

Промежуточная аттестация Модуль 2 «Клиент-серверные приложения на Java. Инструменты разработки»

Формулировка задания:

Задача 1. Установка PostgreSQL.

1. Рекомендована 12 версия PostgreSQL. Установить PostgreSQL на локальную машину <https://postgrespro.ru/docs/postgrespro/14/binary-installation-on-windows>
2. Настроить административную панель. Задать логин и пароль администратора.
3. Проверить подключение к PostgreSQL из IDE IntelliJ Idea <https://www.jetbrains.com/help/idea/postgresql.html#connect-to-postgresql-database>

Упрощение задачи 1. Реализовать схему в онлайн-редакторе. Например, в редакторе <https://sqliteonline.com/>. Обратите внимание, что реализовать требуется с помощью PostgreSQL.

Задача 2. Создать схему базы данных.

1. Создать schema.sql файл, который содержит базу данных из 3 таблиц. Назначение таблиц и поля указаны ниже:

Товар

- id
- описание
- стоимость
- количество

Покупатель

-
- id
 - имя/фамилия

Заказ

- id-товара (внешний ключ)
- id-заказчика (внешний ключ)
- дата заказа
- количество товаров

В базе данных должны быть реализованы следующие действия:

1) Создание таблицы с заданными полями. Создания таблиц дополнить комментарием с описанием таблицы. Учесть, что файл может быть запущен несколько раз подряд (команда CREATE TABLE IF NOT EXISTS).

2) Заполнение таблиц данными. Достаточно заполнить 10 строк в каждой таблице. Обратить внимание на связи между Товар-Заказ и Покупатель-Заказ.

2. Написать sql-запросы к созданным таблицам. Запросы требуется написать на чтение данных, изменение данных, удаление данных. Sql запросы поместить в файл test-queries.sql

DBViewer 25.2.2 - «postgres» schema

Файл Редактирование Избранная Поиск Редактор SQL База данных Окно Справка

SQL Commit Rollback T Auto

postgres attestation02@postgres

База данных: postgres schema

General

Connections

postgres localhost:5432

Базы данных

postgres

Схемы

attestation02

Таблицы

buyer 16K

order 16K

product 16K

Внешние таблицы

Представления

Мат. представления

Индексы

Функции

Последовательности

Типы данных

Агрегатные функции

postgres17

public

Событийные триггеры

Расширения

Хранилища

Системные объекты

Роли

Администрирование

Системные объекты

Bookmarks

Dashboards

Diagrams

Scripts

```
create table if not exists "product"
(
    id bigserial primary key,
    description varchar,
    cost int,
    count int
);

create table if not exists "buyer"
(
    id bigserial primary key,
    name varchar
);

create table if not exists "order"
(
    product_id int references "product"(id),
    buyer_id int references "buyer"(id),
    order_date timestamp,
    count_orders int
);

comment on table "product" is 'Product table';
comment on table "buyer" is 'Buyers table';
comment on table "order" is 'Orders table';
```

Статистика 1

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Tue Oct 07 21:48:28 EET 2025
Finish time	Tue Oct 07 21:48:28 EET 2025
Query	comment on table "order" is 'Orders table'

Схема: attestation02

EET ru

```
<postgres> schema <postgres> data X
insert into product (description, cost, count)
values ('Гимнастический ролик', 668, 2), ('Скакалка', 314, 12), ('Страховочная система', 430, 12),
('Карабин', 524, 5), ('Эспандер', 250, 4), ('Турник', 2230, 2), ('Шведская стенка', 7500, 1),
('Волейбольный мяч', 1200, 3), ('Теннисный мячик', 300, 6), ('Теннисная ракетка', 2700, 2);

insert into buyer (name)
values ('Максим'), ('Юлия'), ('Ева'), ('Алиса'), ('Леонид'),
('Александр'), ('Светлана'), ('Ирина'), ('Анна'), ('Дмитрий');

insert into "order" (product_id, buyer_id, order_date, count_orders)
values (2, 2, '2025-05-10 18:00:00', 2), (1, 4, '2025-05-19 10:23:54', 1), (10, 10, '2025-06-15 12:17:00', 1),
(9, 6, '2025-06-19 17:24:12', 3), (5, 2, '2025-06-25 15:30:54', 1), (8, 5, '2025-07-17 17:27:54', 1),
(6, 1, '2025-08-08 12:21:23', 1), (8, 3, '2025-08-14 18:21:43', 2), (3, 9, '2025-09-17 15:15:34', 1),
(4, 9, '2025-09-18 15:36:41', 4);
```

buyer 1			
select * from buyer Введите SQL выражение чтобы отфильтровать результаты			
Таблица Текст Запись		123 id	A-Z name
	1	1	Максим
	2	2	Юлия
	3	3	Ева
	4	4	Алиса
	5	5	Леонид
	6	6	Александр
	7	7	Светлана
	8	8	Ирина
	9	9	Анна
	10	10	Дмитрий
Обновить Save Cancel Экспорт данных ... 200 10 10 строк полу			

product 1

Таблица

Текст

Запись

123 id

A-Z description

123 cost

123 count

1	1	Гимнастический ролик	668	2
2	2	Скакалка	314	12
3	3	Страховочная система	430	12
4	4	Карабин	524	5
5	5	Эспандер	250	4
6	6	Турник	2 230	2
7	7	Шведская стенка	7 500	1
8	8	Волейбольный мяч	1 200	3
9	9	Теннисный мячик	300	6
10	10	Теннисная ракетка	2 700	2

Обновить

Save

Cancel

+

-

<

>

↶

↷

Экспорт данных ...

200

10

10 строк получ

EET

ru

Запись

Инт. вставка

10 : 69 : 586

Sel: 0 | 0

order 1

select * from "order"

Введите SQL выражение чтобы отфильтровать результаты

product_id

buyer_id

order_date

count_orders

1	2	2	2025-05-10 18:00:00.000	2
2	1	4	2025-05-19 10:23:54.000	1
3	10	10	2025-06-15 12:17:00.000	1
4	9	6	2025-06-19 17:24:12.000	3
5	5	2	2025-06-25 15:30:54.000	1
6	8	5	2025-07-17 17:27:54.000	1
7	6	1	2025-08-08 12:21:23.000	1
8	8	3	2025-08-14 18:21:43.000	2
9	3	9	2025-09-17 15:15:34.000	1
10	4	9	2025-09-18 15:36:41.000	4

Значение

Edit value:
8

product

Type part of dictionary value to search

Значение	Описание
5	Эспандер
6	Турник
7	Шведская стенка
8	Волейбольный мяч

Обновить

Save

Cancel

Экспорт данных ...

200

10

10 строк получено - 0.0s (0.0s получ., 2025-10-07 в 22:31:04)

order 1

select * from "order"

Введите SQL выражение чтобы отфильтровать результаты

123 product_id

123 buyer_id

order_date

123 count_orders

1	2	2	2025-05-10 18:00:00.000	2
2	1	4	2025-05-19 10:23:54.000	1
3	10	10	2025-06-15 12:17:00.000	1
4	9	6	2025-06-19 17:24:12.000	3
5	5	2	2025-06-25 15:30:54.000	1
6	8	5	2025-07-17 17:27:54.000	1
7	6	1	2025-08-08 12:21:23.000	1
8	8	3	2025-08-14 18:21:43.000	2
9	3	9	2025-09-17 15:15:34.000	1
10	4	9	2025-09-18 15:36:41.000	4

Значение

Edit value:
6

buyer

Type part of dictionary value to search

Значение	Описание
3	Ева
4	Алиса
5	Леонид
6	Александр

Обновить

Save

Cancel

Экспорт данных ...

200

10

10 строк получено - 0.0s (0.0s получ., 2025-10-07 в 22:31:04

EET

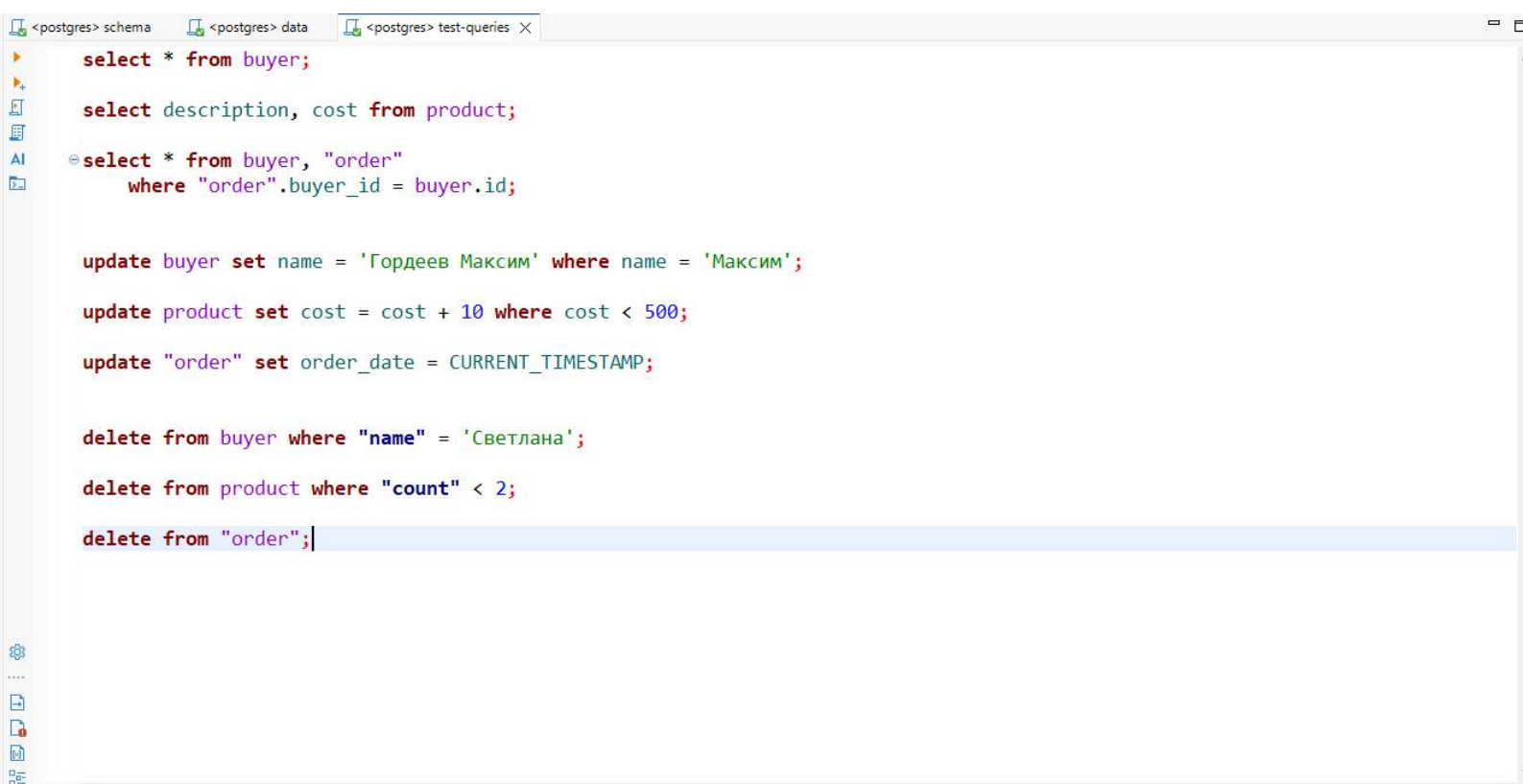
ru

Запись

Инт. вставка

16 : 12 : 988

Sel: 0 | 0



The screenshot shows a web-based interface for PostgreSQL queries. At the top, there are three tabs: "<postgres> schema", "<postgres> data", and "<postgres> test-queries". The "test-queries" tab is active. On the left side, there is a vertical toolbar with icons for running queries, saving, and other database management functions. The main area contains a list of SQL queries, with the last one selected and highlighted in blue. The queries are as follows:

```
select * from buyer;

select description, cost from product;

select * from buyer, "order"
  where "order".buyer_id = buyer.id;

update buyer set name = 'Гордеев Максим' where name = 'Максим';

update product set cost = cost + 10 where cost < 500;

update "order" set order_date = CURRENT_TIMESTAMP;

delete from buyer where "name" = 'Светлана';

delete from product where "count" < 2;

delete from "order";
```

- JavaDevHomeworks D:\JavaProject\JavaDevHomeworks
 - .idea
 - out
 - src
 - attestation
 - attestation01
 - attestation02
 - .mvn
 - src
 - main
 - java
 - resources
 - data.sql
 - schema.sql
 - test-queries.sql
 - test
 - pom.xml
 - homeworks
 - homework01
 - homework02
 - homework03
 - homework04
 - homework05
 - homework06
 - homework07
 - homework08
 - homework09
 - homework10
 - homework11
 - homework12
 - homework13
 - homework14
 - homework15
 - homework16
 - homework17
 - Main
 - test
 - .gitignore
 - JavaDevHomeworks.iml
 - External Libraries
 - Scratches and Consoles