DATA_FOUND THEN
    RAISE_APPLICATION_ERROR(-20000, 'Nu exista date pentru a fi
introduce in tabel.');
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20001,'Alte eroare!');

END afisare_depozitare_carti;

```

```

FUNCTION avertizare_review
(nr_star NUMBER DEFAULT 2.5)
RETURN NUMBER IS
    v_rec detalii_rec := detalii_rec();
    nr_r NUMBER;
    bool_verif NUMBER;

```

BEGIN

```

nr_r := 0;

SELECT c.id_client, c.numere_client, c.prenume_client, c.email_client,
c.username_client, r.continut_recenzie, r.numar_stele, car.titlu_carte
BULK COLLECT INTO v_rec
FROM client c
JOIN recenzie r ON (c.id_client = r.id_client)
JOIN carte car ON (car.id_carte = r.id_carte)
ORDER BY c.id_client;

FOR i IN v_rec.FIRST..v_rec.LAST LOOP

    bool_verif := 0;

    IF v_rec(i).r_stele <= nr_star THEN
        bool_verif := 1;
        DBMS_OUTPUT.PUT_LINE('Posibile probleme cu recenzia ' ||
v_rec(i).car_titlu || ' de ' || v_rec(i).c_uname);
        DBMS_OUTPUT.PUT_LINE('Cauza: Rating foarte mic!');
        DBMS_OUTPUT.PUT_LINE('-----');
    END IF;

    IF LENGTH(v_rec(i).r_cont) <= 10 THEN
        bool_verif := 1;
        DBMS_OUTPUT.PUT_LINE('Posibile probleme cu recenzia ' ||
v_rec(i).car_titlu || ' de ' || v_rec(i).c_uname);
        DBMS_OUTPUT.PUT_LINE('Cauza: Recenzie mult prea scurta!');
        DBMS_OUTPUT.PUT_LINE('-----');
    END IF;

    IF bool_verif = 1 THEN
        nr_r := nr_r + 1;
    END IF;
END LOOP;

RETURN nr_r;

EXCEPTION
    WHEN NO_DATA_FOUND THEN

```

```
        RAISE_APPLICATION_ERROR(-20000, 'Nu exista date pentru a fi
introduce in tabel.');
WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001,'Eroare necunoscuta! Incercati
din nou.');
END avertizare_review;

END pachet_ex14;
```

Testare:

```
SET SERVEROUTPUT ON;
BEGIN
    pachet_ex14.afisare_angajati_neperformanti;
END;

BEGIN
    DBMS_OUTPUT.PUT_LINE('Procent comenzi intarziate: ' ||
pachet_ex14.intarziere_comenzi(2021));
END;

BEGIN
    pachet_ex14.afisare_depozitare_carti;
END;

BEGIN
    DBMS_OUTPUT.PUT_LINE('Exista posibile probleme cu ' ||
pachet_ex14.avertizare_review(2) || ' recenzii. Verificati tabelul de mai sus!');
END;
```

```

END pachet_ex14;

SET SERVEROUTPUT ON;
BEGIN
    pachet_ex14.afisare_angajati_neperformanti;
END;

BEGIN
    DBMS_OUTPUT.PUT_LINE(pachet_ex14.intarziree_comenzi(2021));
END;

PL/SQL procedure successfully completed.

2
Mihailescu
Dorina

```



```

END pachet_ex14;

SET SERVEROUTPUT ON;
BEGIN
    pachet_ex14.afisare_angajati_neperformanti;
END;

BEGIN
    DBMS_OUTPUT.PUT_LINE('Procent comenzi intarziate: ' || pachet_ex14.intarziree_comenzi(2021));
END;

BEGIN
    pachet_ex14.afisare_depozitare_carti;
END;

Procent comenzi intarziate: 71.43

PL/SQL procedure successfully completed.

```

Oracle SQL Developer : C:\Users\Radu Mihai\SGBD_Project.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
 - Laborator ID
 - Lab2
 - Lab3
 - Radu-Ioan Mihai
 - SGBD_Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadecors
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - External Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Model Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

SGBD_Project.sql - Welcome Page - SGBD_Project

SQL Worksheet History

Worksheet | Query Builder

```
BEGIN
    pachet ex14.afisare depozitare carti;
```

Depozitul numarul 1:Gerrard: My Autobiography; Umbra Vantului; Biblia Pierduta;

Depozitul numarul 2:Nu au fost vandute exemplare din acest depozit

Depozitul numarul 3:Nu au fost vandute exemplare din acest depozit

Depozitul numarul 4:Ferma Animalelor;

Depozitul numarul 5:Portretul lui Dorian Grey; Umbra Vantului; Umbra Vantului;

Depozitul numarul 6:Bohemian Rhapsody;

Depozitul numarul 7:Ferma Animalelor; Umbra Vantului;

Depozitul numarul 8:Nu au fost vandute exemplare din acest depozit

Depozitul numarul 9:Transparenta; Inferno;

Depozitul numarul 10:Bohemian Rhapsody;

Compiler - Log

Type here to search

Oracle SQL Developer : C:\Users\Radu Mihai\SGBD_Project.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
 - Laborator ID
 - Lab2
 - Lab3
 - Radu-Ioan Mihai
 - SGBD_Project
 - Tables (Filtered)
 - Views
 - Sequences
 - Packages
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Seadecors
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - External Links
 - Public Database Links
 - Directories
- Reports
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Model Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

SGBD_Project.sql - Welcome Page - SGBD_Project

SQL Worksheet History

Worksheet | Query Builder

```
END;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Există posibile probleme cu ' || pachet_ex14.avertizare_review(2) || ' recenzii. Verificați tabelul de mai sus!');
END;
```

Possible probleme cu recenzie Sub Un Cer Sangeriu de ioanair
Cauza: Recenzie mult prea scurta!

Possible probleme cu recenzie Transparenta de ioanair
Cauza: Recenzie mult prea scurta!

Possible probleme cu recenzie Portretul lui Dorian Grey de radub
Cauza: Rating foarte mic!

Possible probleme cu recenzie Biblia Pierduta de aanda
Cauza: Recenzie mult prea scurta!

Există posibile probleme cu 4 recenzii. Verificați tabelul de mai sus!

Compiler - Log

Type here to search

33°C Sunny 1641 27-Aug-22

33°C Sunny 1642 27-Aug-22