

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа по базам данных №1

Вариант № 31071

Выполнил:

Студент группы Р3106

Мельник Фёдор Александрович

Проверил:

Вербовой Александр Александрович,

Преподаватель-практик ФПИиКТ

Санкт-Петербург, 2025

Оглавление

Текст задания	3
Описание предметной области.....	3
Список сущностей и их классификация.....	3
Инфологическая модель.....	4
Даталогическая модель	4
Реализация даталогической модели на SQL	5
Вывод	8

Текст задания

Для выполнения лабораторной работы №1 необходимо:

1. На основе предложенной предметной области (текста) составить ее описание. Из полученного описания выделить сущности, их атрибуты и связи.
2. Составить инфологическую модель.
3. Составить даталогическую модель. При описании типов данных для атрибутов должны использоваться типы из СУБД PostgreSQL.
4. Реализовать даталогическую модель в PostgreSQL. При описании и реализации даталогической модели должны учитываться ограничения целостности, которые характерны для полученной предметной области.
5. Заполнить созданные таблицы тестовыми данными.

Для создания объектов базы данных у каждого студента есть своя схема. Название схемы соответствует имени пользователя в базе studs (sXXXXXX). Команда для подключения к базе studs:

```
psql -h pg -d studs
```

Каждый студент должен использовать свою схему при работе над лабораторной работой №1 (а также в рамках выполнения 2, 3 и 4 этапа курсовой работы).

Описание предметной области

Но даже и в этом случае он мог бы полностью вернуться к жизни после одной-двух минут пребывания в вакууме; за такой срок кровь и лимфа, хорошо защищенные в своих системах, еще не успевают закипеть. Требовалось одно - правильная рекомпрессия, постепенный возврат к нормальному давлению. Рекордное время пребывания в вакууме равно почти пяти минутам. Это доказано не экспериментом, а несчастным случаем; пострадавшего удалось спасти, хотя он и остался частично парализованным из-за воздушной эмболии.

Список сущностей и их классификация

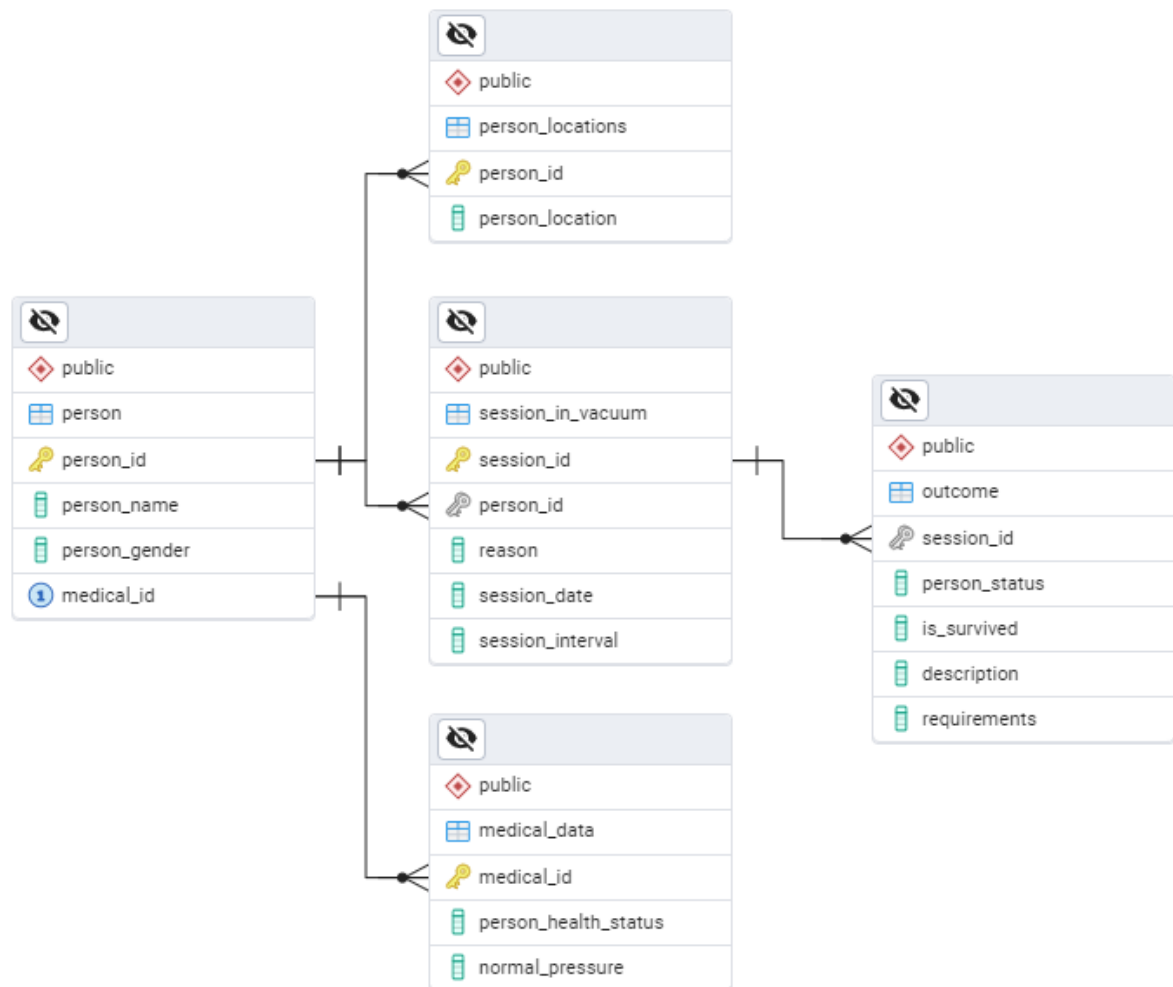
Стержневые: person

Характеристические: medical_data, outcome, person_locations

Ассоциативные: session_in_vacuum

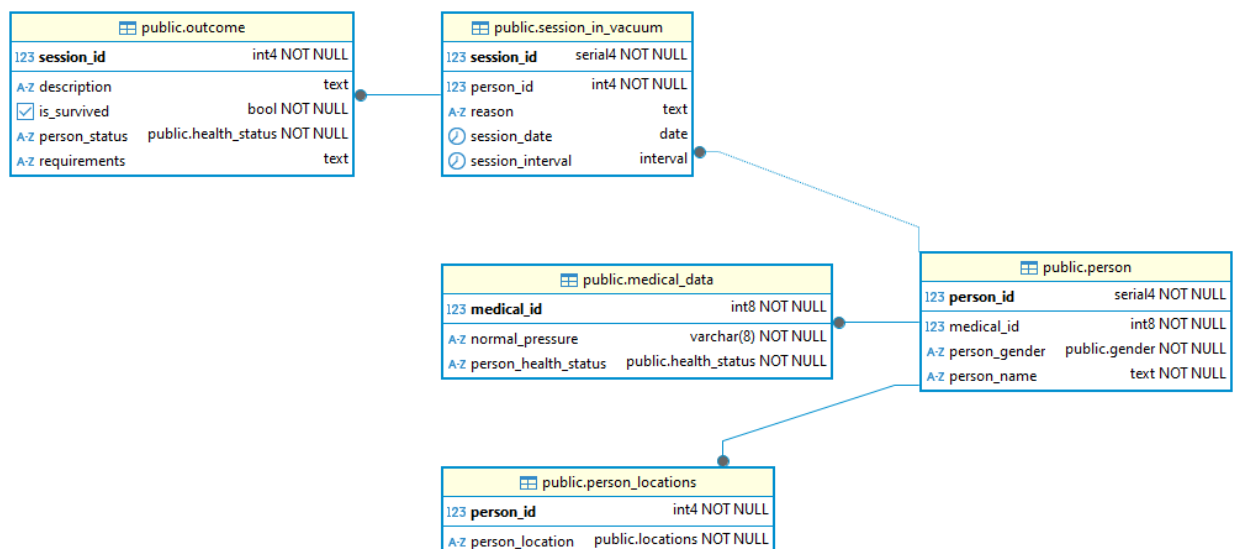
Инфологическая модель

https://github.com/ldpst/itmo/blob/main/sem-2_databases/labs/lab1/ER-diagram.png



Даталогическая модель

https://github.com/ldpst/itmo/blob/main/sem-2_databases/labs/lab1/DM-diagram.png



Реализация даталогической модели на SQL

https://github.com/ldpst/itmo/blob/main/sem-2_databases/labs/lab1/build.sql

```
BEGIN;

CREATE TYPE gender AS ENUM ('мужской', 'женский');

CREATE TABLE IF NOT EXISTS person
(
    person_id SERIAL PRIMARY KEY,
    person_name TEXT NOT NULL,
    person_gender gender NOT NULL,
    medical_id BIGINT UNIQUE NOT NULL
);

CREATE TYPE health_status AS ENUM ('мёртв', 'ранен', 'травмирован', 'здоров');

CREATE TABLE IF NOT EXISTS medical_data
(
    medical_id BIGINT UNIQUE REFERENCES person(medical_id) ON DELETE
    CASCADE,
    person_health_status health_status NOT NULL,
    normal_pressure VARCHAR(8) NOT NULL,
    PRIMARY KEY (medical_id)
);

CREATE FUNCTION validate_pressure()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW.normal_pressure !~* '^[1-9][0-9]{0,3}/[1-9][0-9]{0,2}$'
    THEN
        RAISE EXCEPTION 'Неверный формат normal_pressure';
```

```

        END IF;

        RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER check_pressure BEFORE INSERT OR UPDATE ON medical_data
FOR EACH ROW EXECUTE FUNCTION validate_pressure();

CREATE TYPE locations AS ENUM ('вакуум', 'нормальное');

CREATE TABLE IF NOT EXISTS person_locations (
    person_id INT UNIQUE REFERENCES person(person_id) ON DELETE CASCADE,
    person_location locations NOT NULL,
    PRIMARY KEY (person_id)
);

CREATE TABLE IF NOT EXISTS session_in_vacuum (
    session_id SERIAL PRIMARY KEY,
    person_id INT NOT NULL REFERENCES person(person_id) ON DELETE CASCADE,
    reason TEXT,
    session_date DATE,
    session_interval INTERVAL
);

CREATE TABLE IF NOT EXISTS outcome (
    session_id INT NOT NULL UNIQUE REFERENCES session_in_vacuum(session_id)
ON DELETE CASCADE,
    person_status health_status NOT NULL,
    is_survived BOOLEAN NOT NULL,
    description TEXT,
    requirements TEXT
);

```

```
INSERT INTO person(person_name, person_gender, medical_id) VALUES

('Алекс', 'мужской', '12345'),

('Майк', 'мужской', '54321'),

('Дроджия', 'женский', '89654');

INSERT INTO medical_data VALUES

('12345', 'здоров', '120/79'),

('54321', 'травмирован', '121/80'),

('89654', 'мёртв', '135/80');

INSERT INTO person_locations VALUES

('1', 'вакуум'),

('2', 'нормальное'),

('3', 'нормальное');

INSERT INTO session_in_vacuum(person_id, reason, session_date, session_interval) VALUES

('1', 'эксперимент', '2025-02-21', '54 seconds'),

('2', 'несчастный случай', '1999-01-23', '4 minutes 56 seconds'),

('3', 'убийство', '2003-04-12', '3 hours 12 minutes 14 seconds');

INSERT INTO outcome VALUES

('1', 'здоров', TRUE, NULL, 'рекомпрессия'),

('2', 'травмирован', TRUE, 'удалось спасти, но остался частично парализованным из-за воздушной эмболии', null),

('3', 'мёртв', FALSE, 'летальный исход', NULL);

END;
```

Вывод

В процессе выполнения лабораторной работы я научился создавать инфологическую и даталогическую модели сущностей, по которым реализовал базу данных на SQL (PostgreSQL)