**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

кафедра физики и прикладной математики

**ОТЧЕТ**

**ПО КУРСОВОЙ РАБОТЕ**

**по дисциплине «Базы данных»**

**на тему**

**Разработка и администрирование базы данных «Приём   
пациентов»**

Студента Дыгановой Дарьи Дмитриевны

Направления 02.03.02 «Фундаментальная информатика и информационные технологии»

Группы ИТ-123

*Руководитель*:

ассистент кафедры ФиПМ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.А. Тройникова

Владимир 2025

ВЛАДИМИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Зав. кафедрой

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

по дисциплине «Базы данных»

Выдано студенту Дыгановой Дарье Дмитриевне группы ИТ-123

Направление подготовки бакалавров 02.03.02 «Фундаментальная информатика и информационные технологии»

1. Тема работы «Разработка и администрирование базы данных «Приём пациентов»

2. Срок сдачи законченной работы 02.06.2025

3. Исходные данные к курсовой работе PostgreSQL, схема базы данных

4. Содержание работы

Часть 1. Подготовка базы данных;

Часть 2. Резервное копирование и восстановление;

Часть 3. Импорт и экспорт данных;

Часть 4. Управление доступом.

5. Содержание отчета: 1. Введение. 2. Подготовка базы данных. 3. Резервное копирование и восстановление. 4. Импорт и экспорт данных. 5. Управление доступом. 6. Заключение.

6. Исходные библиографические источники

1. pgAdmin Documentation: Import/Export Data [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.pgadmin.org/docs/pgadmin4/latest/import\_export\_data.html. – Текст: электронный.

2. Моргунов, Е. П. PostgreSQL. Основы языка SQL : учебное пособие / Е. П. Моргунов. — Москва: ДМК Пресс, 2021. — 324 с. — ISBN 978-5-93700-123-7.

3. Астахова, И. Ф. SQL в примерах и задачах: учебное пособие / И. Ф. Астахова, А. П. Толстобров, В. М. Мельников. — Санкт-Петербург: БХВ-Петербург, 2022. — 416 с. — ISBN 978-5-9775-6789-4.

4. PostgreSQL Documentation: Backup and Restore [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/backup.html. – Текст: электронный.

5. PostgreSQL Documentation: Database Roles [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/user-manag.html. – Текст: электронный.

6. PostgreSQL Documentation: SQL Commands – COPY [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/sql-copy.html. – Текст: электронный.

7. pgAdmin Documentation: Managing Roles [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.pgadmin.org/docs/pgadmin4/latest/roles.html. – Текст: электронный.

Дата выдачи задания         03.02.2025

Руководитель Тройникова Е.А.

Задание приняли к исполнению Дыганова Д.Д.

Содержание

[Введение 4](#_Toc208948460)

[1 ПОДГОТОВКА БАЗЫ ДАННЫХ 5](#_Toc208948461)

[1.1 Скрипт создания структуры базы данных 5](#_Toc208948462)

[1.2 Заполнение таблиц данными 6](#_Toc208948463)

[1.3 Удаление данных из всех таблиц 8](#_Toc208948464)

[2 РЕЗЕРВНОЕ КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ 9](#_Toc208948465)

[2.1 Резервное копирование и восстановление базы данных 9](#_Toc208948466)

[2.2 Резервное копирование всех баз данных текущего сервера 10](#_Toc208948467)

[4 ИМПОРТ И ЭКСПОРТ ДАННЫХ 12](#_Toc208948468)

[4.1 Импорт данных из csv-файла 12](#_Toc208948469)

[4.2 Импорт и экспорт данных с помощью командной строки 13](#_Toc208948470)

[5 УПРАВЛЕНИЕ ДОСТУПОМ 16](#_Toc208948471)

[5.1 Sql-скрипт создания ролей 16](#_Toc208948472)

[5.2 Sql-скрипт создания пользователей 16](#_Toc208948473)

[5.3 Работа с правами доступа пользователей 18](#_Toc208948474)

[Заключение 21](#_Toc208948475)

[Список использованных источников 22](#_Toc208948476)

# Введение

**Цель курсовой работы**

Разработка и администрирование базы данных медицинского учреждения с реализацией механизмов резервного копирования, управления доступом и обработки данных.

**Задачи работы**

1. **Проектирование структуры базы данных**:

- Создание таблиц (врачи, пациенты, приемы, диагнозы, лечения)

- Настройка связей между таблицами (внешние ключи)

1. **Наполнение базы данных тестовыми данными**

- Вставка данных во все таблицы (минимум 10 записей в каждой)

1. **Резервное копирование и восстановление**

- Реализация скриптов для pg\_dump и pg\_restore

- Экспорт/импорт через интерфейс DBeaver/PgAdmin

1. **Управление доступом**

- Создание ролей (doctor\_role, patient\_role, admin\_role)

- Назначение прав (GRANT, REVOKE)

- Автоматическая генерация пользователей из данных таблиц

1. **Тестирование работоспособности**

- Проверка корректности резервного копирования

- Демонстрация ограничений доступа для разных ролей

# ПОДГОТОВКА БАЗЫ ДАННЫХ

## Скрипт создания структуры базы данных

**create** **table** doctor (

doctor\_id serial **primary** **key**,

name **varchar**(100) **not** **null**,

speciality **varchar**(50) **not** **null**,

phone\_number **varchar**(20)

);

**create** **table** patient (

patient\_id serial **primary** **key**,

name **varchar**(100) **not** **null**,

birth\_date **date**,

adress **varchar**(200),

phone\_number **varchar**(20)

);

**create** **table** appointment (

appointment\_id serial **primary** **key**,

doctor\_id **int** **references** doctor(doctor\_id),

patient\_id **int** **references** patient(patient\_id),

date **timestamp** **not** **null**,

duration **int** **not** **null** -- в минутах

);

**create** **table** disease (

disease\_id serial **primary** **key**,

name **varchar**(100) **not** **null**,

symptoms **text**

);

**create** **table** treatment (

treatment\_id serial **primary** **key**,

name **varchar**(100) **not** **null**,

effect **text**,

indications **text**,

contraindications **text**,

side\_effects **text**,

usage **text**

);

**create** **table** diagnosis (

diagnosis\_id serial **primary** **key**,

appointment\_id **int** **references** appointment(appointment\_id),

disease\_id **int** **references** disease(disease\_id),

notes **text**

);

**create** **table** perscribed\_treatment (

diagnosis\_id **int** **references** diagnosis(diagnosis\_id),

treatment\_id **int** **references** treatment(treatment\_id),

**primary** **key** (diagnosis\_id, treatment\_id)

);

**create** **table** possible\_treatment (

diagnosis\_id **int** **references** diagnosis(diagnosis\_id),

treatment\_id **int** **references** treatment(treatment\_id),

**primary** **key** (diagnosis\_id, treatment\_id)

);

## Заполнение таблиц данными

**insert** **into** doctor (name, speciality, phone\_number) **values**

('Иванов Иван Иванович', 'Терапевт', '+79161234567'),

('Петрова Анна Сергеевна', 'Кардиолог', '+79162345678'),

('Сидоров Михаил Петрович', 'Невролог', '+79163456789'),

('Кузнецова Елена Владимировна', 'Офтальмолог', '+79164567890'),

('Смирнов Алексей Дмитриевич', 'Хирург', '+79165678901'),

('Федорова Ольга Николаевна', 'Эндокринолог', '+79166789012'),

('Николаев Денис Андреевич', 'Оториноларинголог', '+79167890123'),

('Алексеева Татьяна Викторовна', 'Гастроэнтеролог', '+79168901234'),

('Григорьев Павел Сергеевич', 'Уролог', '+79169012345'),

('Дмитриева Светлана Игоревна', 'Гинеколог', '+79160123456');

**insert** **into** patient (name, birth\_date, adress, phone\_number) **values**

('Соколов Андрей Васильевич', '1985-05-15', 'ул. Ленина, 10, кв. 5', '+79151112233'),

('Михайлова Екатерина Олеговна', '1990-07-22', 'ул. Пушкина, 15, кв. 12', '+79152223344'),

('Волков Дмитрий Игоревич', '1978-03-30', 'ул. Гагарина, 20, кв. 7', '+79153334455'),

('Ковалева Марина Александровна', '1995-11-05', 'ул. Мира, 5, кв. 3', '+79154445566'),

('Белов Сергей Петрович', '1982-09-18', 'ул. Советская, 25, кв. 9', '+79155556677'),

('Зайцева Анна Дмитриевна', '1973-12-10', 'ул. Садовая, 30, кв. 15', '+79156667788'),

('Павлов Игорь Владимирович', '1992-02-28', 'ул. Центральная, 12, кв. 8', '+79157778899'),

('Семенова Людмила Викторовна', '1965-08-14', 'ул. Лесная, 7, кв. 4', '+79158889900'),

('Фролов Артем Сергеевич', '1988-06-25', 'ул. Молодежная, 18, кв. 11', '+79159990011'),

('Орлова Наталья Ивановна', '1970-04-03', 'ул. Школьная, 3, кв. 6', '+79151001122');

**insert** **into** disease (name, symptoms) **values**

('Грипп', 'Высокая температура, головная боль, слабость, боль в мышцах'),

('Гипертония', 'Повышенное давление, головная боль, головокружение'),

('Гастрит', 'Боль в желудке, изжога, тошнота'),

('Сахарный диабет', 'Жажда, частое мочеиспускание, усталость'),

('Бронхит', 'Кашель, одышка, хрипы в груди'),

('Остеохондроз', 'Боль в спине, ограничение подвижности'),

('Конъюнктивит', 'Покраснение глаз, зуд, слезотечение'),

('Ангина', 'Боль в горле, высокая температура, слабость'),

('Цистит', 'Частое мочеиспускание, боль внизу живота'),

('Аллергия', 'Зуд, покраснение кожи, чихание');

**insert** **into** treatment (name, effect, indications, contraindications, side\_effects, usage) **values**

('Парацетамол', 'Жаропонижающее, обезболивающее', 'Грипп, простуда, головная боль', 'Беременность, заболевания печени', 'Тошнота, аллергия', 'По 1 таблетке 3 раза в день'),

('Амоксициллин', 'Антибактериальное', 'Ангина, бронхит, цистит', 'Аллергия на пенициллины', 'Диарея, сыпь', 'По 500 мг 3 раза в день'),

('Эналаприл', 'Снижает давление', 'Гипертония', 'Беременность, почечная недостаточность', 'Головокружение, кашель', 'По 5-10 мг 1 раз в день'),

('Омепразол', 'Снижает кислотность', 'Гастрит, язва', 'Дети до 12 лет', 'Головная боль, диарея', 'По 20 мг 1 раз в день'),

('Метформин', 'Снижает сахар в крови', 'Сахарный диабет', 'Почечная недостаточность', 'Тошнота, диарея', 'По 500 мг 2 раза в день'),

('Амброксол', 'Отхаркивающее', 'Бронхит', 'Первый триместр беременности', 'Тошнота, аллергия', 'По 30 мг 3 раза в день'),

('Диклофенак', 'Обезболивающее, противовоспалительное', 'Остеохондроз, артрит', 'Язва желудка, астма', 'Боль в желудке, головокружение', 'По 50 мг 2 раза в день'),

('Левомицетин', 'Антибактериальное', 'Конъюнктивит', 'Дети до 1 года', 'Жжение в глазах', 'По 1 капле 3 раза в день'),

('Фурагин', 'Антибактериальное', 'Цистит', 'Беременность, почечная недостаточность', 'Тошнота, головокружение', 'По 50 мг 4 раза в день'),

('Лоратадин', 'Антигистаминное', 'Аллергия', 'Дети до 2 лет', 'Сонливость, сухость во рту', 'По 10 мг 1 раз в день');

**insert** **into** appointment (doctor\_id, patient\_id, date, duration) **values**

(1, 1, '2023-01-10 09:00:00', 30),

(2, 2, '2023-01-11 10:00:00', 20),

(3, 3, '2023-01-12 11:00:00', 40),

(4, 4, '2023-01-13 12:00:00', 25),

(5, 5, '2023-01-14 13:00:00', 35),

(6, 6, '2023-01-15 14:00:00', 30),

(7, 7, '2023-01-16 15:00:00', 20),

(8, 8, '2023-01-17 16:00:00', 40),

(9, 9, '2023-01-18 17:00:00', 25),

(10, 10, '2023-01-19 18:00:00', 35);

**insert** **into** diagnosis (appointment\_id, disease\_id, notes) **values**

(1, 1, 'Тяжелая форма'),

(2, 2, 'Первая стадия'),

(3, 3, 'Обострение'),

(4, 4, 'Тип 2'),

(5, 5, 'Острый бронхит'),

(6, 6, 'Поясничный отдел'),

(7, 7, 'Бактериальный'),

(8, 8, 'Гнойная ангина'),

(9, 9, 'Острый цистит'),

(10, 10, 'Сезонная аллергия');

**insert** **into** perscribed\_treatment (diagnosis\_id, treatment\_id) **values**

(1, 1),

(2, 3),

(3, 4),

(4, 5),

(5, 6),

(6, 7),

(7, 8),

(8, 2),

(9, 9),

(10, 10);

**insert** **into** possible\_treatment (diagnosis\_id, treatment\_id) **values**

(1, 2),

(2, 4),

(3, 5),

(4, 6),

(5, 7),

(6, 8),

(7, 9),

(8, 10),

(9, 1),

(10, 3);

## Удаление данных из всех таблиц

**delete** **from** possible\_treatment;

**delete** **from** perscribed\_treatment;

**delete** **from** diagnosis;

**delete** **from** appointment;

**delete** **from** doctor;

**delete** **from** patient;

**delete** **from** disease;

**delete** **from** treatment;

# РЕЗЕРВНОЕ КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ

## Резервное копирование и восстановление базы данных

Используется интерфейс DBeaver для копирования и восстановления

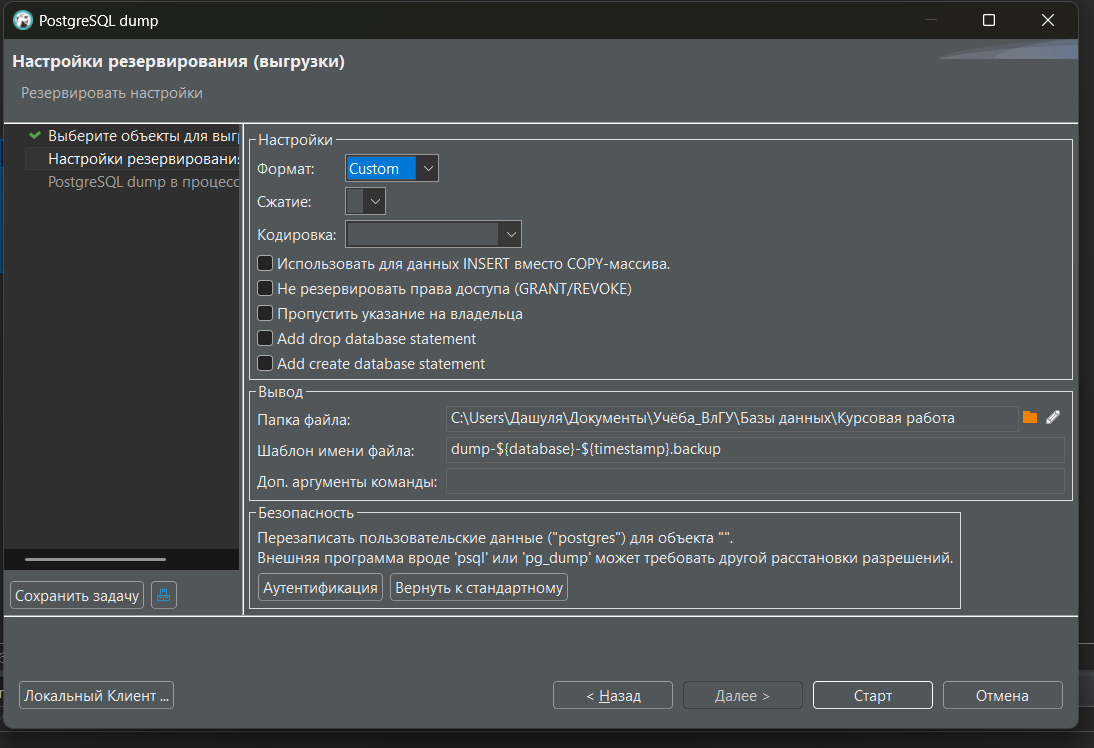


Рисунок 1 – Выгружаем данные в резервную копию

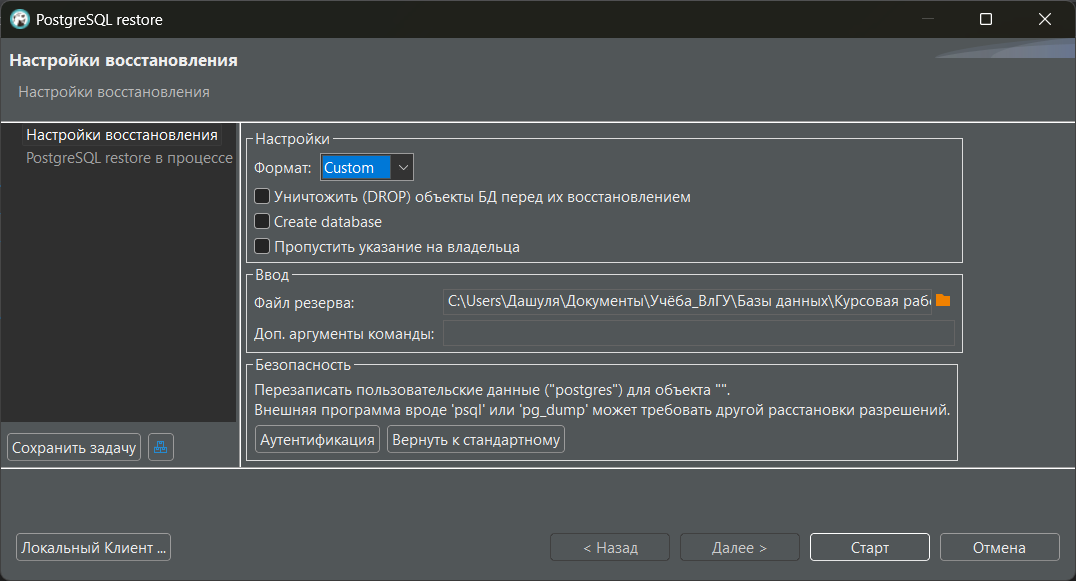


Рисунок 2 – Восстанавливаем данные в базу данных test1

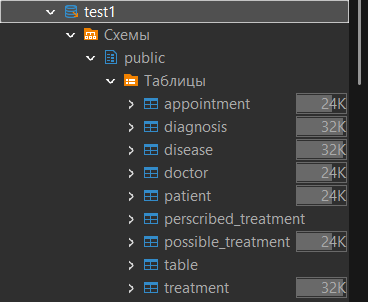


Рисунок 3 – Резервное копирование прошло успешно

## Резервное копирование всех баз данных текущего сервера

Скрипт для командной строки (резервное копирование):

@echo off

setlocal

REM Настройки

set PG\_PATH="C:\Program Files\PostgreSQL\15\bin"

set BACKUP\_DIR=C:\PG\_Backups

set DATE=%date:~-4%-%date:~3,2%-%date:~0,2%

set PG\_USER=postgres

set PG\_PASSWORD=postgres

REM Создаем папку для бэкапов

if not exist "%BACKUP\_DIR%" mkdir "%BACKUP\_DIR%"

REM Экспорт всех баз (кроме служебных)

%PG\_PATH%\pg\_dumpall.exe -U %PG\_USER% --exclude-database="template\*" --exclude-database="postgres" -f "%BACKUP\_DIR%\all\_databases\_%DATE%.sql"

echo Резервное копирование завершено. Файл: %BACKUP\_DIR%\all\_databases\_%DATE%.sql

endlocal

Скрипт для командной строки (восстановление):

"C:\Program Files\PostgreSQL\15\bin\pg\_restore.exe" -U postgres -d новая\_база "C:\PG\_Backups\имя\_базы.backup"

# ИМПОРТ И ЭКСПОРТ ДАННЫХ

## Импорт данных из csv-файла

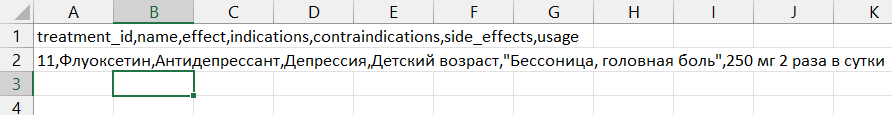


Рисунок 4 – Скриншот таблицы .csv, откуда импортируются файлы

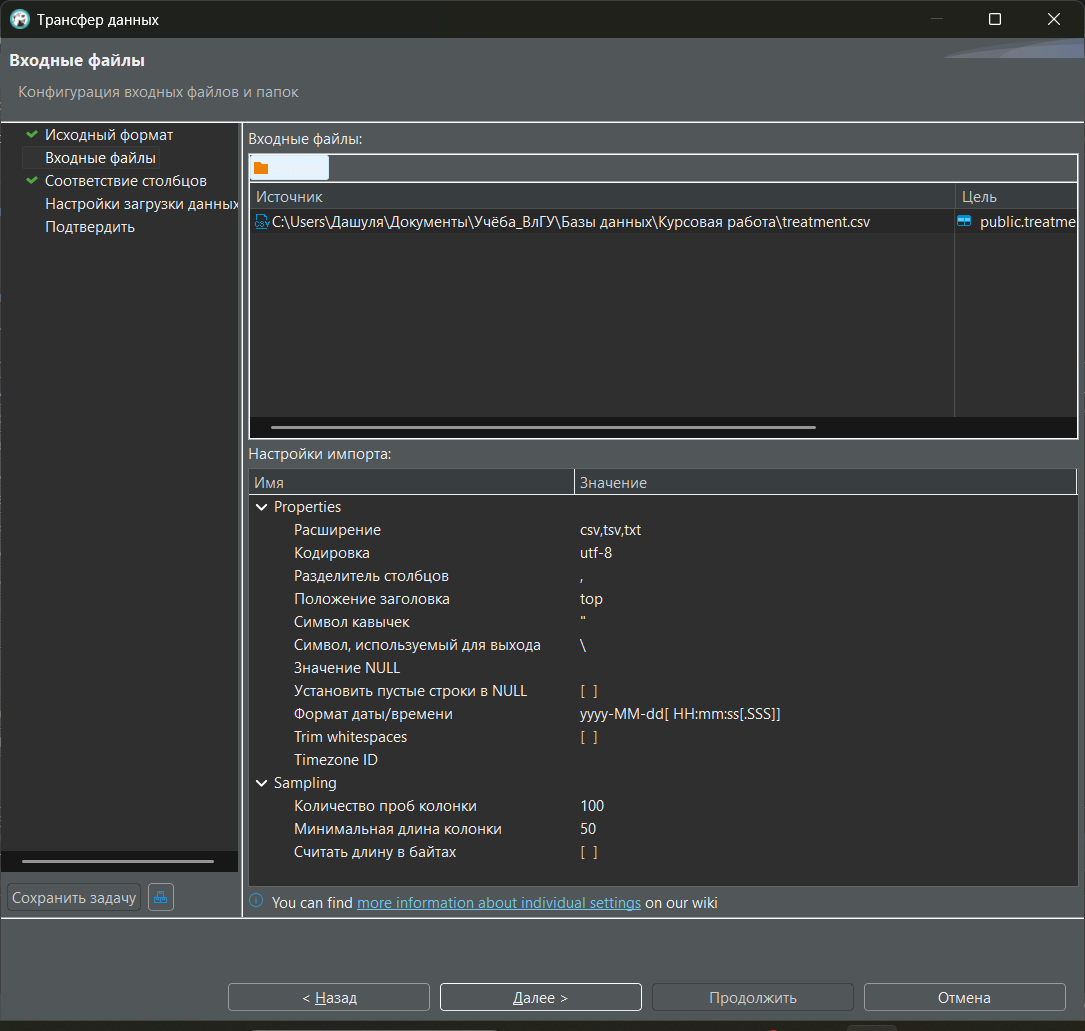


Рисунок 5 – Импортируем

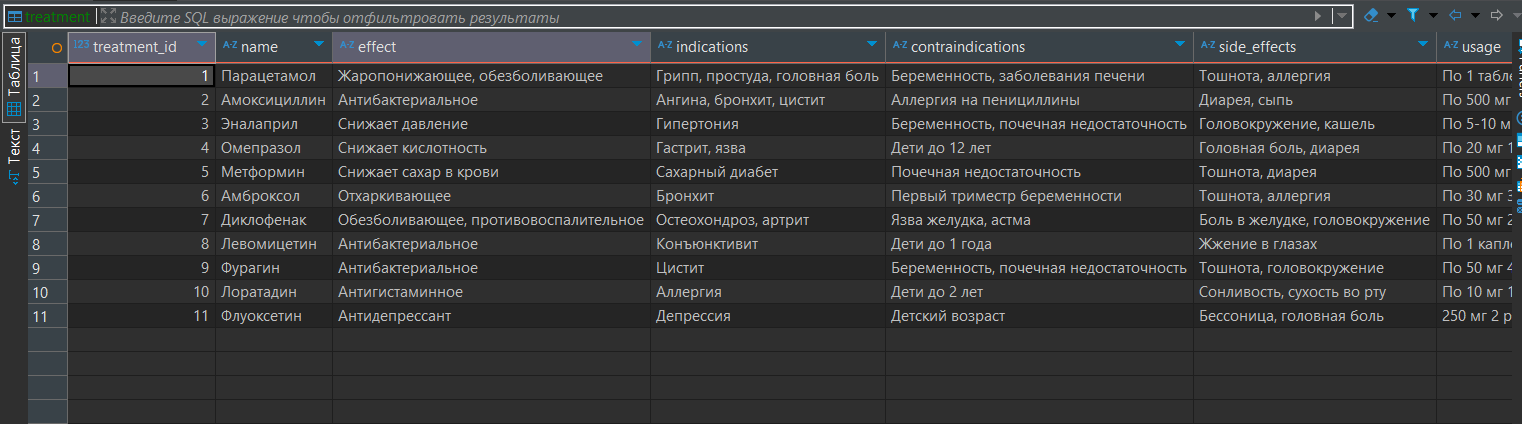


Рисунок 6 – Импорт выполнен успешно

SQL-скрипт для импорта:

**copy** treatment(treatment\_id,**name**,effect,indications,contraindications,side\_effects,**usage**)

**from** 'C:\Users\Дашуля\Документы\Учёба\_ВлГУ\Базы данных\Курсовая работа\treatment — копия.csv'

**delimiter** ','

**csv** **header**;

## Импорт и экспорт данных с помощью командной строки

@echo off

setlocal enabledelayedexpansion

REM ---- Настройки ----

set DB\_NAME=coursework

set NEW\_DB\_NAME=new\_coursework

set DB\_USER=postgres

set DB\_PASSWORD=postgres

set BACKUP\_DIR=C:\DB\_Backups

set DATE=%date:~-4%-%date:~3,2%-%date:~0,2%

REM ---- Создаем папку для бэкапов ----

if not exist "%BACKUP\_DIR%" mkdir "%BACKUP\_DIR%"

REM ---- a) Экспорт основной таблицы ----

set EXPORT\_TABLE=Doctor

echo Exporting %EXPORT\_TABLE%...

psql -U %DB\_USER% -d %DB\_NAME% -c "COPY %EXPORT\_TABLE% TO '%BACKUP\_DIR%\%EXPORT\_TABLE%-%DATE%.csv' WITH CSV HEADER;"

REM ---- b) Экспорт связанных таблиц ----

for %%T in (Patient Appointment Diagnosis) do (

echo Exporting %%T...

psql -U %DB\_USER% -d %DB\_NAME% -c "COPY %%T TO '%BACKUP\_DIR%\%%T-%DATE%.csv' WITH CSV HEADER;"

)

REM ---- c) Создание новой БД ----

echo Creating new database...

psql -U %DB\_USER% -c "DROP DATABASE IF EXISTS %NEW\_DB\_NAME%;"

psql -U %DB\_USER% -c "CREATE DATABASE %NEW\_DB\_NAME%;"

psql -U %DB\_USER% -d %NEW\_DB\_NAME% -f "C:\path\to\create\_tables.sql"

REM ---- d) Импорт в новую БД ----

for %%T in (Doctor Patient Appointment Diagnosis) do (

if exist "%BACKUP\_DIR%\%%T-%DATE%.csv" (

echo Importing %%T...

psql -U %DB\_USER% -d %NEW\_DB\_NAME% -c "COPY %%T FROM '%BACKUP\_DIR%\%%T-%DATE%.csv' WITH CSV HEADER;"

)

)

REM ---- e) Сравнение данных ----

echo Original DB:

psql -U %DB\_USER% -d %DB\_NAME% -c "SELECT \* FROM Doctor LIMIT 5;"

echo New DB:

psql -U %DB\_USER% -d %NEW\_DB\_NAME% -c "SELECT \* FROM Doctor LIMIT 5;"

endlocal

Альтернативный вариант через скрипты и интерфейс:

**select** *t*.\*

**from** treatment *t*

**join** possible\_treatment *pt* **on** *t*.treatment\_id = *pt*.treatment\_id

**join** diagnosis *d* **on** *pt*.diagnosis\_id = *d*.diagnosis\_id

**join** disease *ds* **on** *d*.disease\_id = *ds*.disease\_id

**where** *ds*.name = 'Аллергия';

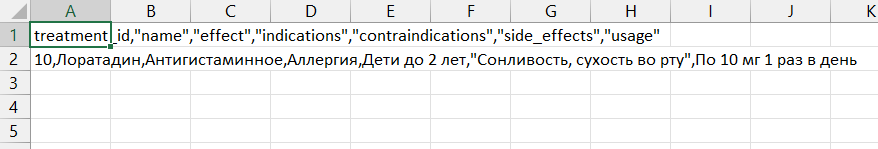


Рисунок 7 – Экспорт данных возможного лечения заболевания «Аллергия»

# УПРАВЛЕНИЕ ДОСТУПОМ

## Sql-скрипт создания ролей

-- создание ролей

**create** **role** doctor\_role;

**create** **role** patient\_role;

**create** **role** admin\_role;

**create** **role** analyst\_role;

-- назначение прав для роли врача

**grant** **select**, **insert**, **update** **on**

doctor, appointment, diagnosis, prescribed\_treatment

**to** doctor\_role;

-- назначение прав для роли пациента

**grant** **select** **on**

patient, appointment, diagnosis, treatment

**to** patient\_role;

-- назначение прав администратора

**grant** **all** **privileges** **on** **all** **tables** **in** **schema** public

**to** admin\_role;

-- назначение прав аналитика

**grant** **select** **on** **all** **tables** **in** **schema** public

**to** analyst\_role;

**grant** **insert**, **update** **on**

disease, treatment

**to** analyst\_role;

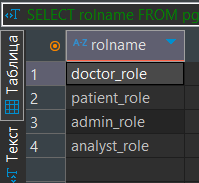


Рисунок 8 – Создание ролей выполнено успешно

## Sql-скрипт создания пользователей

**do** **$$**

**declare**

doc record;

username **text**;

username\_count **int**;

**begin**

**for** doc **in** **select** \* **from** doctor **loop**

username := **lower**(

**substring**(doc.**name** **from** 1 **for** 1) ||

**substring**(**split\_part**(doc.**name**, ' ', 2) **from** 1 **for** 1) ||

**case** **when** **split\_part**(doc.**name**, ' ', 3) <> ''

**then** **substring**(**split\_part**(doc.**name**, ' ', 3) **from** 1 **for** 1)

**else** '' **end**

);

**select** **count**(\*) **into** username\_count

**from** pg\_user

**where** usename = username;

**if** username\_count > 0 **then**

username := username || username\_count;

**end** **if**;

**execute** 'create user ' || username || ' with password ''' || username || '123''';

**execute** 'grant doctor\_role to ' || username;

**raise** **notice** 'создан пользователь: %', username;

**end** **loop**;

**end** **$$**;

**do** **$$**

**declare**

pat record;

username **text**;

username\_count **int**;

**begin**

**for** pat **in** **select** \* **from** patient **loop**

username := **lower**(

**substring**(pat.**name** **from** 1 **for** 1) ||

**substring**(**split\_part**(pat.**name**, ' ', 2) **from** 1 **for** 1) ||

**case** **when** **split\_part**(pat.**name**, ' ', 3) <> ''

**then** **substring**(**split\_part**(pat.**name**, ' ', 3) **from** 1 **for** 1)

**else** '' **end**

);

**select** **count**(\*) **into** username\_count

**from** pg\_user

**where** usename = username;

**if** username\_count > 0 **then**

username := username || username\_count;

**end** **if**;

**execute** 'create user ' || username || ' with password ''' || username || '123''';

**execute** 'grant patient\_role to ' || username;

**end** **loop**;

**end** **$$**;

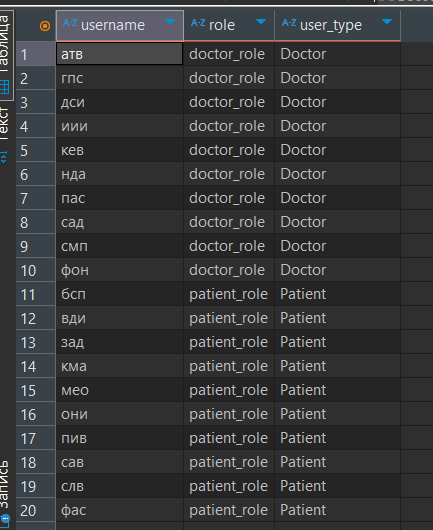


Рисунок 9 – Успешное создание пользователей, логин которых   
формируется из инициалов

## Работа с правами доступа пользователей

Скрипт для командной строки:

@echo off

setlocal enabledelayedexpansion

:: Configuration

set DB\_NAME=coursework

set PG\_PATH="C:\Program Files\PostgreSQL\15\bin\psql"

set OUTPUT\_FILE=access\_test\_results.log

:: Clear previous results

echo === Access Control Test Results === > %OUTPUT\_FILE%

:: 1. Test doctor\_role

echo Testing doctor\_role...

%PG\_PATH% -U гпс -d %DB\_NAME% -c "

DO \$\$

BEGIN

-- Allowed operations

PERFORM 1 FROM Doctor LIMIT 1;

RAISE NOTICE 'PASS: Doctor can SELECT from Doctor table';

INSERT INTO Appointment(doctor\_id, patient\_id, date, duration)

VALUES (1, 1, NOW(), 30);

RAISE NOTICE 'PASS: Doctor can INSERT into Appointment';

-- Denied operations

BEGIN

DELETE FROM Patient;

RAISE EXCEPTION 'FAIL: Doctor should not delete patients';

EXCEPTION WHEN insufficient\_privilege THEN

RAISE NOTICE 'PASS: Doctor cannot DELETE from Patient (as expected)';

END;

END \$\$" -W >> %OUTPUT\_FILE% 2>&1

:: 2. Test patient\_role

echo Testing patient\_role...

%PG\_PATH% -U пив -d %DB\_NAME% -c "

DO \$\$

BEGIN

-- Allowed

PERFORM 1 FROM Patient WHERE patient\_id = 1;

RAISE NOTICE 'PASS: Patient can view own record';

-- Denied

BEGIN

UPDATE Doctor SET name = 'Hacked';

RAISE EXCEPTION 'FAIL: Patient should not update doctors';

EXCEPTION WHEN insufficient\_privilege THEN

RAISE NOTICE 'PASS: Patient cannot UPDATE Doctor table';

END;

END \$\$" -W >> %OUTPUT\_FILE% 2>&1

:: Show results

echo Testing completed. Results saved to %OUTPUT\_FILE%

type %OUTPUT\_FILE%

endlocal

# Заключение

В ходе выполнения курсовой работы была разработана структура базы данных для медицинского учреждения (7 связанных таблиц).Были   
реализованы SQL-скрипты создания и наполнения БД**,** резервное копирование через командную строку (pg\_dump) и интерфейс DBeaver**,** импорт данных из CSV-файлов, так же были протестированы ограничения доступа:врачи могут редактировать приемы, но не удалять пациентов**,** пациенты имеют доступ только к своим данным**,** администраторы обладают полными правами.Была создана полнофункциональная база данных с механизмами защиты информации и восстановления после сбоев. Результаты тестирования подтвердили корректность работы SQL-запросов**,** эффективность разграничения прав доступа, надежность резервного копирования.

Перспективы развития:

- Добавление шифрования конфиденциальных данных

- Интеграция с веб-интерфейсом для врачей и пациентов

# Список использованных источников

1. pgAdmin Documentation: Managing Roles [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.pgadmin.org/docs/pgadmin4/latest/roles.html. – Текст: электронный
2. PostgreSQL Documentation: Database Roles [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/user-manag.html. – Текст: электронный.
3. Моргунов, Е. П. PostgreSQL. Основы языка SQL : учебное пособие / Е. П. Моргунов. — Москва: ДМК Пресс, 2021. — 324 с. — ISBN 978-5-93700-123-7.
4. Астахова, И. Ф. SQL в примерах и задачах: учебное пособие / И. Ф. Астахова, А. П. Толстобров, В. М. Мельников. — Санкт-Петербург: БХВ-Петербург, 2022. — 416 с. — ISBN 978-5-9775-6789-4.
5. PostgreSQL Documentation: Backup and Restore [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/backup.html. – Текст: электронный.
6. PostgreSQL Documentation: Database Roles [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/user-manag.html. – Текст: электронный.
7. PostgreSQL Documentation: SQL Commands – COPY [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.postgresql.org/docs/current/sql-copy.html. – Текст: электронный.
8. pgAdmin Documentation: Import/Export Data [Электронный ресурс]. – Электрон. дан. – Режим доступа: https://www.pgadmin.org/docs/pgadmin4/latest/import\_export\_data.html. – Текст: электронный.