Отчет

О проделанной работе на тему:

«Разработка структуры базы данных для хранения диагностикой информации болезни Паркинсона»

Выполнил студент

Группы 12448/2 ИНЖДАН

Лобанов С.А.

22.06.2023

Содержание:

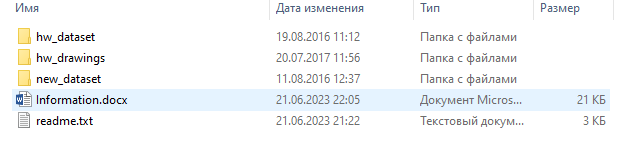
1. Задание стр.
2. Описание выполнения практического задания стр.
3. Выводы о результатах проделанной работы стр.

Задание:

1. Ознакомьтесь со структурой и описанием файлов, содержащих диагностическую информацию болезни Паркинсона.
2. Разработайте логическую и физическую модели базы данных, обеспечивающее хранение диагностической информации болезни Паркинсона, включая:
   1. Перечень таблиц для хранения данных, обоснование перечня с указанием целевого назначения каждой из таблиц
   2. Описания полей каждой из таблиц для хранения данных, включая типы полей для физической модели
   3. Описания и типизацию связей таблиц
3. Подготовьте отчет о проделанной работе, в виде документа, включающего:
   1. Титульный лист
   2. Содержание
   3. Задание
   4. Описание выполнения практического задания, включая снимки экрана, иллюстрирующие этапы выполнения задания, и пояснения к ним.
   5. Выводы о результатах выполнения заданий
4. Отчет прикрепите в качестве ответа на задание

Описание выполнения практической работы:

Для выполнения практической работы в первую очередь были проанализированы данные, полученные в рамках работы.



Изучив файлы readme, было выявлено следующее: данные были получены в результате исследования пациентов в Турции. У испытуемых были обнаружены симптомы болезни «Паркинсона». Для анализа и составления медицинской практики, данные были преобразованы в цифровой вид и переданы в виде «датасетов» для дальнейшего анализа.

Условно данные можно разложить на 2 исследования: Звуковое и исследование на базе данных, полученных от пациентов с помощью диагностических рисунков.

Звуковые данные занесены в датасет через запятую и имеют следующий формат:

Датасет представляет собой результат анализа звуковых записей испытуемых

1 столбец : идентификатор испытуемого

столбцы 2-6: - Дрожание (локальное), Дрожание (локальное, абсолютное), дрожание (rap), дрожание (ppq5), дрожание (ddp)

столбцы 7-12: Мерцание (локальное), Мерцание (локальное, dB), мерцание (apq3), Мерцание (apq5), мерцание (apq11), мерцание (dda),

столбцы 13-14: AC,NTH,HTN [Определенного рода характеристикаи анализа]

столбцы 15-19: Медианный тон[голоса],Средний тон, Стандартное отклонение тона,Минимум тона,Максимум тона

столбцы 20-23: Число пульсаций,Число периодов,Среднее периодов,Стандартное отклонение периода,

столбцы 24-26: Доля не озвученных фрагментов,Число разрывов голоса,Степень сломленности речи

28 столбец - Оценка UPDRS (Единая шкала оценки болезни Паркинсона)

29 столбец - [1 - есть болезнь п., 2 - болезни п. нет

Данные изображений содержат несколько видов тестов (статический и динамический).

В датасетах имеют следующий формат:

Набор данных разделяется значениями CSV следующим образом;

X ; Y; Z; Давление; Угол захвата; Временная метка; Идентификатор теста

----------------

Идентификатор теста:

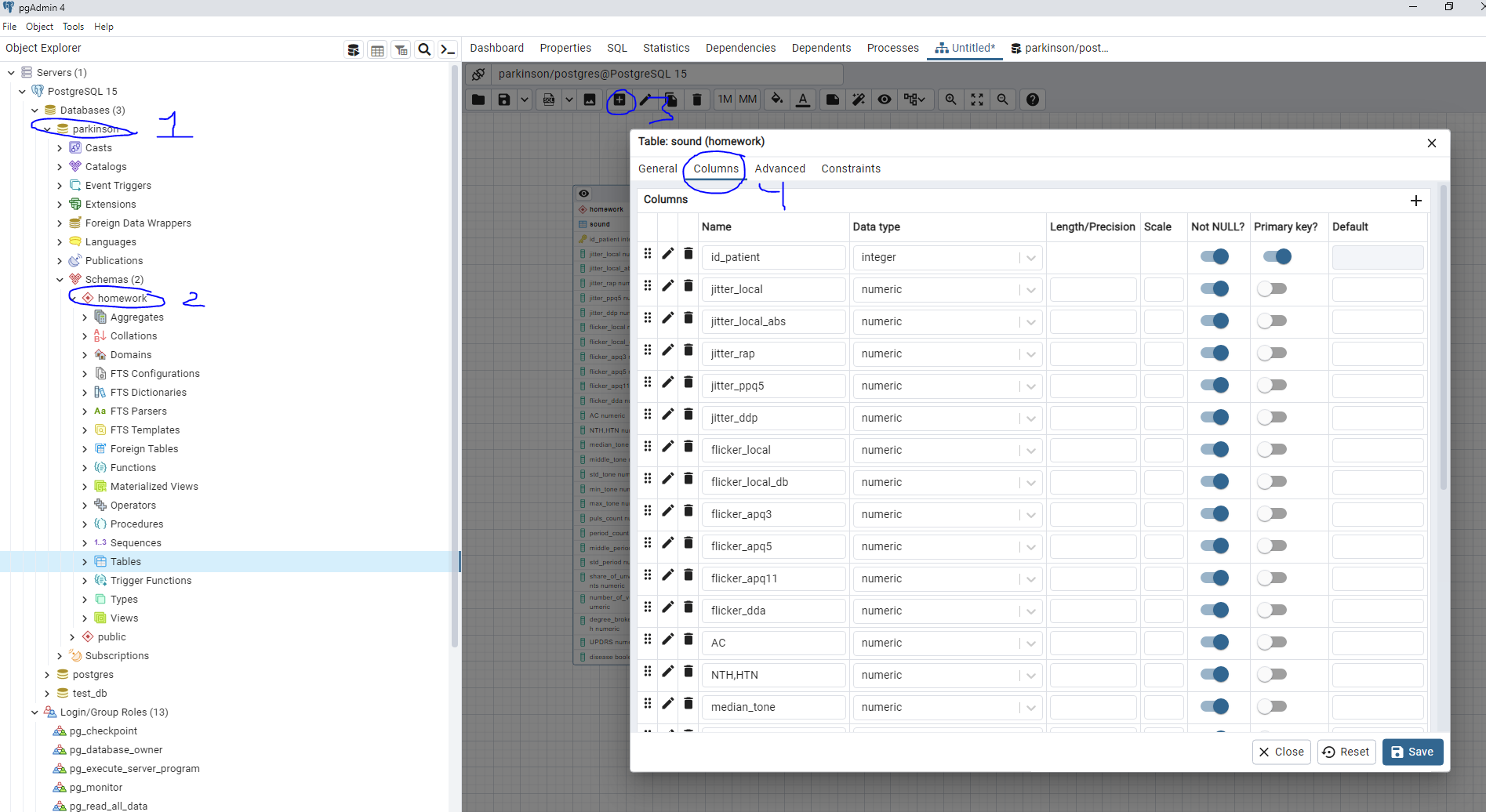
0: Статический спиральный тест (Нарисуйте по заданному спиральному рисунку)

1: Динамический спиральный тест (спиральный узор будет мигать через определенное время, поэтому испытуемым необходимо продолжить рисование)

2: Тест на круговые движения (испытуемые рисуют круги вокруг красной точки)

Для разработки логический и физической модели данных я использовал ПО PostgeSQL, а именно компонент pgAdmin.

Логическая структура была создана с помощью инструмента ERD tools.

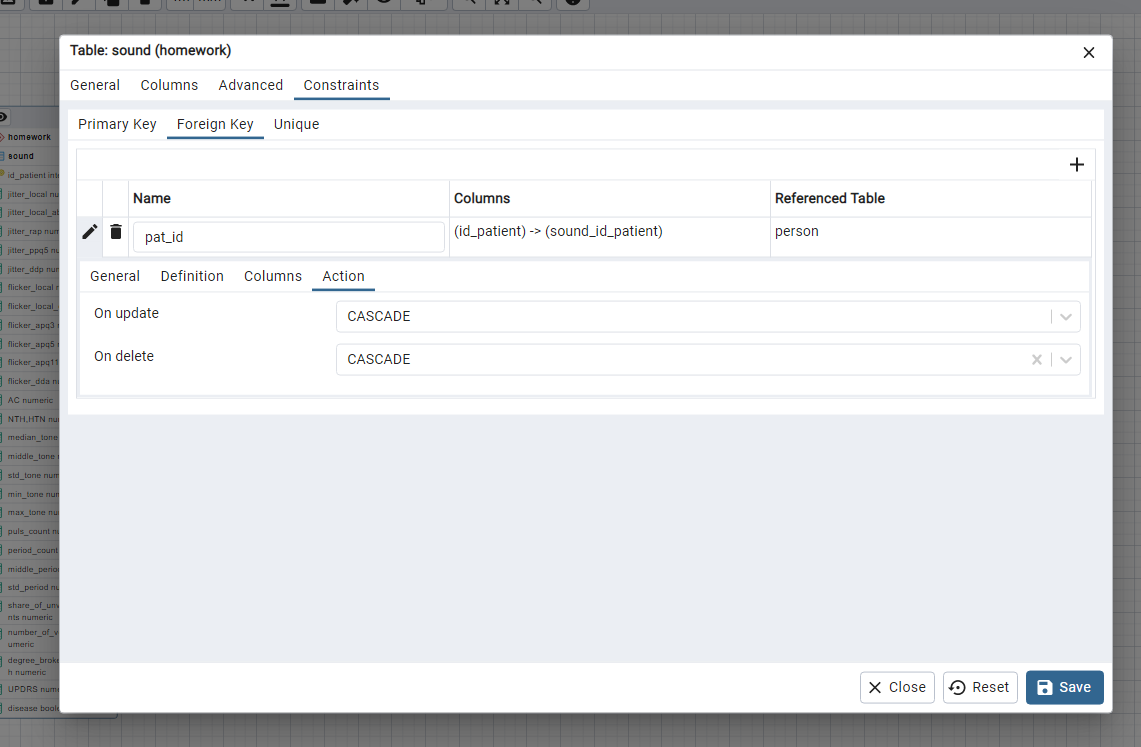


Для начала я создал новую базу данных «parkinson»

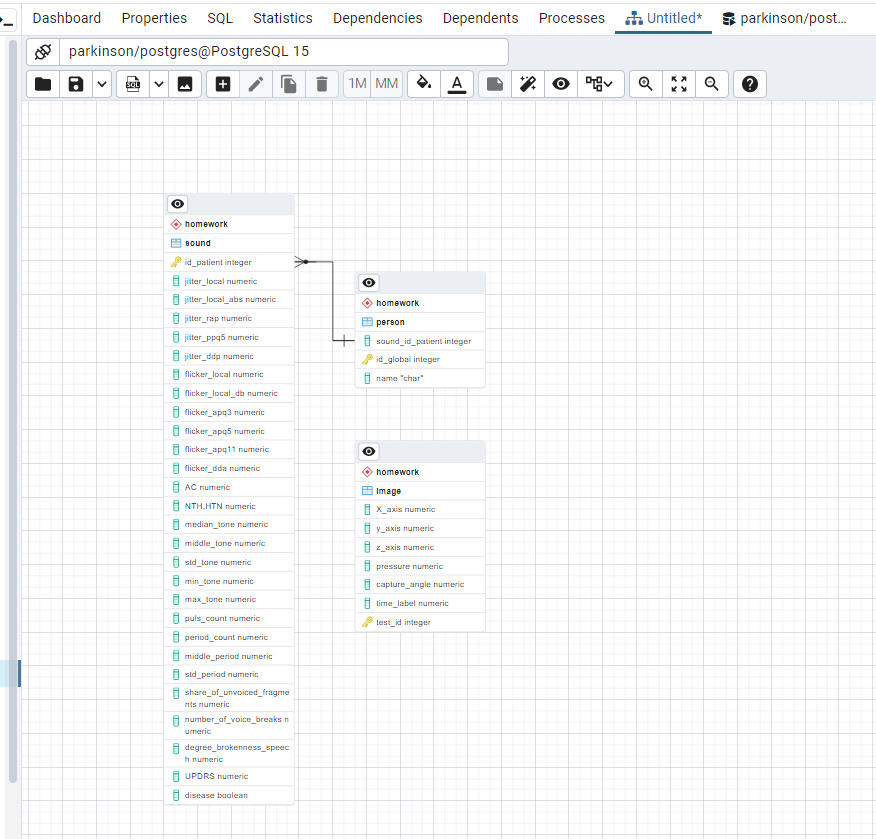
В ней создал схему «homework»

Далее запустил инструмент ERD и через нее создал таблицу sound / image / person

Создав доп таблицу для дальнейшей обработки и нормализации данных person, связал первичный ключ таблицы sound с ней



Логическая схема на данном этапе получилась вида:



После этого я выполнил SQL запрос для создания таблиц в базе, используя мастер ERD.

-- This script was generated by the ERD tool in pgAdmin 4.

-- Please log an issue at https://redmine.postgresql.org/projects/pgadmin4/issues/new if you find any bugs, including reproduction steps.

BEGIN;

CREATE TABLE IF NOT EXISTS homework.sound

(

id\_patient integer NOT NULL,

jitter\_local numeric NOT NULL,

jitter\_local\_abs numeric NOT NULL,

jitter\_rap numeric NOT NULL,

jitter\_ppq5 numeric NOT NULL,

jitter\_ddp numeric NOT NULL,

flicker\_local numeric NOT NULL,

flicker\_local\_db numeric NOT NULL,

flicker\_apq3 numeric NOT NULL,

flicker\_apq5 numeric NOT NULL,

flicker\_apq11 numeric NOT NULL,

flicker\_dda numeric NOT NULL,

"AC" numeric NOT NULL,

"NTH,HTN" numeric NOT NULL,

median\_tone numeric NOT NULL,

middle\_tone numeric NOT NULL,

std\_tone numeric NOT NULL,

min\_tone numeric NOT NULL,

max\_tone numeric NOT NULL,

puls\_count numeric NOT NULL,

period\_count numeric NOT NULL,

middle\_period numeric NOT NULL,

std\_period numeric NOT NULL,

share\_of\_unvoiced\_fragments numeric NOT NULL,

number\_of\_voice\_breaks numeric NOT NULL,

degree\_brokenness\_speech numeric NOT NULL,

"UPDRS" numeric NOT NULL,

disease boolean NOT NULL DEFAULT 1

);

CREATE TABLE IF NOT EXISTS homework.image

(

"X\_axis" numeric NOT NULL,

y\_axis numeric NOT NULL,

z\_axis numeric NOT NULL,

pressure numeric NOT NULL,

"сapture\_angle" numeric NOT NULL,

time\_label numeric NOT NULL,

test\_id integer NOT NULL,

PRIMARY KEY (test\_id)

);

CREATE TABLE IF NOT EXISTS homework.person

(

sound\_id\_patient integer NOT NULL,

id\_global integer NOT NULL GENERATED ALWAYS AS IDENTITY,

name "char" NOT NULL,

PRIMARY KEY (id\_global)

);

ALTER TABLE IF EXISTS homework.sound

ADD CONSTRAINT pat\_id FOREIGN KEY (id\_patient)

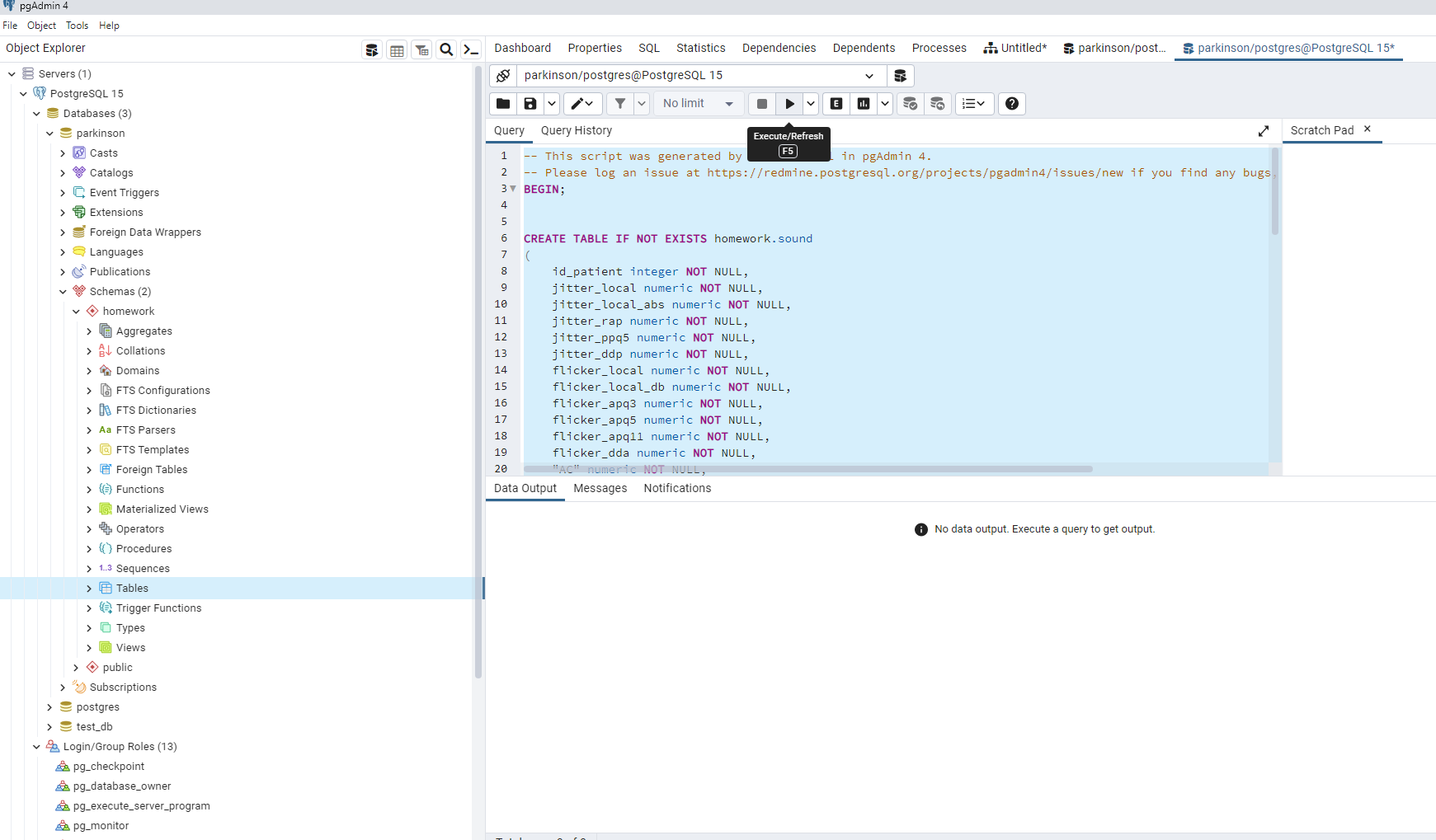
REFERENCES homework.person (sound\_id\_patient) MATCH SIMPLE

ON UPDATE CASCADE

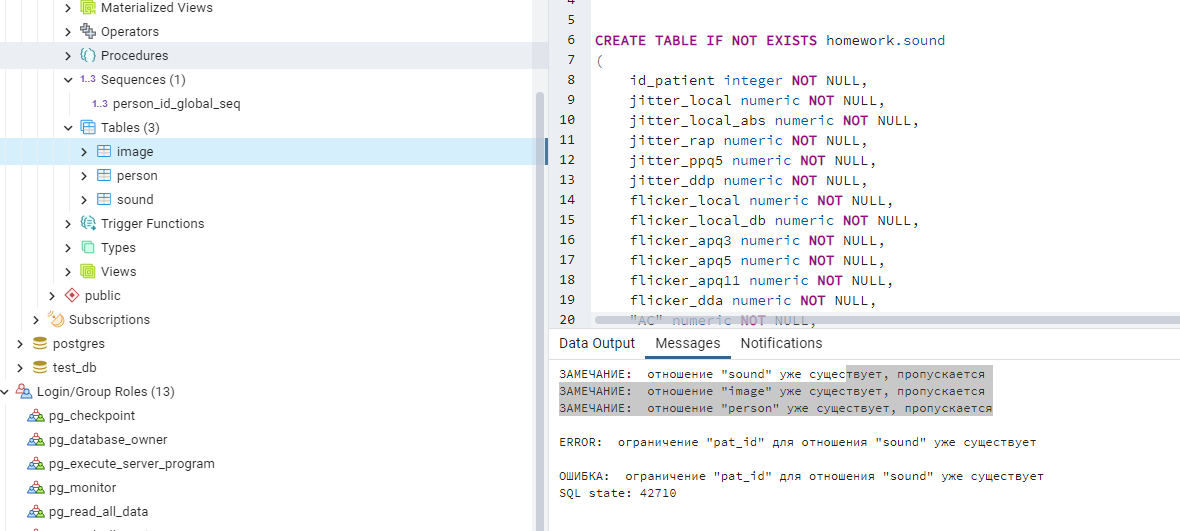
ON DELETE CASCADE

NOT VALID;

END;



После выполняем SQL запрос и получаем таблицы



Выводы о результатах проделанной работы:

По мере выполнения работы, я проанализировал предоставленные датасеты и создал логическую и физическую структуру базы данных для дальнейшей работы с ними.

Изучил основные инструменты ERD tools и создал базу, схему, таблицы.