Технически Университет – Варна

Проект  
по “Бази Данни”

Студент: Иван Радославов Димов, Ф.№:20621603, Група: 4а, 3к спец. „СИТ“

Съдържание

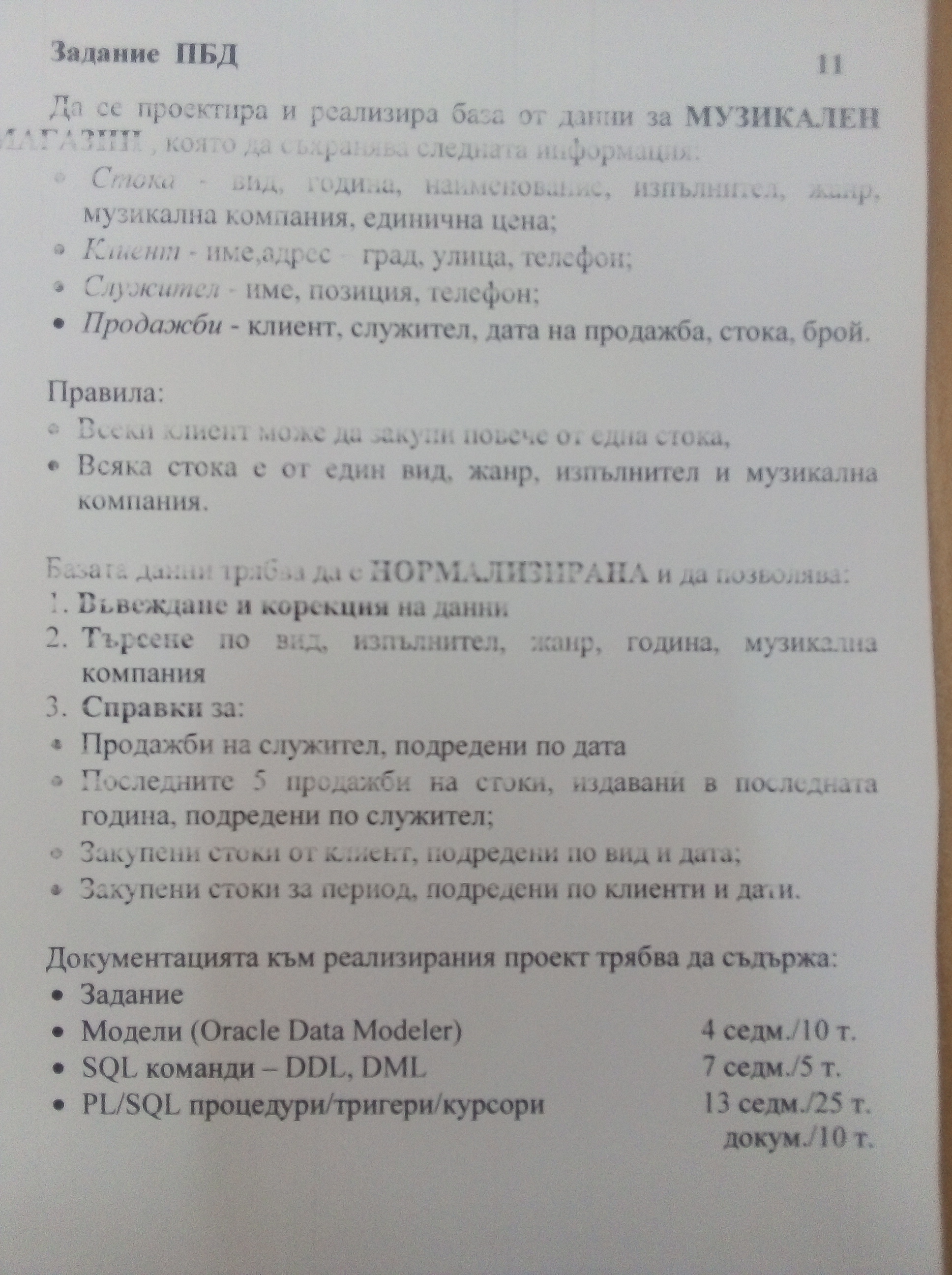
1. Задание

2. Модели

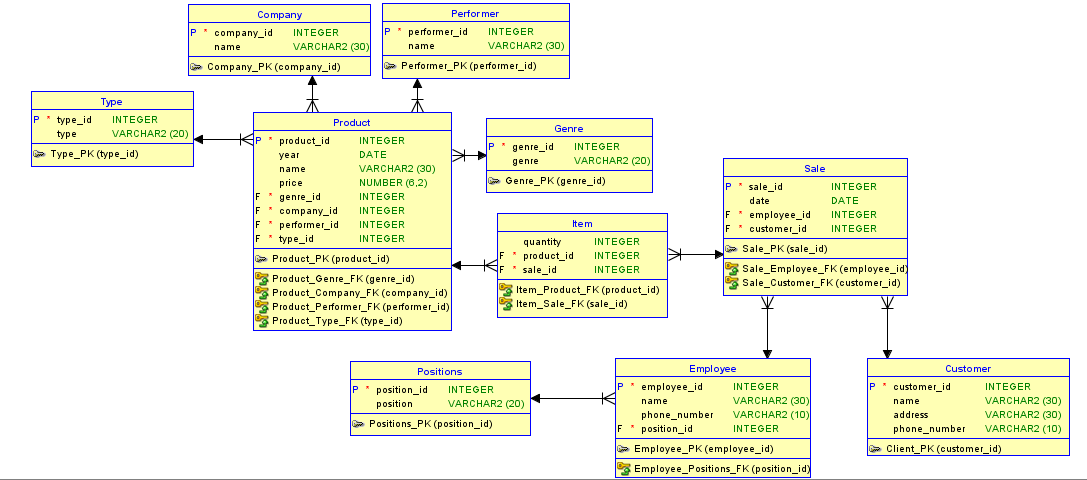
3. SQL команди

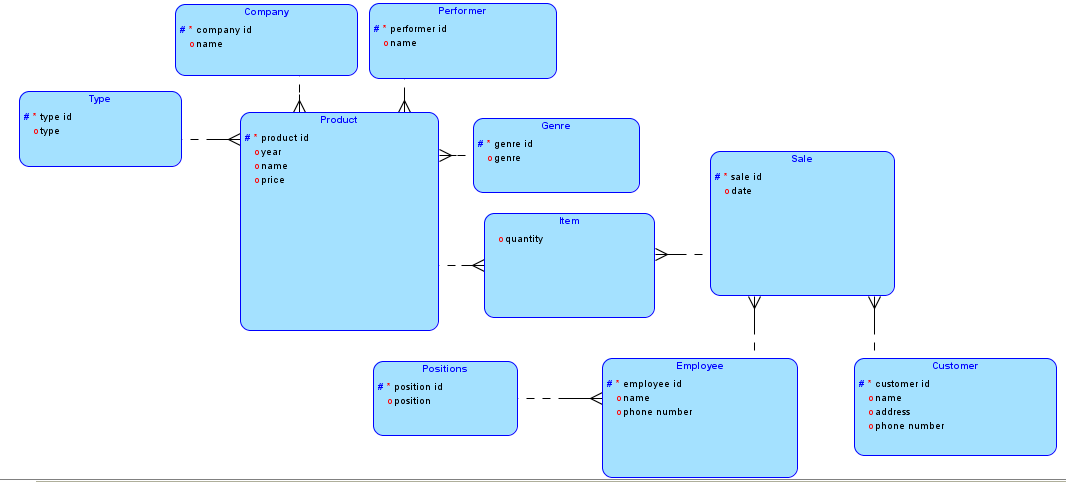
4. PL/SQL процедури/тригери/курсори

5. Примерна работа

1. Задание

2. Модели





3. SQL команди

Създаване на таблиците

create table p\_type(

p\_type\_id int not null,

p\_type\_name varchar(20)

);

alter table p\_type add constraint type\_PK Primary key(p\_type\_id);

create table company(

company\_id int not null,

company\_name varchar(30)

);

alter table company add constraint company\_PK Primary key(company\_id);

create table performer(

performer\_id int not null,

performer\_name varchar(30)

);

alter table performer add constraint performer\_PK Primary key(performer\_id);

create table genre(

genre\_id int not null,

genre\_name varchar(30)

);

alter table genre add constraint genre\_PK Primary key(genre\_id);

create table product(

product\_id int not null,

year\_published date,

product\_name varchar(30),

price number(6,2),

genre\_id int not null,

company\_id int not null,

performer\_id int not null,

p\_type\_id int not null

);

alter table product add constraint product\_PK Primary key(product\_id);

alter table product add constraint product\_genre\_FK Foreign key(genre\_id) references genre(genre\_id);

alter table product add constraint product\_company\_FK Foreign key(company\_id) references company(company\_id);

alter table product add constraint product\_performer\_FK Foreign key(performer\_id) references performer(performer\_id);

alter table product add constraint product\_type\_FK Foreign key(p\_type\_id) references p\_type(p\_type\_id);

create table e\_position(

e\_position\_id int not null,

position\_name varchar(20)

);

alter table e\_position add constraint e\_position\_PK Primary key(e\_position\_id);

create table employee(

employee\_id int not null,

employee\_name varchar(30),

phone\_number varchar(10),

e\_position\_id int not null

);

alter table employee add constraint employee\_PK Primary key(employee\_id);

alter table employee add constraint employee\_position\_FK Foreign key(e\_position\_id) references e\_position(e\_position\_id);

create table customer(

customer\_id int not null,

customer\_name varchar(30),

address varchar(30),

phone\_number varchar(10)

);

alter table customer add constraint customer\_PK Primary key(customer\_id);

create table sale(

sale\_id int not null,

sale\_date date,

employee\_id int not null,

customer\_id int not null

);

alter table sale add constraint sale\_PK Primary key(sale\_id);

alter table sale add constraint sale\_employee\_FK Foreign key(employee\_id) references employee(employee\_id);

alter table sale add constraint sale\_customer\_FK Foreign key(customer\_id) references customer(customer\_id);

create table item(

quantity int,

product\_id int not null,

sale\_id int not null

);

alter table item add constraint item\_product\_FK Foreign key(product\_id) references product(product\_id);

alter table item add constraint item\_sale\_FK Foreign key(sale\_id) references sale(sale\_id);

Въвеждане на примерни данни

insert into company values(1, 'Seven Eight');

insert into company values(2, 'Nuclear Blast');

insert into genre values(1, 'pop-folk');

insert into genre values(2, 'metal');

insert into genre values(3, 'rap');

insert into performer values(1, 'Slavi Trifonov');

insert into performer values(2, 'Joakim Broaden');

insert into performer values(3, 'Eminem');

insert into p\_type values(1, 'CD');

insert into p\_type values(2, 'Disk');

insert into p\_type values(3, 'Blue-Ray');

--update p\_type set p\_type\_name = 'DVD' where p\_type\_id = 2;

insert into product values(1, '15-AUG-2002', 'Nie Produlzhavame', 16.50, 1, 1, 1, 2);

insert into product values(2, '8-SEP-2000', 'Vaxpopuli', 12.60, 1, 1, 1, 1);

insert into product values(3, '2-FEB-2003', 'No Mercy', 16.50, 1, 1, 1, 3);

insert into product values(4, '27-AUG-2001', '40 to 1', 19.99, 2, 2, 2, 1);

insert into product values(5, '6-OCT-2004', 'Primo Victoria', 19.99, 2, 2, 2, 3);

insert into product values(6, '3-JAN-2005', 'Last Stand', 19.99, 2, 2, 2, 1);

insert into product values(7, '1-AUG-2001', 'Til I Collapse', 10.20, 3, 2, 3, 2);

insert into product values(8, '30-JUN-2009', 'Rap God', 10.20, 3, 2, 3, 1);

insert into product values(9, '19-MAR-2002', 'Without me', 23.99, 3, 2, 3, 3);

--update product set price = 17.99 where product\_id = 8;

insert into e\_position values(1, 'clerk');

insert into e\_position values(2, 'manager');

insert into e\_position values(3, 'advertiser');

insert into employee values(1, 'Hristo Ivanov', '0893140560', 1);

insert into employee values(2, 'Georgi Lazarov', '0893535187', 1);

insert into employee values(3, 'Boris Petkov', '0873037189', 2);

insert into employee values(4, 'Hristo Plamenov', '0863137123', 3);

update employee set employee\_name = 'Petar Vasilev' where employee\_id = 4;

insert into customer values(1, 'Bozhana Stancheva', 'Vasil Levski 15', '0817534480');

insert into customer values(2, 'Petar Stoilov', 'tsar Samuil 4', '0847339085');

insert into customer values(3, 'Martin Stoyanov', 'Trakia 6', '0893584581');

insert into customer values(4, 'Todor Petkov', 'Studetska 9', '0827572692');

insert into customer values(5, 'Kolyo Grigorov', 'Hristo Botev 12', '0893084083');

insert into customer values(6, 'Mustafa Cholakov', 'Saedinenie 8', '0823981029');

update customer set address = 'Trakia 3' where customer\_id = 3;

insert into sale values(1, '22-AUG-2022', 1, 1);

insert into sale values(2, '19-SEP-2022', 2, 1);

insert into sale values(3, '20-SEP-2022', 1, 2);

insert into sale values(4, '20-SEP-2022', 3, 1);

insert into sale values(5, '22-SEP-2022', 1, 6);

insert into sale values(6, '25-SEP-2022', 2, 5);

insert into sale values(7, '26-SEP-2022', 1, 3);

insert into sale values(8, '28-SEP-2022', 4, 4);

insert into sale values(9, '28-SEP-2022', 1, 3);

insert into sale values(10, '2-OCT-2022', 2, 1);

insert into sale values(11, '5-OCT-2022', 1, 2);

insert into sale values(12, '6-OCT-2022', 2, 5);

insert into sale values(13, '8-OCT-2022', 1, 6);

insert into sale values(14, '8-OCT-2022', 2, 2);

insert into sale values(15, '10-OCT-2022', 1, 1);

insert into sale values(16, '11-OCT-2022', 2, 1);

insert into sale values(17, '15-OCT-2022', 1, 2);

insert into sale values(18, '19-OCT-2022', 2, 5);

insert into sale values(19, '20-OCT-2022', 4, 1);

insert into sale values(20, '20-OCT-2022', 2, 2);

insert into sale values(21, '21-OCT-2022', 2, 3);

--update sale set customer\_id = 2 where sale\_id = 4;

--delete from sale where sale\_id = 21;

insert into item values(2, 9, 1);

insert into item values(1, 1, 1);

insert into item values(1, 5, 2);

insert into item values(3, 2, 3);

insert into item values(1, 3, 4);

insert into item values(1, 6, 4);

insert into item values(2, 7, 4);

insert into item values(1, 4, 5);

insert into item values(1, 8, 6);

insert into item values(1, 2, 7);

insert into item values(1, 9, 7);

insert into item values(4, 5, 8);

insert into item values(1, 3, 9);

insert into item values(1, 8, 10);

insert into item values(2, 1, 11);

insert into item values(2, 7, 11);

insert into item values(1, 1, 12);

insert into item values(1, 6, 13);

insert into item values(3, 8, 14);

insert into item values(1, 4, 15);

insert into item values(2, 7, 16);

insert into item values(1, 4, 17);

insert into item values(1, 8, 18);

insert into item values(2, 2, 19);

insert into item values(1, 5, 19);

insert into item values(4, 5, 19);

insert into item values(1, 3, 19);

insert into item values(1, 8, 20);

insert into item values(1, 2, 20);

insert into item values(2, 7, 20);

insert into item values(1, 3, 20);

insert into item values(4, 6, 21);

insert into item values(3, 8, 21);

insert into item values(4, 9, 21);

--update item set quantity=2 where sale\_id = 15;

--delete from item where product\_id = 9 and sale\_id =21;

Търсене по вид

select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(t.p\_type\_name) like lower('&type');

Търсене по изпълнител

select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(pe.performer\_name) like lower('&performer');

Търсене по жанр

select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(g.genre\_name) like lower('&genre');

Търсене по година

select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where extract(year from p.year\_published) = &year\_published;

Търсене по компания

select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(c.company\_name) like lower('&company');

Справки

Продажби на служител, подредени по дата

--1

select s.sale\_date, c.customer\_name, e.employee\_name, p.year\_published, p.product\_name, p.price,g.genre\_name, c.company\_name, pe.performer\_name,

t.p\_type\_name from sale s join customer c on c.customer\_id = s.customer\_id

join employee e on s.employee\_id = e.employee\_id join item i on i.sale\_id = s.sale\_id join product p on p.product\_id = i.product\_id

join genre g on g.genre\_id = p.genre\_id join p\_type t on p.p\_type\_id = t.p\_type\_id join performer pe on pe.performer\_id = p.performer\_id

join company c on c.company\_id = p.company\_id

where lower(e.employee\_name) like lower('&name') order by s.sale\_date;

Последните 5 продажби на стоки, издавани в последната година, подредени по служител

--2

select \* from (select s.sale\_date, c.customer\_name, e.employee\_name, p.year\_published, p.product\_name, p.price,g.genre\_name, c.company\_name, pe.performer\_name,

t.p\_type\_name from sale s join customer c on c.customer\_id = s.customer\_id

join employee e on s.employee\_id = e.employee\_id join item i on i.sale\_id = s.sale\_id join product p on p.product\_id = i.product\_id

join genre g on g.genre\_id = p.genre\_id join p\_type t on p.p\_type\_id = t.p\_type\_id join performer pe on pe.performer\_id = p.performer\_id

join company c on c.company\_id = p.company\_id where extract(year from s.sale\_date) = 2022 order by s.sale\_date desc) where rownum <=5 order by employee\_name;

Закупени стоки от клиент, подредени по вид и дата

--3

select i.quantity, p.product\_name, t.p\_type\_name, s.sale\_date, pe.performer\_name, c.company\_name from item i

join product p on i.product\_id = p.product\_id join sale s on i.sale\_id = s.sale\_id join p\_type t

on p.p\_type\_id = t.p\_type\_id join customer c on s.customer\_id = c.customer\_id join performer pe on p.performer\_id = pe.performer\_id

join company c on p.company\_id = c.company\_id where lower(c.customer\_name) like lower('&customer')

order by t.p\_type\_name, s.sale\_date;

Закупени стоки за период, подредени по клиенти и дато

--4

select i.quantity, p.product\_name, t.p\_type\_name, c.company\_name, pe.performer\_name, s.sale\_date, c.customer\_name from item i

join product p on i.product\_id = p.product\_id join sale s on i.sale\_id = s.sale\_id join p\_type t

on p.p\_type\_id = t.p\_type\_id join customer c on s.customer\_id = c.customer\_id join performer pe on p.performer\_id = pe.performer\_id

join company c on p.company\_id = c.company\_id where s.sale\_date between '&start\_date' and '&end\_date'

order by c.customer\_name, s.sale\_date;

4. PL/SQL процедури/тригери/курсори

Тригери за insert

CREATE SEQUENCE company\_seq START WITH 3;

CREATE OR REPLACE TRIGGER comapany\_id\_auto\_trigger

BEFORE INSERT ON company FOR EACH ROW WHEN (NEW.company\_id IS NULL)

BEGIN

:NEW.company\_id := company\_seq.NEXTVAL;

END;

CREATE SEQUENCE customer\_seq START WITH 7;

CREATE OR REPLACE TRIGGER customer\_id\_auto\_trigger

BEFORE INSERT ON customer FOR EACH ROW WHEN (NEW.customer\_id IS NULL)

BEGIN

:NEW.customer\_id := customer\_seq.NEXTVAL;

END;

CREATE SEQUENCE e\_position\_seq START WITH 4;

CREATE OR REPLACE TRIGGER e\_position\_id\_auto\_trigger

BEFORE INSERT ON e\_position FOR EACH ROW WHEN (NEW.e\_position\_id IS NULL)

BEGIN

:NEW.e\_position\_id := e\_position\_seq.NEXTVAL;

END;

CREATE SEQUENCE employee\_seq START WITH 5;

CREATE OR REPLACE TRIGGER employee\_id\_auto\_trigger

BEFORE INSERT ON employee FOR EACH ROW WHEN (NEW.employee\_id IS NULL)

BEGIN

:NEW.employee\_id := employee\_seq.NEXTVAL;

END;

CREATE SEQUENCE genre\_seq START WITH 4;

CREATE OR REPLACE TRIGGER genre\_id\_auto\_trigger

BEFORE INSERT ON genre FOR EACH ROW WHEN (NEW.genre\_id IS NULL)

BEGIN

:NEW.genre\_id := genre\_seq.NEXTVAL;

END;

CREATE SEQUENCE p\_type\_seq START WITH 4;

CREATE OR REPLACE TRIGGER p\_type\_id\_auto\_trigger

BEFORE INSERT ON p\_type FOR EACH ROW WHEN (NEW.p\_type\_id IS NULL)

BEGIN

:NEW.p\_type\_id := p\_type\_seq.NEXTVAL;

END;

CREATE SEQUENCE performer\_seq START WITH 4;

CREATE OR REPLACE TRIGGER performer\_id\_auto\_trigger

BEFORE INSERT ON performer FOR EACH ROW WHEN (NEW.performer\_id IS NULL)

BEGIN

:NEW.performer\_id := performer\_seq.NEXTVAL;

END;

CREATE SEQUENCE product\_seq START WITH 10;

CREATE OR REPLACE TRIGGER product\_id\_auto\_trigger

BEFORE INSERT ON product FOR EACH ROW WHEN (NEW.product\_id IS NULL)

BEGIN

:NEW.product\_id := product\_seq.NEXTVAL;

END;

CREATE SEQUENCE sale\_seq START WITH 22;

CREATE OR REPLACE TRIGGER sale\_id\_auto\_trigger

BEFORE INSERT ON sale FOR EACH ROW WHEN (NEW.sale\_id IS NULL)

BEGIN

:NEW.sale\_id := sale\_seq.NEXTVAL;

END;

Процедури

CREATE OR REPLACE PROCEDURE InsertCompany(

in\_company\_name IN company.company\_name%TYPE

) IS

BEGIN

INSERT INTO company(company\_name) values(in\_company\_name);

END;

CREATE OR REPLACE PROCEDURE InsertCustomer(

in\_customer\_name IN customer.customer\_name%TYPE,

in\_address IN customer.address%TYPE,

in\_phone\_number IN customer.phone\_number%TYPE

) IS

BEGIN

INSERT INTO customer(customer\_name, address, phone\_number) values(in\_customer\_name, in\_address, in\_phone\_number);

END;

CREATE OR REPLACE PROCEDURE InsertE\_position(

in\_position\_name IN e\_position.position\_name%TYPE

) IS

BEGIN

INSERT INTO e\_position(position\_name) values(in\_position\_name);

END;

CREATE OR REPLACE PROCEDURE InsertEmployee(

in\_employee\_name IN employee.employee\_name%TYPE,

in\_phone\_number IN employee.phone\_number%TYPE,

in\_e\_position\_id IN employee.e\_position\_id%TYPE

) IS

BEGIN

INSERT INTO employee(employee\_name, phone\_number, employee.e\_position\_id)

VALUES(in\_employee\_name, in\_phone\_number, in\_e\_position\_id);

END;

CREATE OR REPLACE PROCEDURE InsertGenre(

in\_genre\_name IN genre.genre\_name%TYPE

) IS

BEGIN

INSERT INTO genre(genre\_name) VALUES(in\_genre\_name);

END;

CREATE OR REPLACE PROCEDURE InsertItem(

in\_quantity IN item.quantity%TYPE,

in\_product\_id IN item.product\_id%TYPE,

in\_sale\_id IN item.sale\_id%TYPE

) IS

BEGIN

INSERT INTO item(quantity, product\_id, sale\_id) VALUES(in\_quantity, in\_product\_id, in\_sale\_id);

END;

CREATE OR REPLACE PROCEDURE InsertP\_type(

in\_p\_type\_name IN p\_type.p\_type\_name%TYPE

) IS

BEGIN

INSERT INTO p\_type(p\_type\_name) VALUES(in\_p\_type\_name);

END;

CREATE OR REPLACE PROCEDURE InsertPerformer(

in\_performer\_name IN performer.performer\_name%TYPE

) IS

BEGIN

INSERT INTO performer(performer\_name) VALUES(in\_performer\_name);

END;

CREATE OR REPLACE PROCEDURE InsertProduct(

in\_year\_published IN product.year\_published%TYPE,

in\_product\_name IN product.product\_name%TYPE,

in\_price IN product.price%TYPE,

in\_genre\_id IN product.genre\_id%TYPE,

in\_company\_id IN product.company\_id%TYPE,

in\_performer\_id IN product.performer\_id%TYPE,

in\_p\_type\_id IN product.p\_type\_id%TYPE

) IS

BEGIN

INSERT INTO product(year\_published, product\_name, price, genre\_id, company\_id, performer\_id, p\_type\_id)

VALUES(in\_year\_published, in\_product\_name, in\_price, in\_genre\_id, in\_company\_id, in\_performer\_id, in\_p\_type\_id);

END;

CREATE OR REPLACE PROCEDURE InsertSale(

in\_sale\_date IN sale.sale\_date%TYPE,

in\_employee\_id IN sale.employee\_id%TYPE,

in\_customer\_id IN sale.customer\_id%TYPE

) IS

BEGIN

INSERT INTO sale(sale\_date, employee\_id, customer\_id)

VALUES(in\_sale\_date, in\_employee\_id, in\_customer\_id);

END;

CREATE OR REPLACE PROCEDURE UpdateCompany(

in\_company\_id IN company.company\_id%TYPE,

in\_company\_name IN company.company\_name%TYPE

) IS

BEGIN

UPDATE company set company\_name = in\_company\_name WHERE company\_id = in\_company\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateCustomer(

in\_customer\_id IN customer.customer\_id%TYPE,

in\_customer\_name IN customer.customer\_name%TYPE,

in\_address IN customer.address%TYPE,

in\_phone\_number IN customer.phone\_number%TYPE

) IS

BEGIN

UPDATE customer SET customer\_name = in\_customer\_name, address = in\_address, phone\_number = in\_phone\_number

WHERE customer\_id = in\_customer\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateE\_position(

in\_e\_position\_id IN e\_position.e\_position\_id%TYPE,

in\_position\_name IN e\_position.position\_name%TYPE

) IS

BEGIN

UPDATE e\_position SET position\_name = in\_position\_name WHERE e\_position\_id = in\_e\_position\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateEmployee(

in\_employee\_id IN employee.employee\_id%TYPE,

in\_employee\_name IN employee.employee\_name%TYPE,

in\_phone\_number IN employee.phone\_number%TYPE,

in\_e\_position\_id IN employee.e\_position\_id%TYPE

) IS

BEGIN

UPDATE employee SET employee\_name = in\_employee\_name, phone\_number = in\_phone\_number, e\_position\_id = in\_e\_position\_id

WHERE employee\_id = in\_employee\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateGenre(

in\_genre\_id IN genre.genre\_id%TYPE,

in\_genre\_name IN genre.genre\_name%TYPE

) IS

BEGIN

UPDATE genre SET genre\_name = in\_genre\_name WHERE genre\_id = in\_genre\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateItem(

in\_quantity IN item.quantity%TYPE,

in\_product\_id IN item.product\_id%TYPE,

in\_sale\_id IN item.sale\_id%TYPE

)IS

BEGIN

UPDATE item SET quantity = in\_quantity WHERE product\_id = in\_product\_id AND sale\_id = in\_sale\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateP\_type(

in\_p\_type\_id IN p\_type.p\_type\_id%TYPE,

in\_p\_type\_name IN p\_type.p\_type\_name%TYPE

) IS

BEGIN

UPDATE p\_type SET p\_type\_name = in\_p\_type\_name WHERE p\_type\_id = in\_p\_type\_id;

END;

CREATE OR REPLACE PROCEDURE UpdatePerformer(

in\_performer\_id IN performer.performer\_id%TYPE,

in\_performer\_name IN performer.performer\_name%TYPE

) IS

BEGIN

UPDATE performer SET performer\_name = in\_performer\_name WHERE performer\_id = in\_performer\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateProduct(

in\_product\_id IN product.product\_id%TYPE,

in\_year\_published IN product.year\_published%TYPE,

in\_product\_name IN product.product\_name%TYPE,

in\_price IN product.price%TYPE,

in\_genre\_id IN product.genre\_id%TYPE,

in\_company\_id IN product.company\_id%TYPE,

in\_performer\_id IN product.performer\_id%TYPE,

in\_p\_type\_id IN product.p\_type\_id%TYPE

) IS

BEGIN

UPDATE product SET year\_published = in\_year\_published, product\_name = in\_product\_name, price = in\_price,

genre\_id = in\_genre\_id, company\_id = in\_company\_id, performer\_id = in\_performer\_id, p\_type\_id = in\_p\_type\_id

WHERE product\_id = in\_product\_id;

END;

CREATE OR REPLACE PROCEDURE UpdateSale(

in\_sale\_id IN sale.sale\_id%TYPE,

in\_sale\_date IN sale.sale\_date%TYPE,

in\_employee\_id IN sale.employee\_id%TYPE,

in\_customer\_id IN sale.customer\_id%TYPE

) IS

BEGIN

UPDATE sale SET sale\_date = in\_sale\_date, employee\_id = in\_employee\_id, customer\_id = in\_customer\_id

WHERE sale\_id = in\_sale\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteCompany(

in\_company\_id IN company.company\_id%TYPE

) IS

BEGIN

DELETE FROM company WHERE company\_id = in\_company\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteCustomer(

in\_customer\_id IN customer.customer\_id%TYPE

) IS

BEGIN

DELETE FROM customer WHERE customer\_id = in\_customer\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteE\_position(

in\_e\_position\_id IN e\_position.e\_position\_id%TYPE

) IS

BEGIN

DELETE FROM e\_position WHERE e\_position\_id = in\_e\_position\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteEmployee(

in\_employee\_id IN employee.employee\_id%TYPE

) IS

BEGIN

DELETE FROM employee WHERE employee\_id = in\_employee\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteGenre(

in\_genre\_id IN genre.genre\_id%TYPE

) IS

BEGIN

DELETE FROM genre WHERE genre\_id = in\_genre\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteItem(

in\_product\_id IN item.product\_id%TYPE,

in\_sale\_id IN item.sale\_id%TYPE

) IS

BEGIN

DELETE FROM item WHERE product\_id = in\_product\_id AND sale\_id = in\_sale\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteP\_type(

in\_p\_type\_id IN p\_type.p\_type\_id%TYPE

) IS

BEGIN

DELETE FROM p\_type WHERE p\_type\_id = in\_p\_type\_id;

END;

CREATE OR REPLACE PROCEDURE DeletePerformer(

in\_performer\_id IN performer.performer\_id%TYPE

) IS

BEGIN

DELETE FROM performer WHERE performer\_id = in\_performer\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteProduct(

in\_product\_id IN product.product\_id%TYPE

) IS

BEGIN

DELETE FROM product WHERE product\_id = in\_product\_id;

END;

CREATE OR REPLACE PROCEDURE DeleteSale(

in\_sale\_id IN sale.sale\_id%TYPE

) IS

BEGIN

DELETE FROM sale WHERE sale\_id = in\_sale\_id;

END;

SET SERVEROUTPUT ON

CREATE OR REPLACE PROCEDURE SearchByType(

in\_type IN p\_type.p\_type\_name%TYPE

) IS

BEGIN

FOR v\_prod IN (select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(t.p\_type\_name) like lower(in\_type))

LOOP

DBMS\_OUTPUT.PUT\_LINE('Id:'||v\_prod.product\_id||' Year published:'||v\_prod.year\_published||' Name:'||v\_prod.product\_name

|| ' Genre:' || v\_prod.genre\_name || ' Company:' || v\_prod.company\_name || ' Performer:' || v\_prod.performer\_name ||

' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE SearchByPerformer(

in\_performer IN performer.performer\_name%TYPE

) IS

BEGIN

FOR v\_prod IN (select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(pe.performer\_name) like lower(in\_performer))

LOOP

DBMS\_OUTPUT.PUT\_LINE('Id:'||v\_prod.product\_id||' Year published:'||v\_prod.year\_published||' Name:'||v\_prod.product\_name

|| ' Genre:' || v\_prod.genre\_name || ' Company:' || v\_prod.company\_name || ' Performer:' || v\_prod.performer\_name ||

' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE SearchByGenre(

in\_genre IN genre.genre\_name%TYPE

) IS

BEGIN

FOR v\_prod IN (select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(g.genre\_name) like lower(in\_genre))

LOOP

DBMS\_OUTPUT.PUT\_LINE('Id:'||v\_prod.product\_id||' Year published:'||v\_prod.year\_published||' Name:'||v\_prod.product\_name

|| ' Genre:' || v\_prod.genre\_name || ' Company:' || v\_prod.company\_name || ' Performer:' || v\_prod.performer\_name ||

' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE SearchByYear(

in\_year IN NUMBER

) IS

BEGIN

FOR v\_prod IN (select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where extract(year from p.year\_published) = in\_year)

LOOP

DBMS\_OUTPUT.PUT\_LINE('Id:'||v\_prod.product\_id||' Year published:'||v\_prod.year\_published||' Name:'||v\_prod.product\_name

|| ' Genre:' || v\_prod.genre\_name || ' Company:' || v\_prod.company\_name || ' Performer:' || v\_prod.performer\_name ||

' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE SearchByCompany(

in\_company IN company.company\_name%TYPE

) IS

BEGIN

FOR v\_prod IN (select p.product\_id, p.year\_published, p.product\_name, p.price, g.genre\_name, c.company\_name, pe.performer\_name, t.p\_type\_name

from product p join genre g on p.genre\_id = g.genre\_id join company c on p.company\_id = c.company\_id join performer pe

on p.performer\_id = pe.performer\_id join p\_type t on p.p\_type\_id = t.p\_type\_id where lower(c.company\_name) like lower(in\_company))

LOOP

DBMS\_OUTPUT.PUT\_LINE('Id:'||v\_prod.product\_id||' Year published:'||v\_prod.year\_published||' Name:'||v\_prod.product\_name

|| ' Genre:' || v\_prod.genre\_name || ' Company:' || v\_prod.company\_name || ' Performer:' || v\_prod.performer\_name ||

' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE QueryEmployee(

in\_name IN employee.employee\_name%TYPE

) IS

BEGIN

FOR v\_prod IN (select s.sale\_date, c.customer\_name, e.employee\_name, p.year\_published, p.product\_name, p.price,g.genre\_name, c.company\_name, pe.performer\_name,

t.p\_type\_name from sale s join customer c on c.customer\_id = s.customer\_id

join employee e on s.employee\_id = e.employee\_id join item i on i.sale\_id = s.sale\_id join product p on p.product\_id = i.product\_id

join genre g on g.genre\_id = p.genre\_id join p\_type t on p.p\_type\_id = t.p\_type\_id join performer pe on pe.performer\_id = p.performer\_id

join company c on c.company\_id = p.company\_id

where lower(e.employee\_name) like lower(in\_name) order by s.sale\_date)

LOOP

DBMS\_OUTPUT.PUT\_LINE('On date:'||v\_prod.sale\_date||' customer:'||v\_prod.customer\_name||' Name:'||v\_prod.product\_name

|| ' Published on:' || v\_prod.year\_published || ' price:' || v\_prod.price || ' Genre:' || v\_prod.genre\_name ||

' Company:' || v\_prod.company\_name || ' Performed by:' || v\_prod.performer\_name || ' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE QueryLastSales IS

BEGIN

FOR v\_prod IN (select \* from (select s.sale\_date, c.customer\_name, e.employee\_name, p.year\_published, p.product\_name, p.price,g.genre\_name, c.company\_name, pe.performer\_name,

t.p\_type\_name from sale s join customer c on c.customer\_id = s.customer\_id

join employee e on s.employee\_id = e.employee\_id join item i on i.sale\_id = s.sale\_id join product p on p.product\_id = i.product\_id

join genre g on g.genre\_id = p.genre\_id join p\_type t on p.p\_type\_id = t.p\_type\_id join performer pe on pe.performer\_id = p.performer\_id

join company c on c.company\_id = p.company\_id where extract(year from s.sale\_date) = extract(year from sysdate) order by s.sale\_date desc) where rownum <=5 order by employee\_name

)

LOOP

DBMS\_OUTPUT.PUT\_LINE('On date:'||v\_prod.sale\_date||' customer:'||v\_prod.customer\_name ||' employee:'||v\_prod.employee\_name

||' Name:'||v\_prod.product\_name

|| ' Published on:' || v\_prod.year\_published || ' price:' || v\_prod.price || ' Genre:' || v\_prod.genre\_name ||

' Company:' || v\_prod.company\_name || ' Performed by:' || v\_prod.performer\_name || ' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE QueryCustomer(

in\_name IN customer.customer\_name%TYPE

) IS

BEGIN

FOR v\_prod IN (select i.quantity, p.product\_name, t.p\_type\_name, s.sale\_date, pe.performer\_name, c.company\_name from item i

join product p on i.product\_id = p.product\_id join sale s on i.sale\_id = s.sale\_id join p\_type t

on p.p\_type\_id = t.p\_type\_id join customer c on s.customer\_id = c.customer\_id join performer pe on p.performer\_id = pe.performer\_id

join company c on p.company\_id = c.company\_id where lower(c.customer\_name) like lower(in\_name)

order by t.p\_type\_name, s.sale\_date)

LOOP

DBMS\_OUTPUT.PUT\_LINE('On date:'||v\_prod.sale\_date||' Quantity:'||v\_prod.quantity||' Name:'||v\_prod.product\_name ||

' Company:' || v\_prod.company\_name || ' Performed by:' || v\_prod.performer\_name || ' Type:' || v\_prod.p\_type\_name);

END LOOP;

END;

CREATE OR REPLACE PROCEDURE QuerySalesInRange(

in\_start IN sale.sale\_date%TYPE,

in\_end IN sale.sale\_date%TYPE

) IS

BEGIN

FOR v\_prod IN (select i.quantity, p.product\_name, t.p\_type\_name, c.company\_name, pe.performer\_name, s.sale\_date, c.customer\_name from item i

join product p on i.product\_id = p.product\_id join sale s on i.sale\_id = s.sale\_id join p\_type t

on p.p\_type\_id = t.p\_type\_id join customer c on s.customer\_id = c.customer\_id join performer pe on p.performer\_id = pe.performer\_id

join company c on p.company\_id = c.company\_id where s.sale\_date between in\_start and in\_end

order by c.customer\_name, s.sale\_date)

LOOP

DBMS\_OUTPUT.PUT\_LINE('On date:'||v\_prod.sale\_date||' customer:'||v\_prod.customer\_name||' Quantity:'|| v\_prod.quantity

|| ' Name:'||v\_prod.product\_name ||

' Company:' || v\_prod.company\_name || ' Performed by:' || v\_prod.performer\_name || ' Type:' || v\_prod.p\_type\_name);

END LOOP;

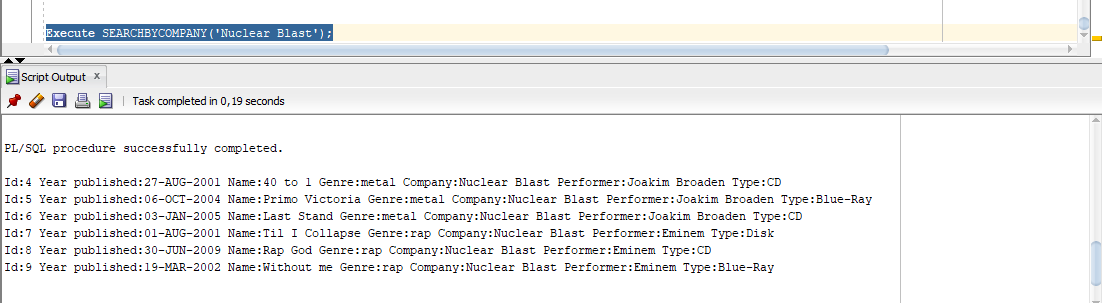
END;

Изпълнение на процедура

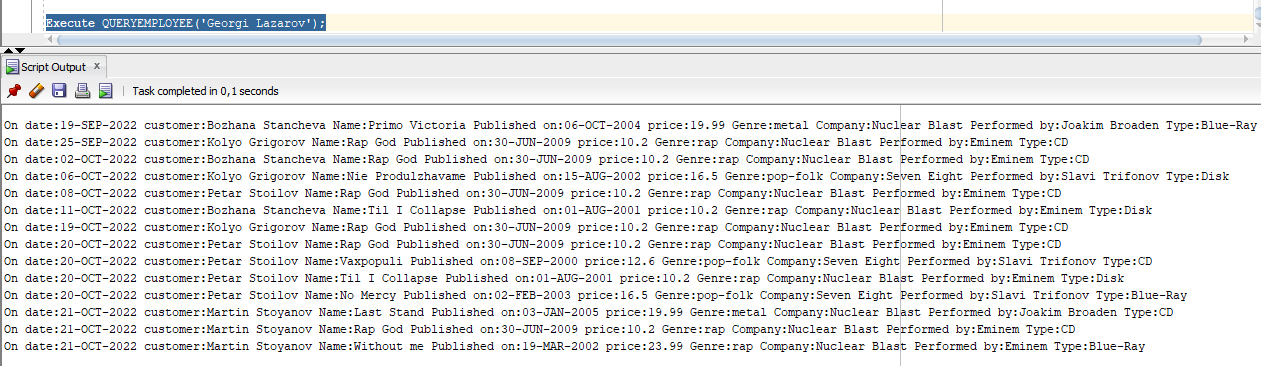
Execute QuerySalesInRange('12-OCT-2022', '20-OCT-2022');

5. Примерна работа

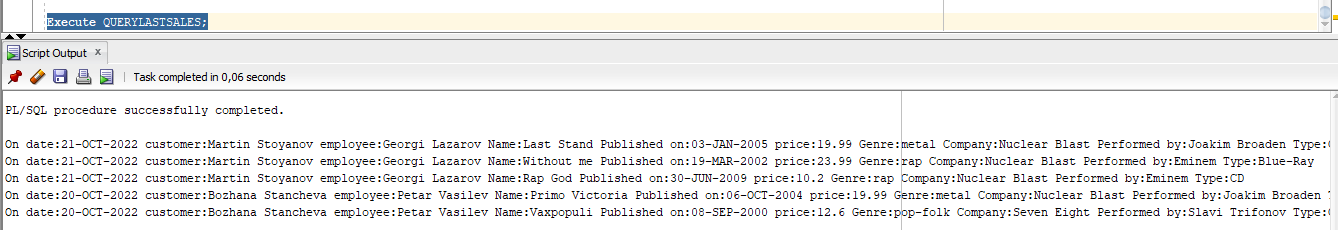
Търсене по компания



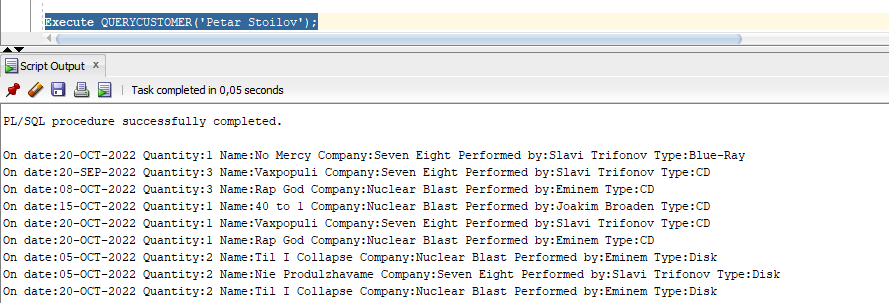
Продажби на служител, подредени по дата



Последите 5 продажби на стоки, издавани в последната година, подредени по служител



Закупени стоки от клиент, подредени по вид и дата



Закупени стоки за период, подредени по клиенти и дати

