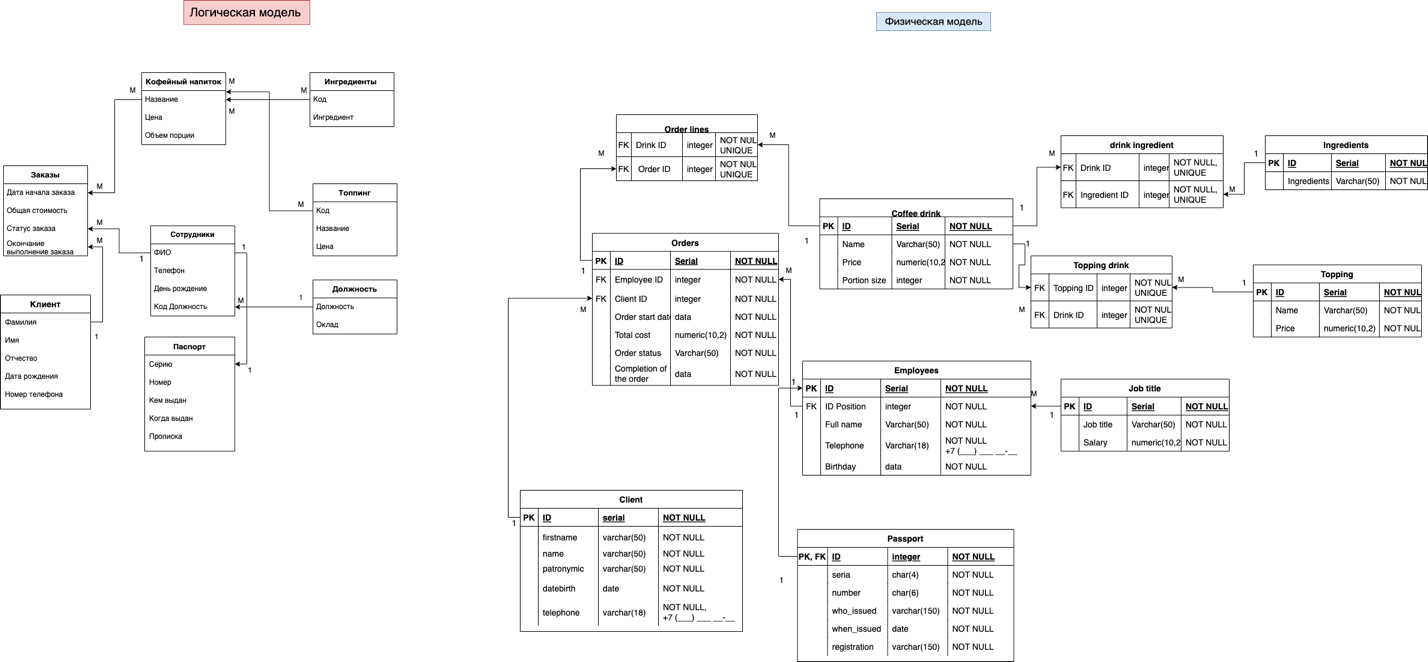
**Промежуточная аттестация**

**Мазуренко Валерия Витальевна**

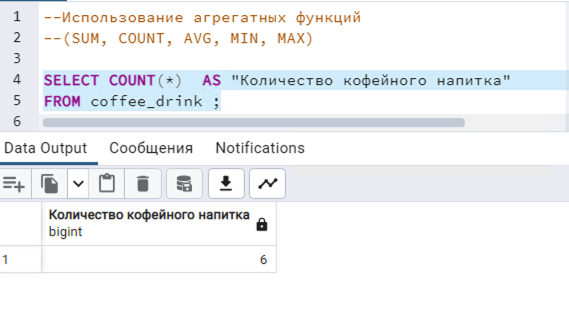
**Проектирование и разработка баз данных на SQL. Промежуточная аттестация**

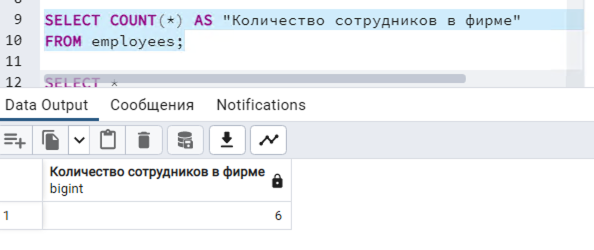
Задание :

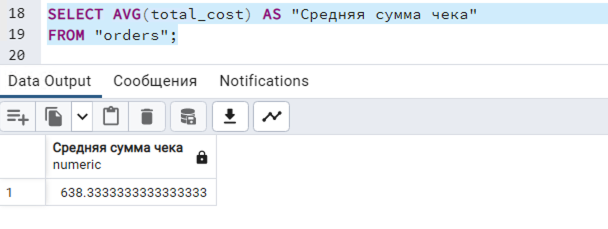
На основе созданной БД из домашнего задания.  
Следует выполнить следующие запросы:  
Обновление нескольких записей из разных таблиц, по разным критериям (UPDATE);  
Удаление нескольких записей из разных таблиц, по разным критериям (DELETE);  
Использование агрегатных функций (SUM, COUNT, AVG, MIN, MAX)  
Сортировка записей по возрастанию и по убыванию;  
Вывод данных из таблиц с помощью оператора WHERE;  
Использование LIMIT и OFFSET;  
Операторы сравнение (>,<,>=,<=,=,!=,<>);  
Логические операторы (AND,OR);  
Использование IN, NOT IN;  
Использование DISTINCT;  
Использование регулярных выражений (LIKE, SIMILAR TO);  
Объединение таблиц (INNER JOIN, RIGHT JOIN, LEFT JOIN);  
Создание красивого вывода при помощи псевдонимов (AS);  
Использование вложенных запросов;  
Создание представления;  
Использование транзакции.

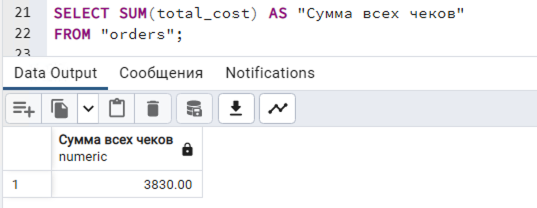
Модель базы данных Кофейня 

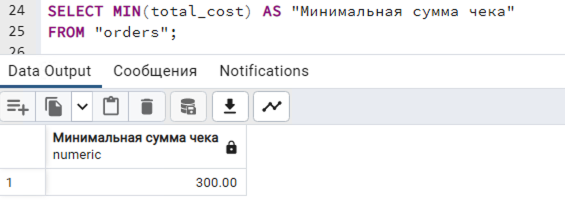
Использование агрегатных функций (SUM, COUNT, AVG, MIN, MAX)

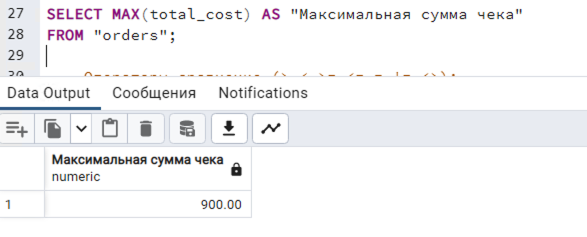


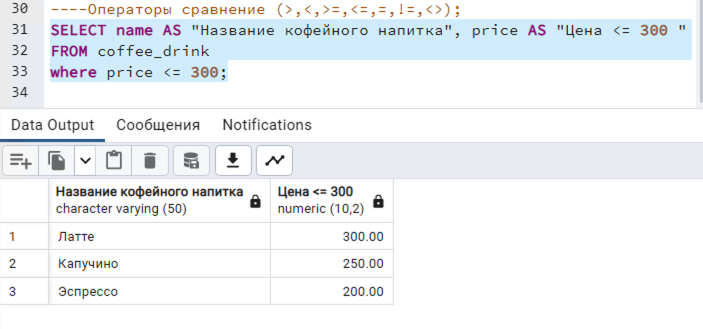


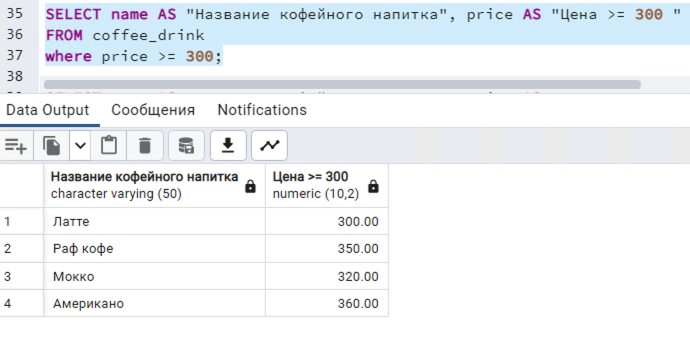


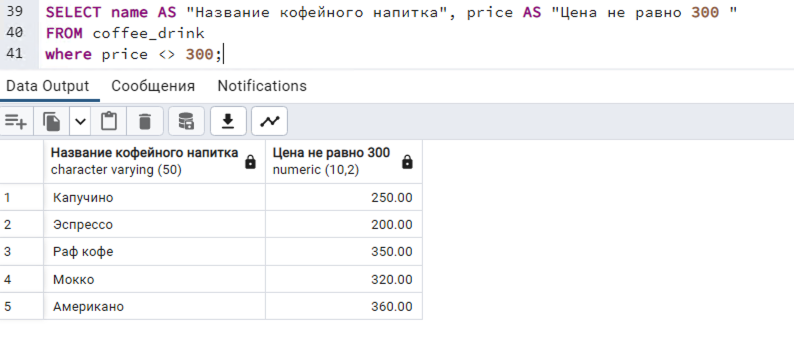


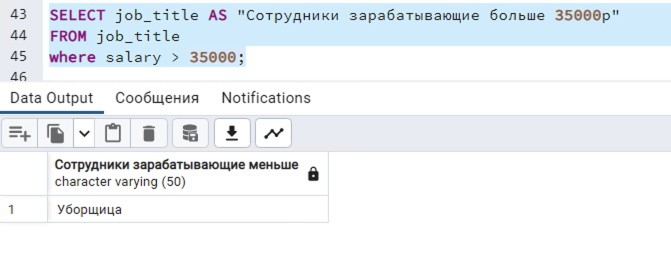


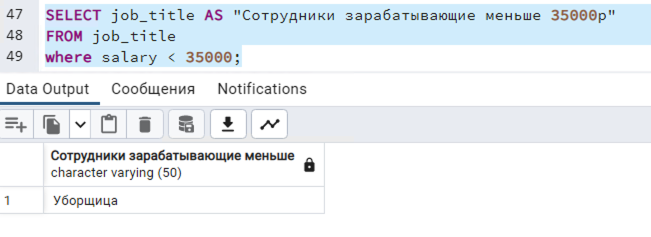


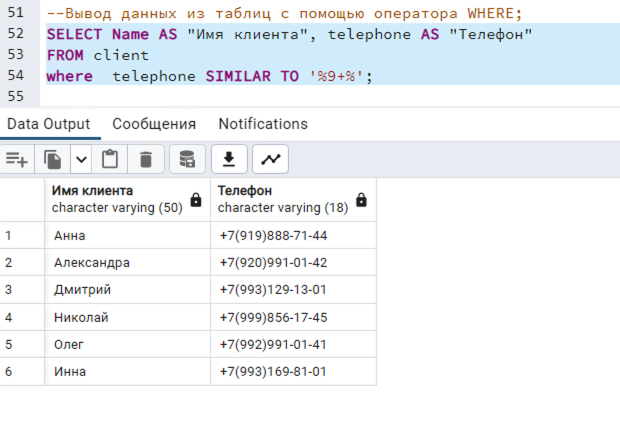
Операторы сравнение (>,<,>=,<=,=,!=,<>)

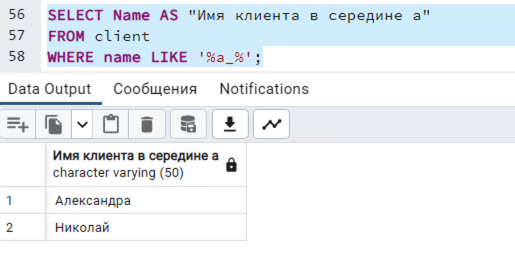






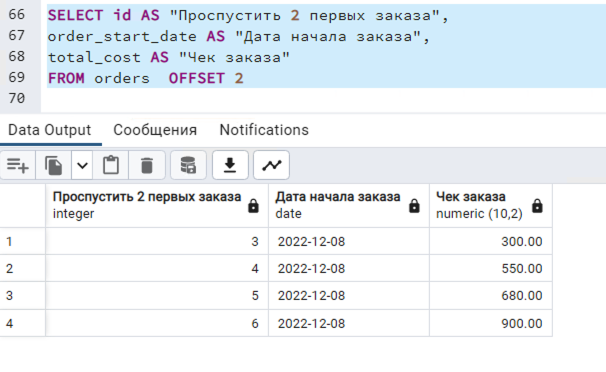


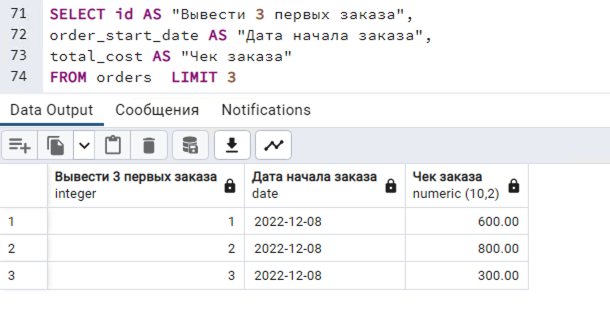
Вывод данных из таблиц с помощью оператора WHERE



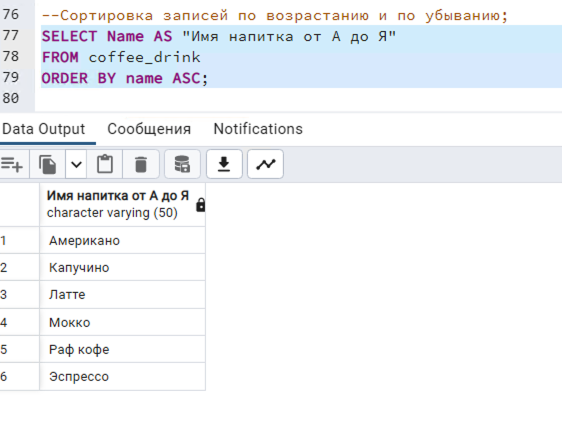
Использование LIMIT и OFFSET

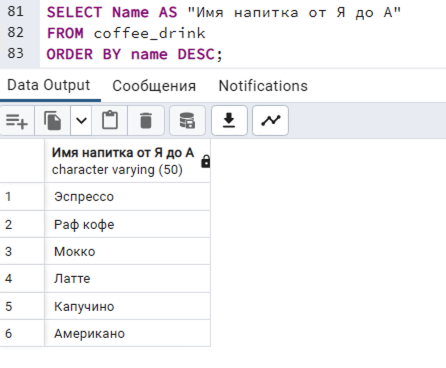


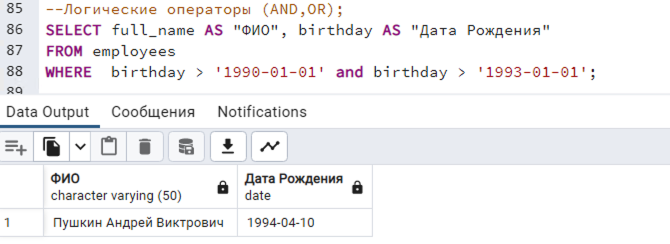


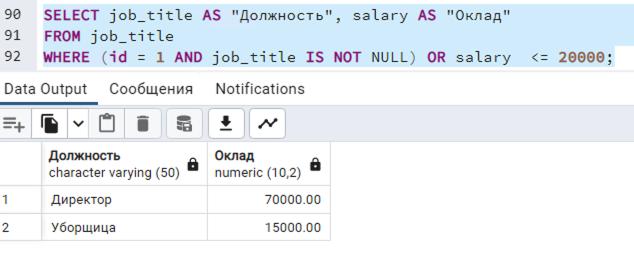


Сортировка записей̆ по возрастанию и по убыванию

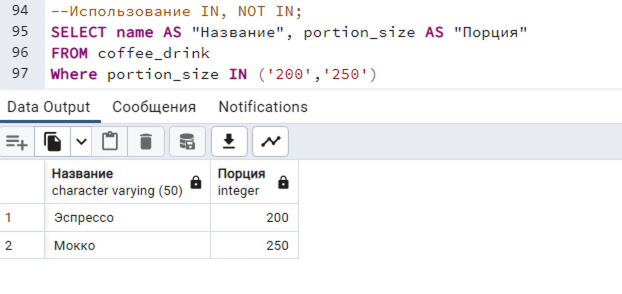


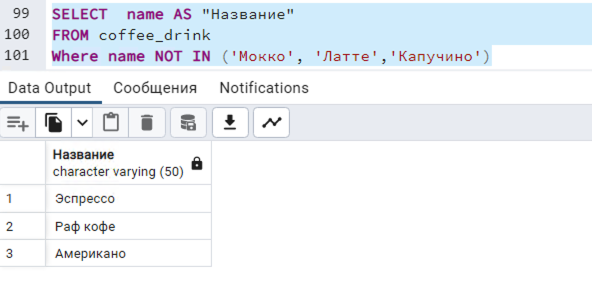


Логические операторы (AND,OR)

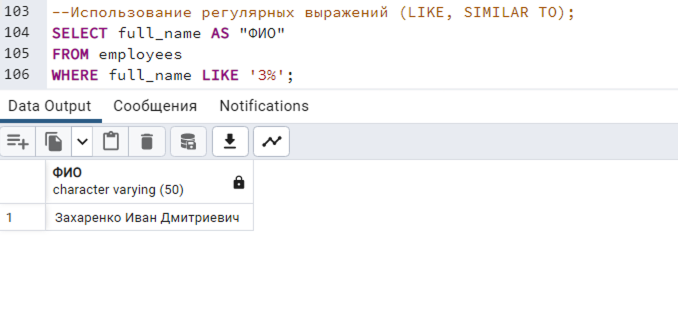


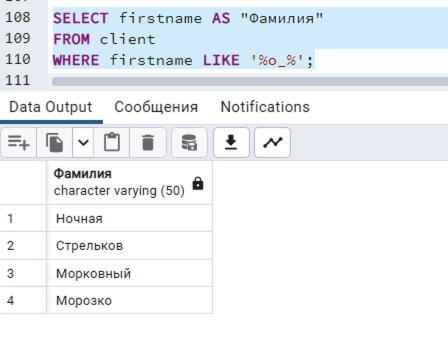
Использование IN, NOT IN

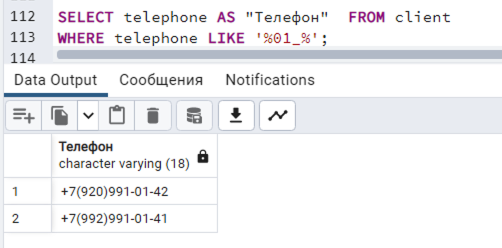


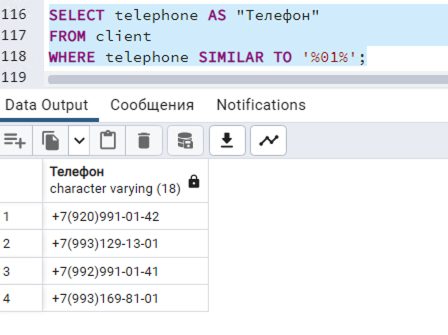


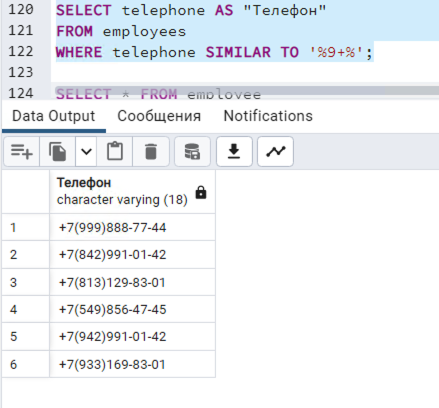
Использование регулярных выражений (LIKE, SIMILAR TO)

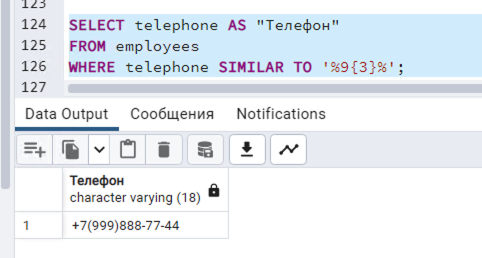




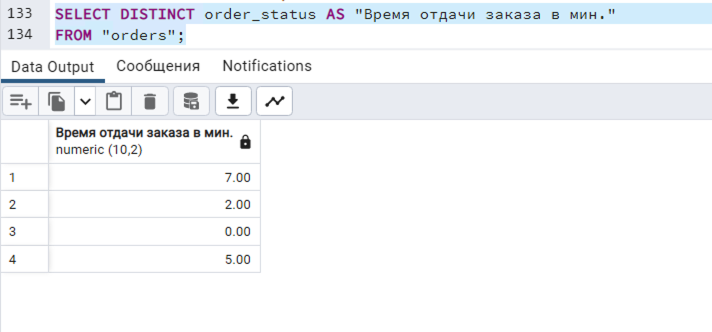






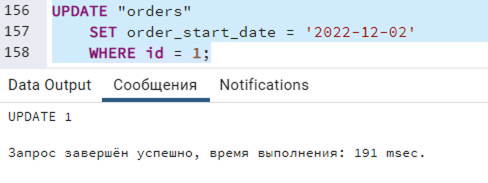


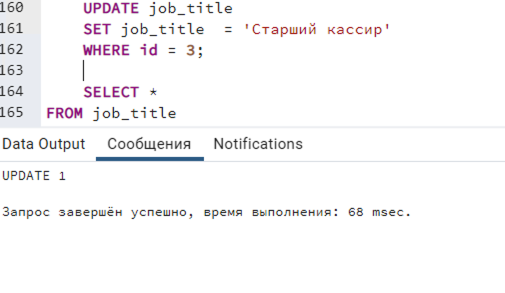
Использование DISTINCT

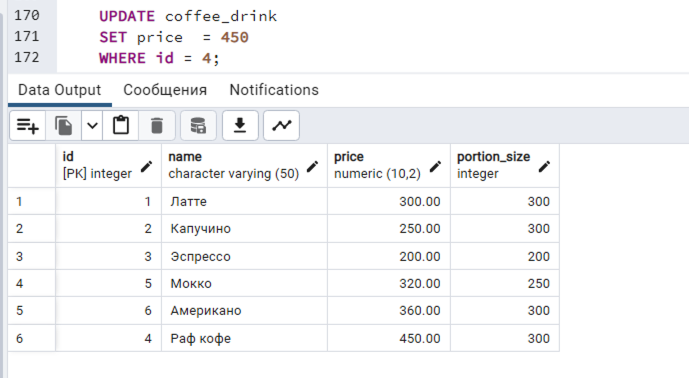


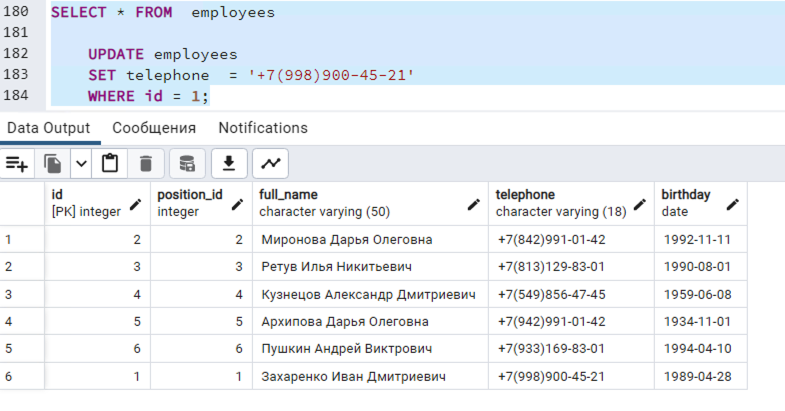


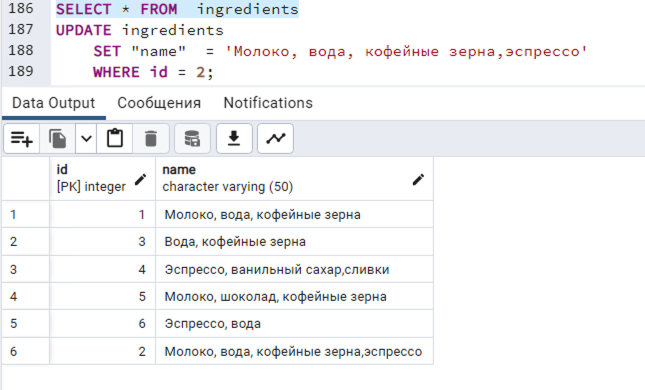
Обновление нескольких записей из разных таблиц, по разным критериям (UPDATE)

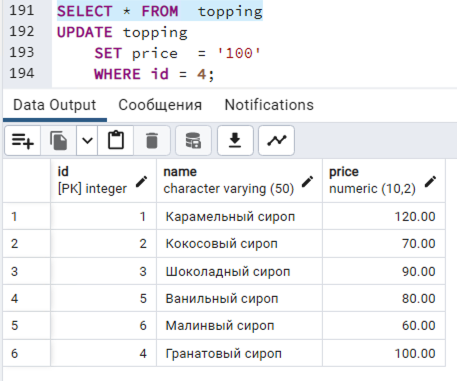


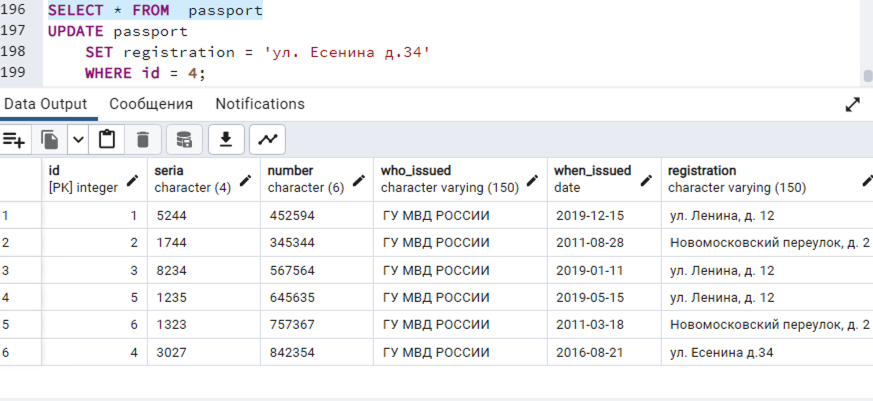




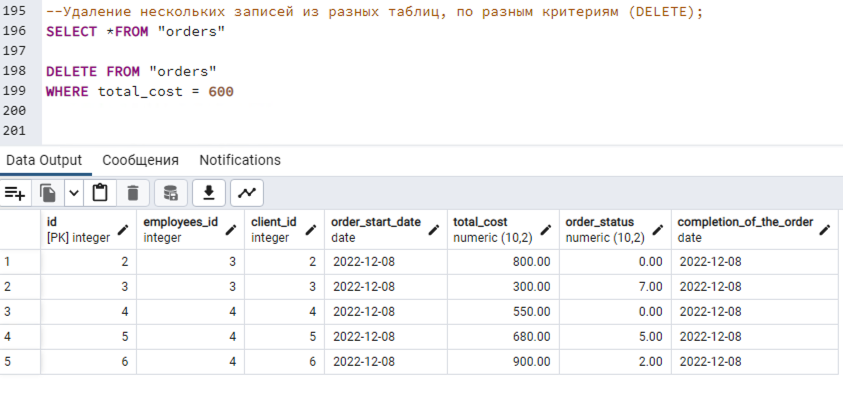


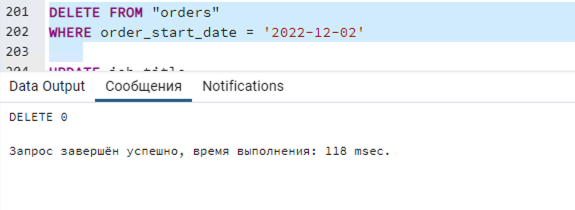


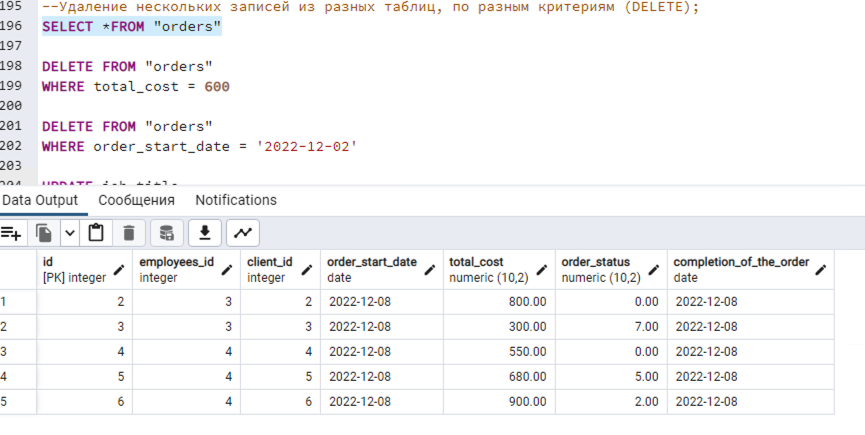


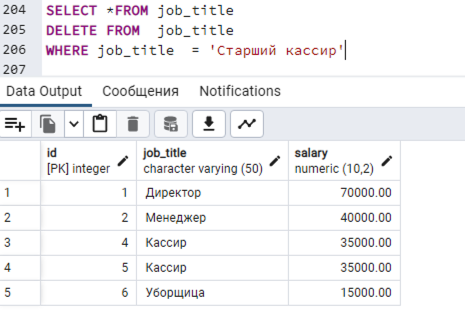


Удаление нескольких записей из разных таблиц, по разным критериям (DELETE)

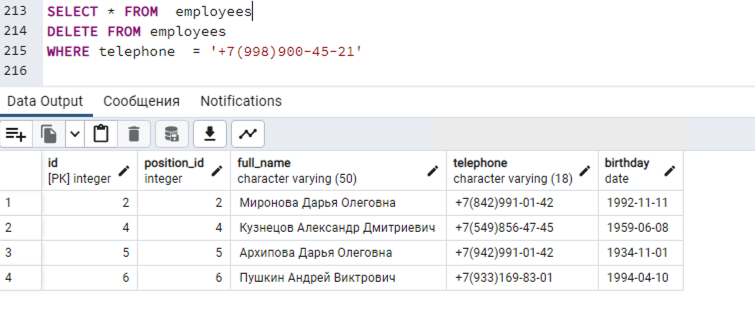


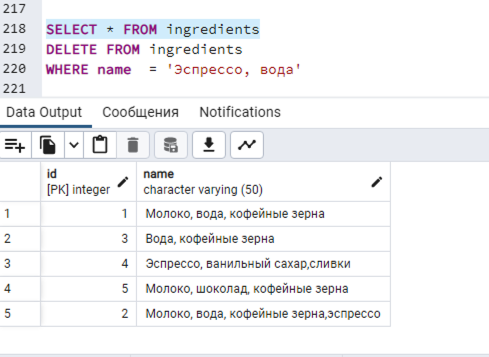


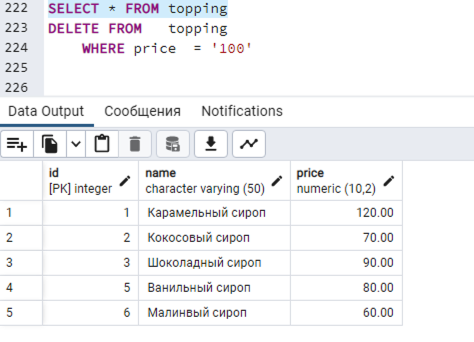


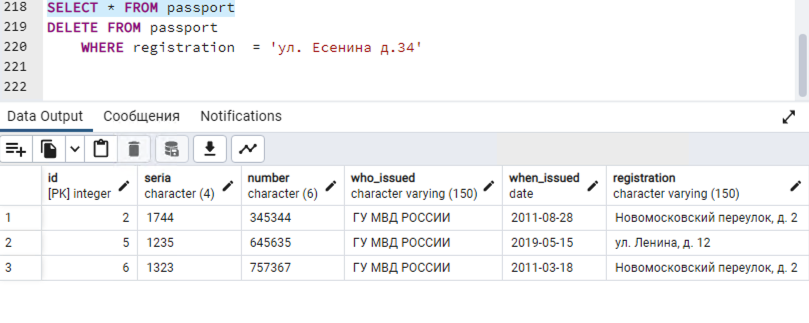




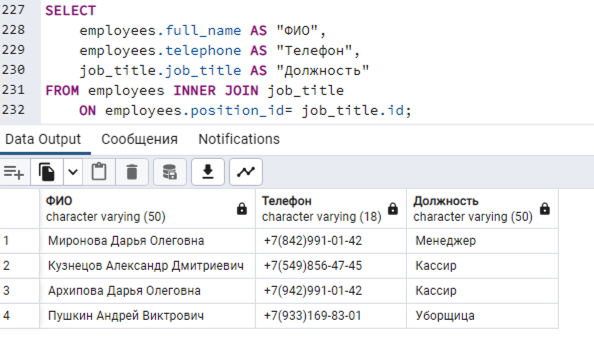


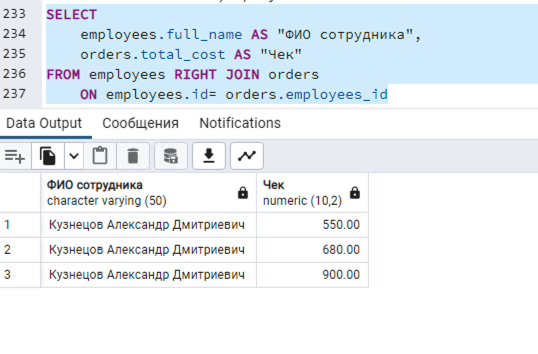






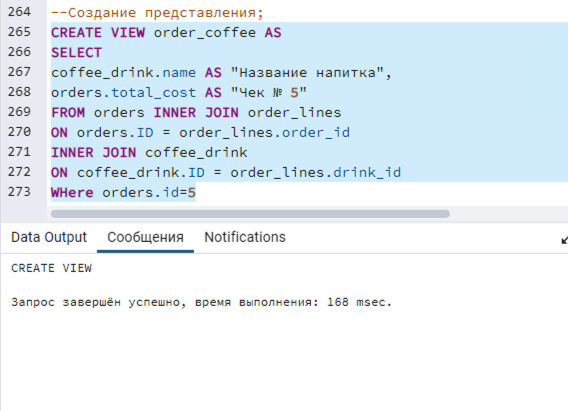
Объединение таблиц (INNER JOIN, RIGHT JOIN, LEFT JOIN)

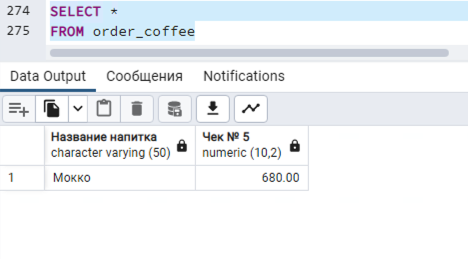




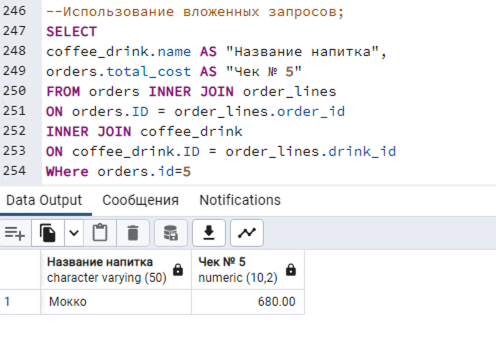


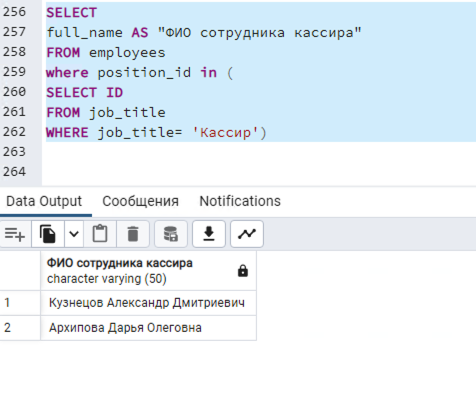
Создание представления

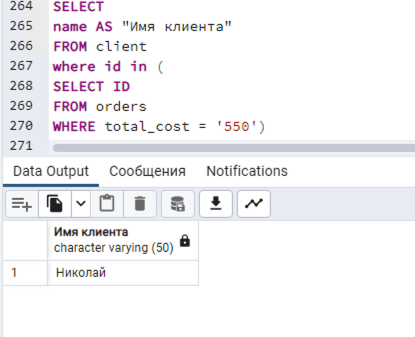




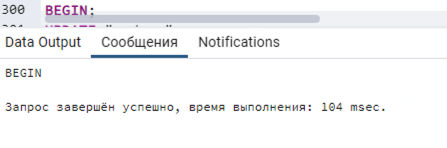
Использование вложенных запросов

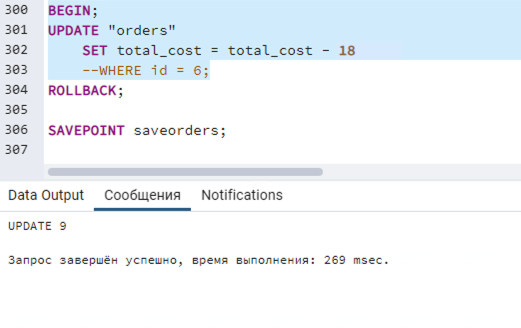


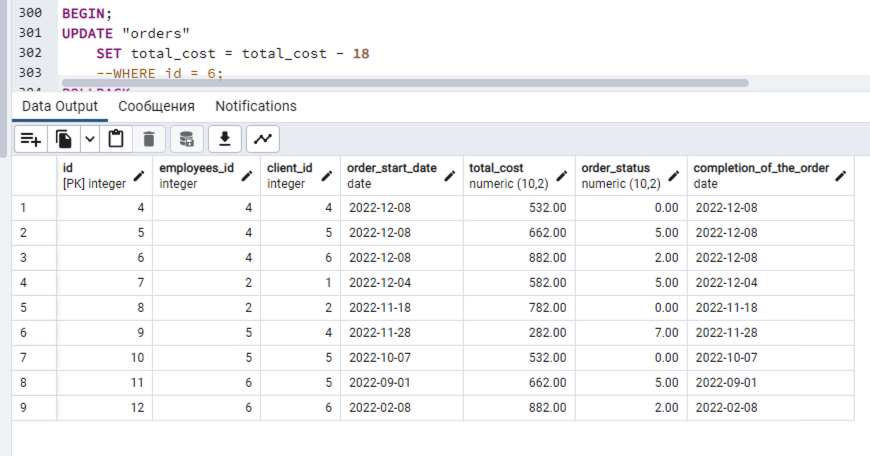


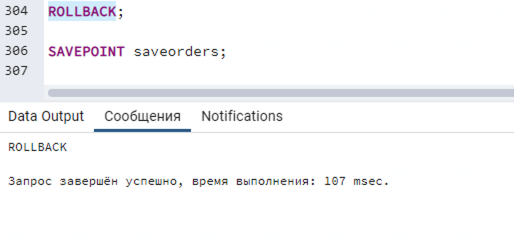


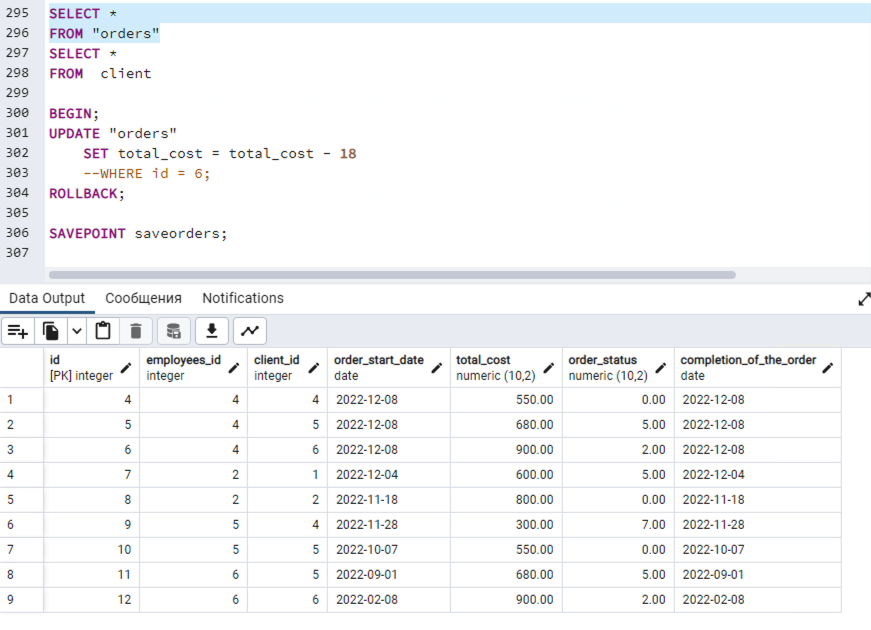
Использование транзакции

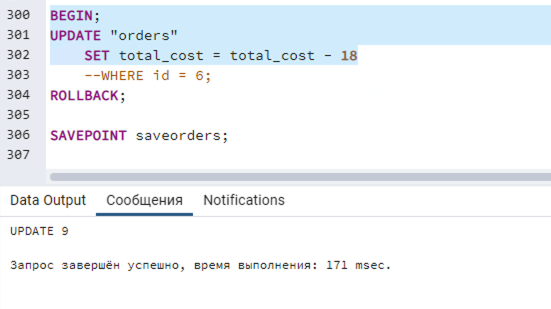


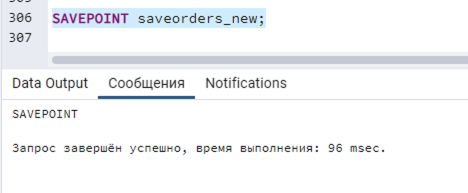


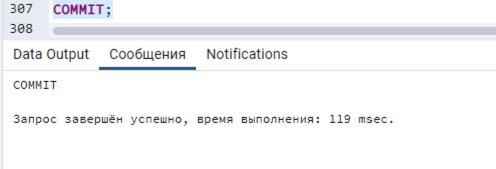












Код программы

--Использование агрегатных функций

--(SUM, COUNT, AVG, MIN, MAX)

SELECT COUNT(\*) AS "Количество кофейного напитка"

FROM coffee\_drink ;

SELECT \* FROM coffee\_drink;

SELECT COUNT(\*) AS "Количество сотрудников в фирме"

FROM employees;

SELECT \*

FROM employees;

SELECT \*

FROM "orders";

SELECT AVG(total\_cost) AS "Средняя сумма чека"

FROM "orders";

SELECT SUM(total\_cost) AS "Сумма всех чеков"

FROM "orders";

SELECT MIN(total\_cost) AS "Минимальная сумма чека"

FROM "orders";

SELECT MAX(total\_cost) AS "Максимальная сумма чека"

FROM "orders";

----Операторы сравнение (>,<,>=,<=,=,!=,<>);

SELECT name AS "Название кофейного напитка", price AS "Цена <= 300 "

FROM coffee\_drink

where price <= 300;

SELECT name AS "Название кофейного напитка", price AS "Цена >= 300 "

FROM coffee\_drink

where price >= 300;

SELECT name AS "Название кофейного напитка", price AS "Цена не равно 300 "

FROM coffee\_drink

where price <> 300;

SELECT job\_title AS "Сотрудники зарабатывающие больше 35000р"

FROM job\_title

where salary > 35000;

SELECT job\_title AS "Сотрудники зарабатывающие меньше 35000р"

FROM job\_title

where salary < 35000;

--Вывод данных из таблиц с помощью оператора WHERE;

SELECT Name AS "Имя клиента", telephone AS "Телефон"

FROM client

where telephone SIMILAR TO '%9+%';

SELECT Name AS "Имя клиента в середине а"

FROM client

WHERE name LIKE '%а\_%';

--Использование LIMIT и OFFSET;

SELECT id AS "Проспустить 1 заказ и вывести 2 первых заказа",

order\_start\_date AS "Дата начала заказа",

total\_cost AS "Чек заказа"

FROM orders OFFSET 1 LIMIT 2 ;

SELECT id AS "Проспустить 2 первых заказа",

order\_start\_date AS "Дата начала заказа",

total\_cost AS "Чек заказа"

FROM orders OFFSET 2

SELECT id AS "Вывести 3 первых заказа",

order\_start\_date AS "Дата начала заказа",

total\_cost AS "Чек заказа"

FROM orders LIMIT 3

--Сортировка записей по возрастанию и по убыванию;

SELECT Name AS "Имя напитка от А до Я"

FROM coffee\_drink

ORDER BY name ASC;

SELECT Name AS "Имя напитка от Я до А"

FROM coffee\_drink

ORDER BY name DESC;

--Логические операторы (AND,OR);

SELECT full\_name AS "ФИО", birthday AS "Дата Рождения"

FROM employees

WHERE birthday > '1990-01-01' and birthday > '1993-01-01';

SELECT job\_title AS "Должность", salary AS "Оклад"

FROM job\_title

WHERE (id = 1 AND job\_title IS NOT NULL) OR salary <= 20000;

--Использование IN, NOT IN;

SELECT name AS "Название", portion\_size AS "Порция"

FROM coffee\_drink

Where portion\_size IN ('200','250')

SELECT name AS "Название"

FROM coffee\_drink

Where name NOT IN ('Мокко', 'Латте','Капучино')

--Использование регулярных выражений (LIKE, SIMILAR TO);

SELECT full\_name AS "ФИО"

FROM employees

WHERE full\_name LIKE 'З%';

SELECT firstname AS "Фамилия"

FROM client

WHERE firstname LIKE '%о\_%';

SELECT telephone AS "Телефон"

FROM client

WHERE telephone LIKE '%01\_%';

SELECT telephone AS "Телефон"

FROM client

WHERE telephone SIMILAR TO '%01%';

SELECT telephone AS "Телефон"

FROM employees

WHERE telephone SIMILAR TO '%9+%';

SELECT telephone AS "Телефон"

FROM employees

WHERE telephone SIMILAR TO '%9{3}%';

SELECT telephone AS "Телефон"

FROM employees

WHERE telephone SIMILAR TO '%(9{3}|42+)%';

--Использование DISTINCT;

SELECT DISTINCT order\_status AS "Время отдачи заказа в мин."

FROM "orders";

SELECT

(COUNT(\*) -

(SELECT COUNT(DISTINCT total\_cost )

FROM "orders"))

AS "Кол-во повторяющихся чеков"

FROM "orders";

--Обновление нескольких записей из разных таблиц, по разным критериям (UPDATE);

SELECT \*

FROM "orders"

BEGIN;

UPDATE "orders"

SET total\_cost = total\_cost - 60

WHERE id = 3;

ROLLBACK;

SAVEPOINT saveorders;

UPDATE "orders"

SET order\_start\_date = '2022-12-02'

WHERE id = 1

SELECT \*FROM job\_title

UPDATE job\_title

SET job\_title = 'Старший кассир'

WHERE id = 3;

UPDATE job\_title

SET price = '450'

WHERE id = 4;

SELECT \* FROM employees

UPDATE employees

SET telephone = '+7(998)900-45-21'

WHERE id = 1;

SELECT \* FROM ingredients

UPDATE ingredients

SET "name" = 'Молоко, вода, кофейные зерна, эспрессо'

WHERE id = 2;

SELECT \* FROM topping

UPDATE topping

SET price = '100'

WHERE id = 4;

SELECT \* FROM passport

UPDATE passport

SET registration = 'ул. Есенина д.34'

WHERE id = 4;

--Удаление нескольких записей из разных таблиц, по разным критериям (DELETE);

SELECT \*FROM "orders"

DELETE FROM "orders"

WHERE total\_cost = 600

DELETE FROM "orders"

WHERE order\_start\_date = '2022-12-02'

SELECT \*FROM job\_title

DELETE FROM job\_title

WHERE job\_title = 'Старший кассир'

SELECT \* FROM coffee\_drink

DELETE FROM coffee\_drink

WHERE price = '450'

SELECT \* FROM employees

DELETE FROM employees

WHERE telephone = '+7(998)900-45-21'

SELECT \* FROM ingredients

DELETE FROM ingredients

WHERE name = 'Эспрессо, вода'

SELECT \* FROM topping

DELETE FROM topping

WHERE price = '100'

SELECT \* FROM passport

DELETE FROM passport

WHERE registration = 'ул. Есенина д.34'

--Объединение таблиц (INNER JOIN, RIGHT JOIN, LEFT JOIN);

SELECT \* FROM employees, job\_title

SELECT

employees.full\_name AS "ФИО",

employees.telephone AS "Телефон",

job\_title.job\_title AS "Должность"

FROM employees INNER JOIN job\_title

ON employees.position\_id= job\_title.id;

SELECT \* FROM orders,employees

SELECT

employees.full\_name AS "ФИО сотрудника",

orders.total\_cost AS "Чек"

FROM employees RIGHT JOIN orders

ON employees.id= orders.

SELECT \* FROM coffee\_drink,ingredients

SELECT

coffee\_drink.name AS "Название напитка",

ingredients.name AS "Состав"

FROM coffee\_drink LEFT JOIN ingredients

ON coffee\_drink.id= ingredients.id

--Использование вложенных запросов;

SELECT

coffee\_drink.name AS "Название напитка",

orders.total\_cost AS "Чек № 5"

FROM orders INNER JOIN order\_lines

ON orders.ID = order\_lines.order\_id

INNER JOIN coffee\_drink

ON coffee\_drink.ID = order\_lines.drink\_id

WHere orders.id=5

SELECT

full\_name AS "ФИО сотрудника кассира"

FROM employees

where position\_id in (

SELECT ID

FROM job\_title

WHERE job\_title= 'Кассир')

SELECT

name AS "Имя клиента"

FROM client

where id in (

SELECT ID

FROM orders

WHERE total\_cost = '550')

--Создание представления;

CREATE VIEW order\_coffee AS

SELECT

coffee\_drink.name AS "Название напитка",

orders.total\_cost AS "Чек № 5"

FROM orders INNER JOIN order\_lines

ON orders.ID = order\_lines.order\_id

INNER JOIN coffee\_drink

ON coffee\_drink.ID = order\_lines.drink\_id

WHere orders.id=5

SELECT \*

FROM order\_coffee

v

--Использование транзакции.

INSERT INTO "orders"(id ,employees\_ID,client\_ID,order\_start\_date,total\_cost,order\_status,completion\_of\_the\_order)

VALUES

(7,2,1,'2022-12-4','600','5','2022-12-4'),

(8,2,2,'2022-11-18','800','0','2022-11-18'),

(9,5,4,'2022-11-28','300','7','2022-11-28'),

(10,5,5,'2022-10-7','550','0','2022-10-7'),

(11,6,5,'2022-09-1','680','5','2022-09-1'),

(12,6,6,'2022-02-8','900','2','2022-02-8');

SELECT \*

FROM "orders"

SELECT \*

FROM client

BEGIN;

UPDATE "orders"

SET total\_cost = total\_cost - 18

--WHERE id = 6;

ROLLBACK;

SAVEPOINT saveorders\_new;

COMMIT;