МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВГТУ», ВГТУ)

Факультет информационных технологий и компьютерной безопасности

# Кафедра автоматизированных и вычислительных систем

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №8

По дисциплине: «Системы хранения и обработки данных»

Тема: «Разработка физической структуры базы данных»

Выполнил работу студент группы змИИВТ-241: Матыкина А.В.

подпись, дата

Принял: Короленко В.В.

подпись, дата

Воронеж 2024

# **Учебная задача**

Сформировать SQL-запросы для создания таблиц в СУБД Postgres, соответствующих логической структуре базы данных, разработанной в предыдущей лабораторной работе

-- SQL-запрос для создания отношения (таблицы) «Вид\_издания»

CREATE TABLE IF NOT EXISTS Вид\_издания (

id\_вида\_издания SERIAL PRIMARY KEY, -- SERIAL-- автоинкрементирующееся числовое значение -- ПК таблицы

Название\_вида\_издания VARCHAR(255),

);

-- SQL-запрос для создания отношения (таблицы) «Издание»

CREATE TABLE IF NOT EXISTS Издание (

id\_издания SERIAL PRIMARY KEY, -- SERIAL-- автоинкрементирующееся числовое значение -- ПК таблицы

Вид\_издания int REFERENCES Вид\_издания (id\_вида\_издания) ,

Название\_издания VARCHAR(255),

Цена DECIMAL

);

-- SQL-запрос для создания отношения (таблицы) «Подписчик»

CREATE TABLE IF NOT EXISTS Подписчик (

Паспорт\_подписчика SERIAL PRIMARY KEY, -- SERIAL-- автоинкрементирующееся числовое значение -- ПК таблицы

ФИО VARCHAR(255),

Улица VARCHAR(255),

Номер\_дома int,

Номер\_кв int,

);

-- SQL-запрос для создания отношения (таблицы) «Доставка»

CREATE TABLE IF NOT EXISTS Доставка (

id\_доставки SERIAL PRIMARY KEY, -- SERIAL-- автоинкрементирующееся числовое значение -- ПК таблицы

Название\_вида\_издания VARCHAR(255),

);

-- SQL-запрос для создания отношения (таблицы) «Подписка»

CREATE TABLE IF NOT EXISTS Подписка (

id\_подписки SERIAL PRIMARY KEY, -- SERIAL-- автоинкрементирующееся числовое значение -- ПК таблицы

Паспорт\_подписчика int REFERENCES Подписчик (Паспорт\_подписчика),

id\_издания int REFERENCES Издание (id\_издания),

Стартовая\_дата DATE,

Кол-во\_месяцев int,

id\_доставки int REFERENCES Доставка (id\_доставки),

Цена\_доставки DECIMAL,

Цена\_подписки DECIMAL,

Итоговая\_цена DECIMAL

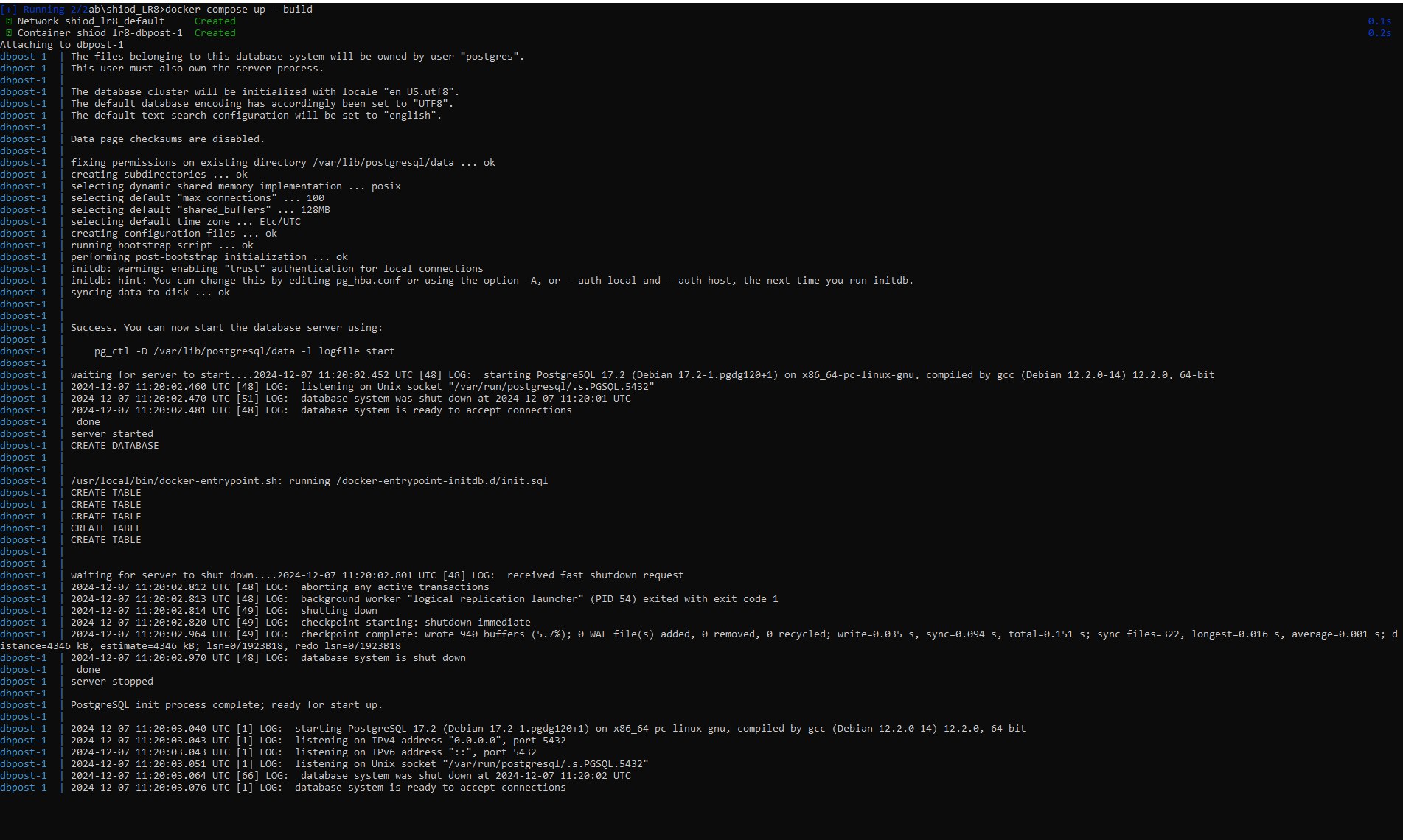
);

Данный запрос состоит из пяти команд создания БД (CREATE TABLE IF NOT EXISTS <*имя\_БД*>), ключевые слова IF NOT EXISTS служат для того, чтобы создавать таблицы только если данные таблицы еще не существуют. Далее в скобках через запятую перечисляются названия столбцов, которые будут принадлежать таблице, после названия столбца указывается тип данных в этом столбце, далее может быть указана роль столбца (первичный ключ или внешний ключ)

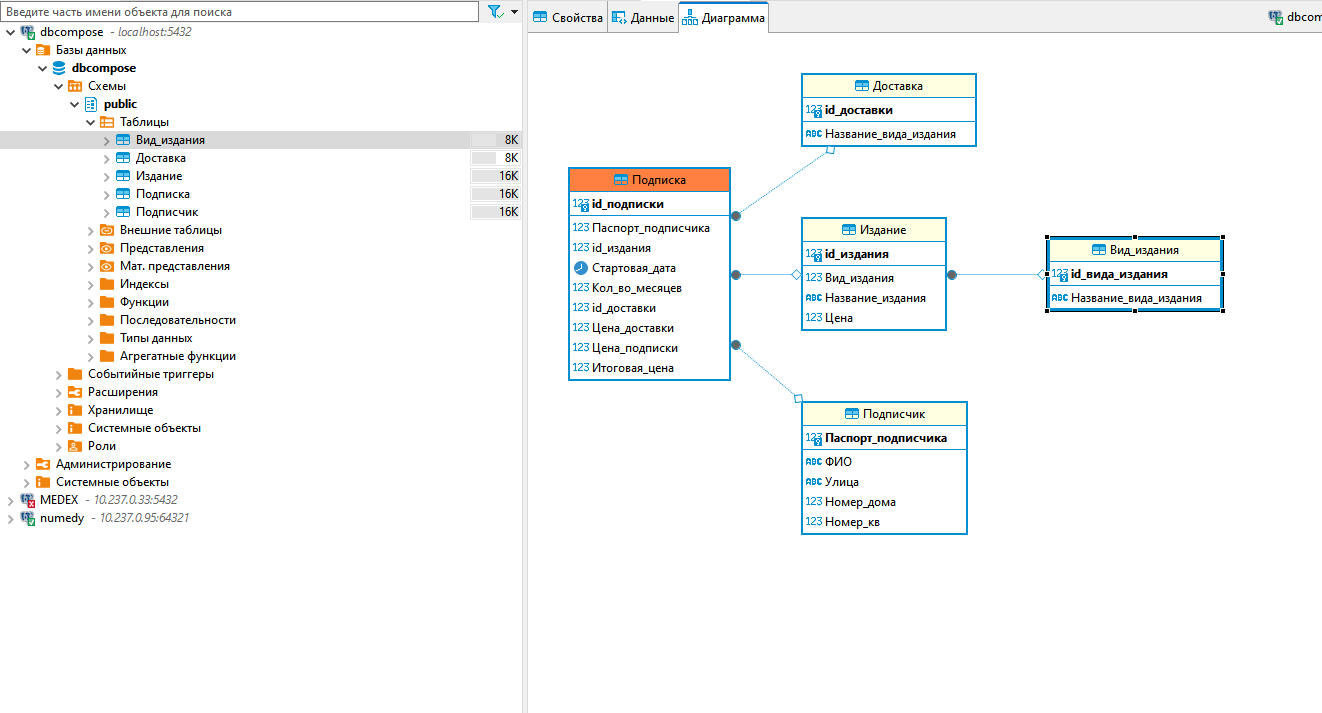
Указание, что столбец является первичным ключом: PRIMARY KEY. Перед этими ключевыми словами можно указать ключевое слово SERIAL.Оно означает, что столбец будет заполняться автоматически – автоматически сгенерированным числовым значением.

Указание, что столбец является внешним ключом: после типа данных столбца указывается ключевое слово REFERENCES далее требуется указать, на что ссылается внешний ключ. Для этого после REFERENCES пишется *<Название таблицы>(<Название столбца>).*

Использую команду docker-compose up –build для создания докер контейнера, с помощью технологии compose.



Подключаюсь к созданной БД в DBeaver для проверки корректности созданной БД:



# **Контрольные вопросы**

1. Особенности PostgreSQL.

Система управления базами данных Postgres (она же PostgreSQL) является свободной объектно-реляционной СУБД

PostgreSQL можно применять в среде клиент-сервер

PostgreSQL подходит для работы в любой операционной системе: Linux, macOS, Windows.

PostgreSQL поддерживает много разных типов и структур данных, в том числе сетевые адреса, данные в текстовом формате JSON и геометрические данные для координат геопозиций. Все эти форматы можно хранить и обрабатывать в СУБД.

При работе с PostgreSQL можно создавать собственные типы данных, их называют пользовательскими

Размер базы данных в PostgreSQL не ограничен и зависит от того, сколько свободной памяти есть в месте хранения: на сервере, локальном компьютере или в облаке.

1. Что такое DDL?

DDL, или Data Definition Language — это группа команд, которые используются для создания и изменения структуры объектов базы данных: таблиц, представлений, схем и индексов.

Наиболее известные команды SQL DDL — CREATE, ALTER, DROP

1. Что такое DML?

DCL, или Data Control Language — это команды SQL, которые используют для предоставления и отзыва привилегий пользователя базы данных. При этом пользователь не может откатить изменения. Наиболее известные команды: GRANT и REVOKE

1. Что такое DCL?

TCL, или Transaction Control Language — одни из наиболее популярных команд SQL. Их используют для обеспечения согласованности базы данных и для управления транзакциями.

Транзакция — это набор SQL-запросов, выполняемых над данными, которые объединены в атомарную секцию. Это значит, что промежуточные результаты операции не видны для других конкурирующих транзакций — и вся секция будет либо выполнена, либо полностью отменена в случае ошибки.

Примеры команд: BEGIN/COMMIT, ROLLBACK.

1. С какой команды, как правило, начинается запрос?

SELECT

1. Какая команда используется для создания базы данных?

CREATE DATABASE

1. Какая команда используется для создания таблицы в базе данных?

CREATE TABLE

1. · SELECT \* FROM employee WHERE mod(mod(sal, 23), 5) = 0;  
   · SELECT \* FROM employee WHERE sal IN (1500, 3000);
2. · SELECT \* FROM employee ORDER BY dept\_id NULLS FIRST;
3. · '1', 'a', 'char'
4. · Select список полей from список таблиц having условия группировки group by поля для группировки

· Select список полей from список таблиц order by поля для сортировки union Select список полей from список таблиц order by поля для сортировки

1. · Select avg (advance), sum (sales) from titles group by type