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 P3106  
Мельник Фёдор Александрович

Проверил:

Вербовой Александр Александрович,

Преподаватель-практик ФПИиКТ

Санкт-Петербург, 2025

Оглавление

[Текст задания 3](#_Toc191011360)

[Описание предметной области 3](#_Toc191011361)

[Список сущностей и их классификация 3](#_Toc191011362)

[Инфологическая модель 4](#_Toc191011363)

[Даталогическая модель 4](#_Toc191011364)

[Реализация даталогической модели на SQL 5](#_Toc191011365)

[Вывод 8](#_Toc191011366)

# Текст задания

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Контент, сгенерированный ИИ, может содержать ошибки.

# 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

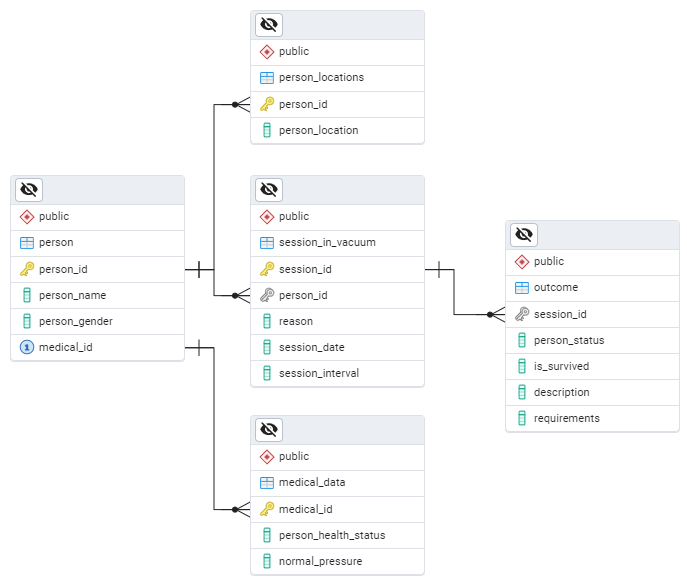
Стержневые: person

Характеристические: medical\_data, outcome, person\_locations

Ассоциативные: session\_in\_vacuum

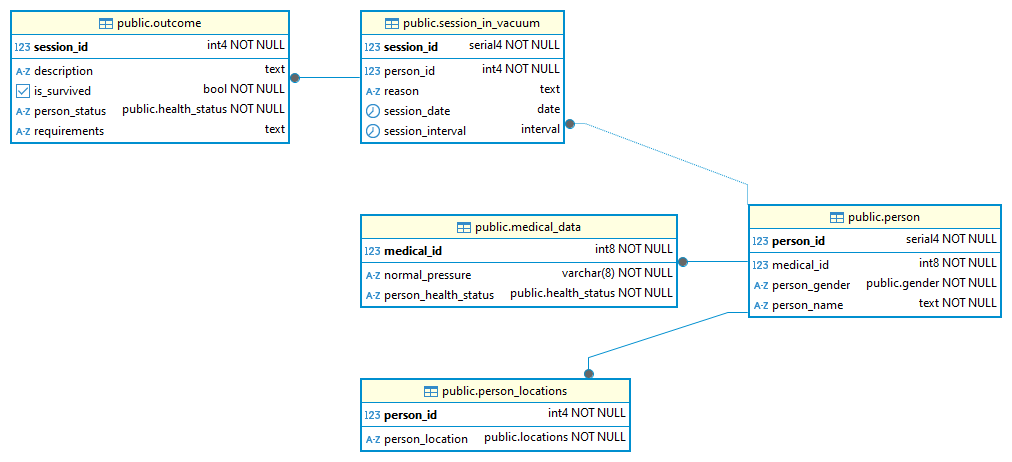
# Инфологическая модель

<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)