

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

*Факультет программной инженерии и компьютерной  
техники*

*Направление подготовки: 09.03.01 - Информатика и  
вычислительная техника, Компьютерные системы и  
технологии*

*Дисциплина «Базы данных»*

**Отчет**  
**По лабораторной работе №1**  
**Вариант №52236**

Студент:  
Миронов Иван Николаевич

Группа: Р3132

Преподаватель:  
Афанасьев Дмитрий Борисович

Санкт-Петербург, 2024 год

## Оглавление

Оглавление .....	2
Текст задания.....	3
Описание предметной области .....	3
Список сущностей и их классификация .....	4
Модели.....	5
Инфологическая модель .....	5
Даталогическая модель.....	5
Реализация даталогической модели на SQL.....	6
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

В строгом смысле слова мы и специалистами-то в них не являемся. Меня, например, Мискатоникский университет направил в Антарктику как геолога: с помощью замечательной буровой установки, сконструированной профессором нашего же университета Фрэнком Х. Пэбоди, мы должны были добыть с большой глубины образцы почвы и пород. Не стремясь прослыть пионером в других областях науки, я тем не менее надеялся, что это новое механическое устройство поможет мне многое разведать и увидеть в ином свете.

## Список сущностей и их классификация

Стержневые:

- Ученый
- Работа
- Результат

Характеристические:

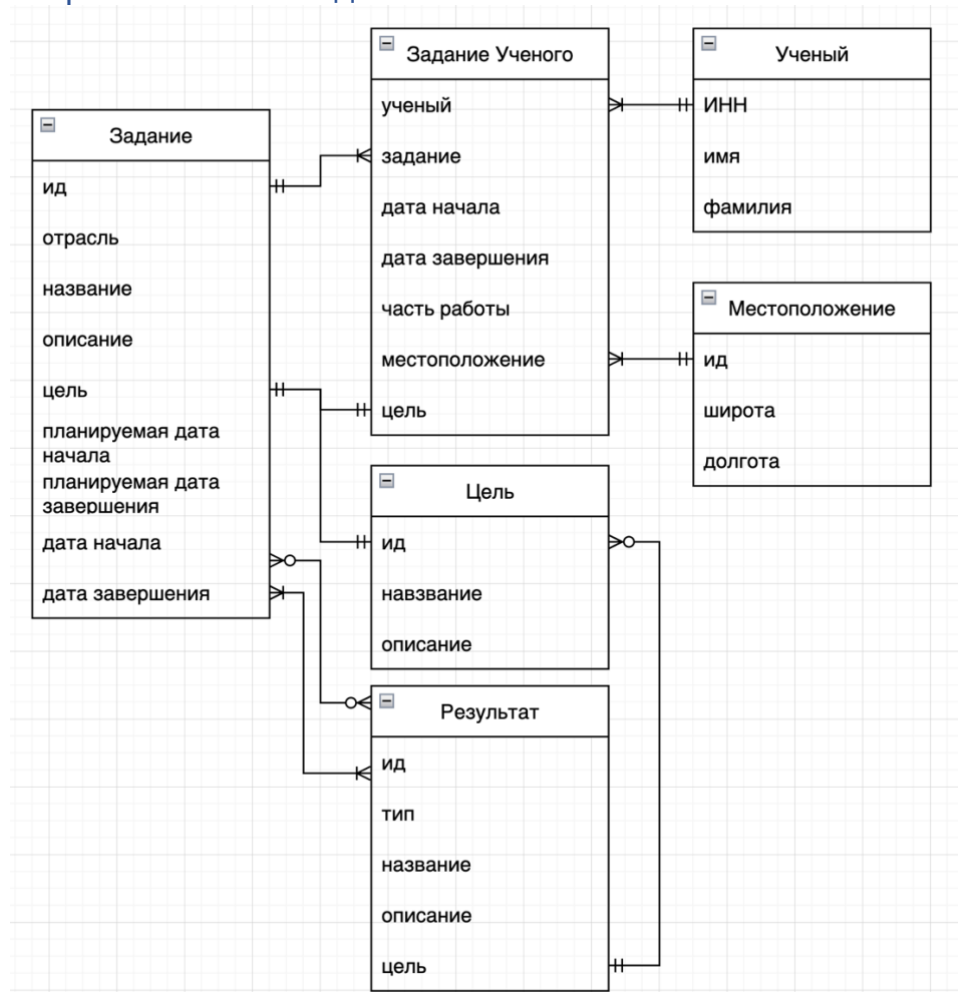
- Местоположение
- Цель

Ассоциативные:

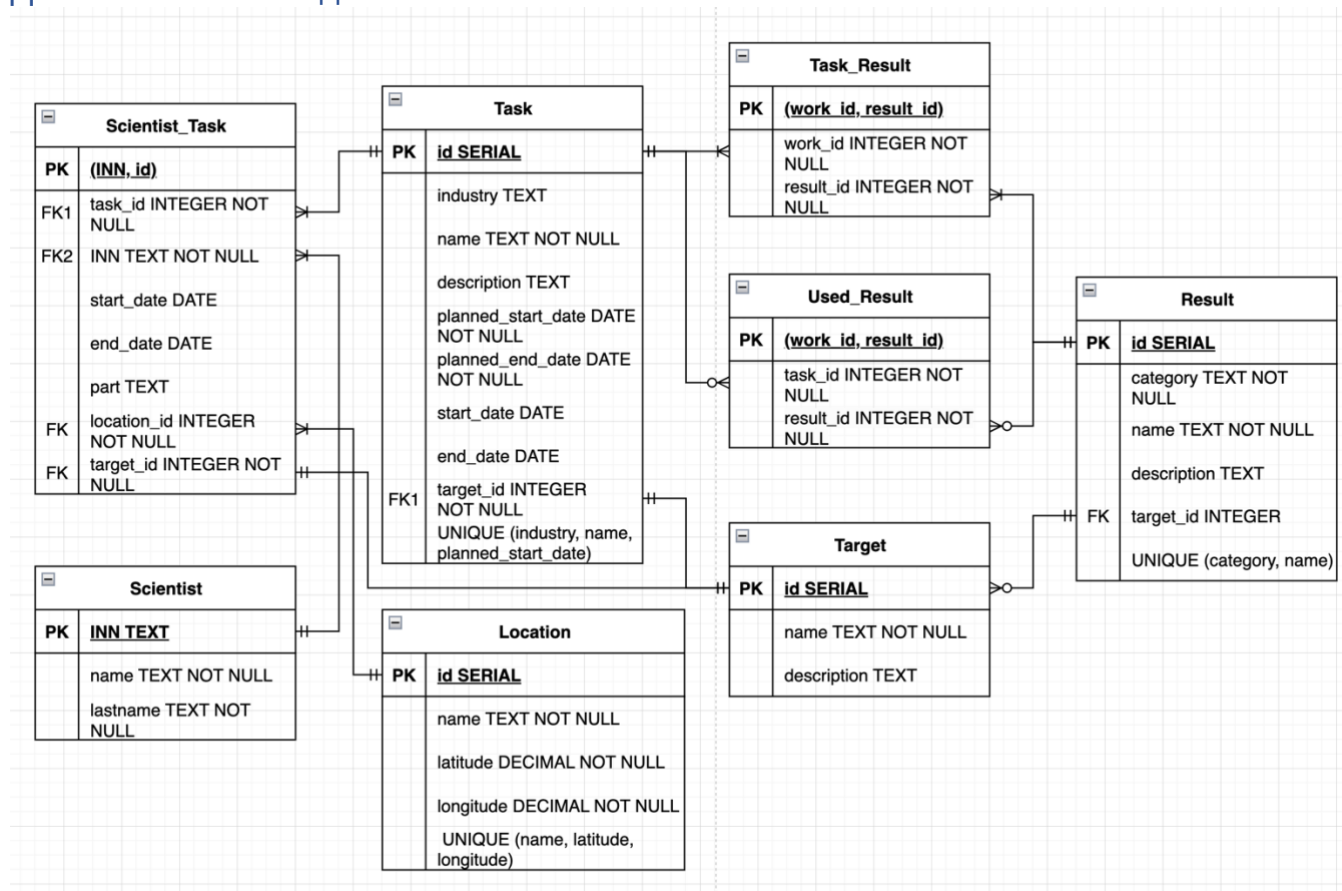
- Работа ученого

# Модели

## Инфологическая модель



## Даталогическая модель



## Реализация даталогической модели на SQL

```
DROP TABLE IF EXISTS scientist CASCADE;
DROP TABLE IF EXISTS task CASCADE ;
DROP TABLE IF EXISTS location CASCADE ;
DROP TABLE IF EXISTS result CASCADE ;
DROP TABLE IF EXISTS target CASCADE ;
DROP TABLE IF EXISTS scientist_task CASCADE ;
DROP TABLE IF EXISTS task_result CASCADE ;
DROP TABLE IF EXISTS used_result CASCADE ;

CREATE TABLE IF NOT EXISTS scientist (
    inn TEXT NOT NULL PRIMARY KEY,
    name TEXT NOT NULL,
    lastname TEXT NOT NULL,
    CHECK (LENGTH(inn)=10)
);

CREATE TABLE IF NOT EXISTS target (
    id SERIAL PRIMARY KEY,
    name TEXT NOT NULL,
    description TEXT
);

CREATE TABLE IF NOT EXISTS task (
    id SERIAL PRIMARY KEY,
    industry TEXT,
    name TEXT NOT NULL,
    description TEXT,
    planned_start_date DATE NOT NULL,
    planned_end_date DATE NOT NULL,
    start_date DATE,
    end_date DATE,
    target_id INTEGER REFERENCES target(id) NOT NULL,
    UNIQUE (industry, name, planned_start_date),
    CHECK ((planned_end_date > planned_start_date) AND (end_date >
start_date))
);

CREATE TABLE IF NOT EXISTS location (
    id SERIAL PRIMARY KEY,
    name TEXT NOT NULL,
    latitude DECIMAL NOT NULL,
    longitude DECIMAL NOT NULL,
    UNIQUE (name, longitude, latitude),
    CHECK ((longitude between 0 AND 180) AND (latitude BETWEEN 0 AND 90))
);

CREATE TABLE IF NOT EXISTS result (
    id SERIAL PRIMARY KEY,
    category TEXT NOT NULL,
    name TEXT NOT NULL,
    description TEXT,
    target_id INTEGER REFERENCES target(id),
    UNIQUE (category, name)
);

CREATE TABLE IF NOT EXISTS scientist_task (
    task_id INTEGER REFERENCES task (id) NOT NULL,
    inn TEXT REFERENCES scientist(inn) NOT NULL,
    start_date DATE,
    end_date DATE,
    part TEXT,
    location INTEGER REFERENCES location(id) NOT NULL,
    target INTEGER REFERENCES target(id),
    PRIMARY KEY (task_id, inn),
```

```

    CHECK (end_date > start_date)
);

CREATE TABLE IF NOT EXISTS task_result (
    task_id INTEGER REFERENCES task (id)    NOT NULL,
    result_id INTEGER REFERENCES result(id) NOT NULL,
    PRIMARY KEY (task_id, result_id)
);

CREATE TABLE IF NOT EXISTS used_result (
    task_id INTEGER REFERENCES task (id)    NOT NULL,
    result_id INTEGER REFERENCES result(id) NOT NULL,
    PRIMARY KEY (task_id, result_id)
);

INSERT INTO scientist (inn, name, lastname) VALUES
('1234567890', 'Frank', 'Peabody'),
('1234567899', 'John', 'Johnson');

INSERT INTO target (name, description) VALUES
('soil samples', 'soil and rock samples from great depths'),
('obtain soil samples', 'obtain soil better than others'),
('drilling machine', 'artesian action drill with the principle of a
rotating hammer'),
('geological survey', 'geological survey ...');

INSERT INTO task (industry, name, description, planned_start_date,
planned_end_date, start_date, end_date, target_id) VALUES
('mining industry', 'obtain soil samples', 'obtain by drill machine',
'2017-03-01', '2017-04-10', '2017-03-01', '2017-04-11', 1),
('engineering industry', 'drilling machine', 'creation of a drilling
machine', '2016-03-01', '2017-01-01', '2016-03-01', '2017-01-01', 3);

INSERT INTO location (name, latitude, longitude) VALUES
('antartica', 68.34, 77.58),
('Miskatonic University', 71.41, 41.81);

INSERT INTO result (category, name, description, target_id) VALUES
('thing', 'drilling machine', 'one example of drilling machine', 2);

INSERT INTO result (category, name, description) VALUES
('thing', 'soil samples', 'soil and rock samples from great depths'),
('conclusion', 'no soil samples', 'no land under the ice');

INSERT INTO scientist_task (task_id, inn, start_date, end_date, part,
location, target) VALUES
(1, '1234567890', '2016-03-01', '2017-01-01', 'all drilling machine', 2,
3),
(1, '1234567899', '2017-03-01', '2017-04-11', 'geologist', 1, 4);

INSERT INTO task_result (task_id, result_id) VALUES
(1, 2),
(2, 1);

INSERT INTO used_result (task_id, result_id) VALUES
(1, 2);

```

## Вывод

Во время работы над лабораторной работой я ознакомился с архитектурой построения БД ANSI-SPARC, синтаксисом PostgreSQL, освоил азы проектирования БД.