

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 «Информатика»*

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

Выполнил:  
Миронов Иван Николаевич

Группа:  
Р3132

Принял:  
Афанасьев Дмитрий Борисович

Санкт-Петербург, 2024 год

## Оглавление

Текст задания: .....	3
Описание предметной области: .....	4
Список сущностей и их классификация .....	5
Модели .....	6
Инфологическая модель: .....	6
Даталогическая модель: .....	6
Реализация даталогической модели на SQL .....	7
Вывод: .....	9

## 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:

- Человек – имя, фамилия, специальность, надежда
- Машина – название, предназначение
- Университет – название
- Командировка – местоположение, университет, цель

Характеристические:

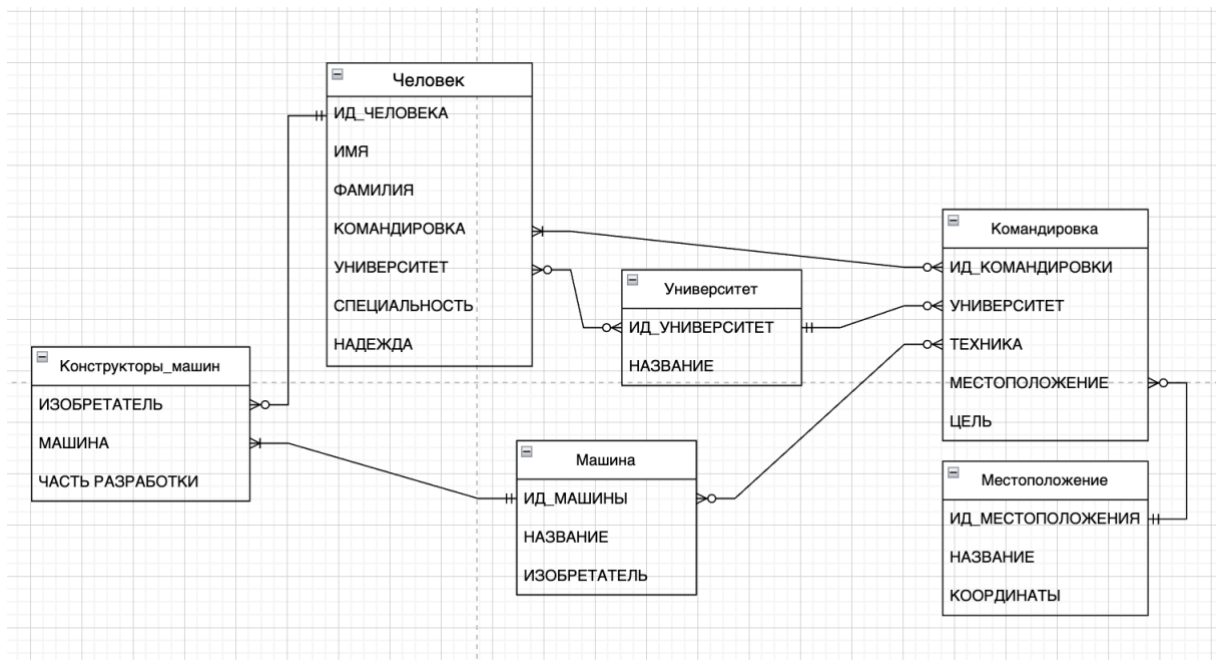
- Местоположение – название, координаты

Ассоциативные:

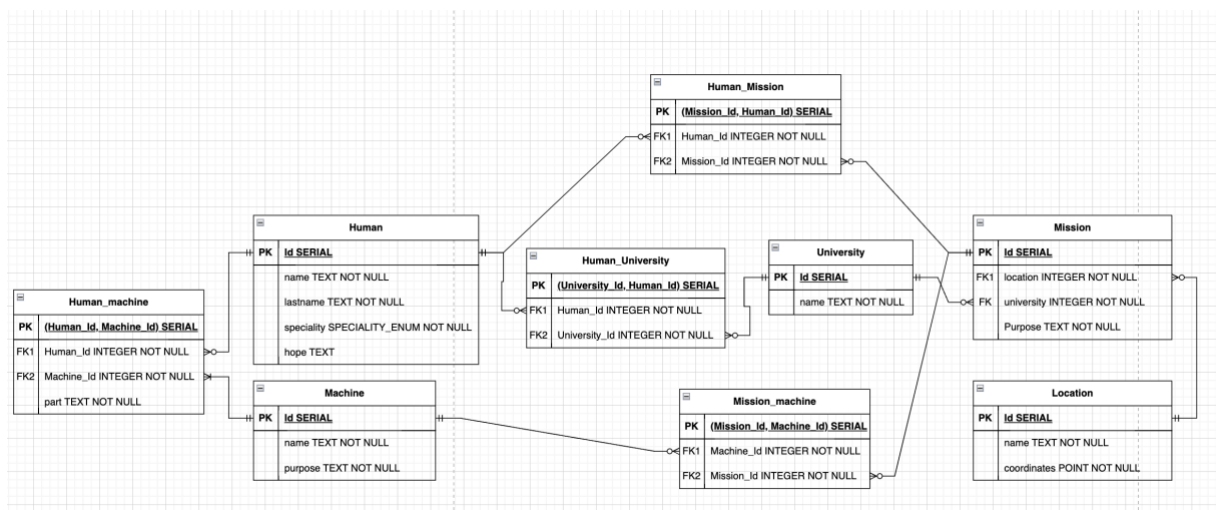
- Конструкторы машин – машина–человек, вклад в создание модели

# Модели

## Инфологическая модель:



## Даталогическая модель:



## Реализация даталогической модели на SQL

```
DROP TYPE IF EXISTS speciality_enum CASCADE;
```

```
DROP TABLE IF EXISTS human CASCADE;  
DROP TABLE IF EXISTS machine CASCADE;  
DROP TABLE IF EXISTS university CASCADE;  
DROP TABLE IF EXISTS location CASCADE;  
DROP TABLE IF EXISTS mission CASCADE;  
DROP TABLE IF EXISTS mission_machine CASCADE;  
DROP TABLE IF EXISTS human_mission CASCADE;  
DROP TABLE IF EXISTS human_university CASCADE;  
DROP TABLE IF EXISTS human_machine CASCADE;
```

```
CREATE TYPE speciality_enum AS ENUM ('geologist', 'physicist', 'professor');
```

```
CREATE TABLE IF NOT EXISTS human (  
    id SERIAL PRIMARY KEY,  
    name TEXT NOT NULL,  
    lastname TEXT NOT NULL,  
    speciality speciality_enum NOT NULL,  
    hope TEXT  
);
```

```
CREATE TABLE IF NOT EXISTS machine (  
    id SERIAL PRIMARY KEY,  
    name TEXT NOT NULL,  
    purpose TEXT NOT NULL  
);
```

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

```
CREATE TABLE IF NOT EXISTS location (  
    id SERIAL PRIMARY KEY,  
    name TEXT NOT NULL,  
    coordinates POINT NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS mission (  
    id SERIAL PRIMARY KEY,  
    location INTEGER REFERENCES location(id) NOT NULL,  
    university INTEGER REFERENCES university(id) NOT NULL,  
    purpose TEXT NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS mission_machine (  
    mission_id INTEGER REFERENCES mission(id) NOT NULL,  
    machine_id INTEGER REFERENCES machine(id) NOT NULL,  
    PRIMARY KEY (mission_id, machine_id)  
);
```

```
CREATE TABLE IF NOT EXISTS human_mission (  

```

```

mission_id INTEGER REFERENCES mission(id) NOT NULL,
human_id INTEGER REFERENCES human(id) NOT NULL,
PRIMARY KEY (mission_id, human_id)
);

CREATE TABLE IF NOT EXISTS human_university (
human_id INTEGER REFERENCES human(id) NOT NULL,
university_id INTEGER REFERENCES university(id) NOT NULL,
PRIMARY KEY (human_id, university_id)
);

CREATE TABLE IF NOT EXISTS human_machine (
human_id INTEGER REFERENCES human(id) NOT NULL,
machine_id INTEGER REFERENCES machine(id) NOT NULL,
part TEXT NOT NULL,
PRIMARY KEY (human_id, machine_id)
);

INSERT INTO human (name, lastname, speciality, hope) VALUES
('John', 'Johnson', 'geologist', 'mechanical device assistance');

INSERT INTO human (name, lastname, speciality) VALUES
('Thom', 'Yorke', 'physicist'),
('Frank', 'Peabody', 'professor'),
('Jim', 'Morrison', 'professor');

INSERT INTO machine (name, purpose) VALUES
('drilling rig', 'obtain soil samples'),
('washing machine', 'wash clothes');

INSERT INTO university (name) VALUES
('Miskatonic University'),
('ITMO');

INSERT INTO location (name, coordinates) VALUES
('Antarctica', POINT(69.3, 65)),
('Moscow', POINT(55.7558, 37.6173));

INSERT INTO mission (location, university, purpose) VALUES
(1, 1, 'obtain soil samples'),
(2, 1, 'wash sneakers in machine');

INSERT INTO human_mission (human_id, mission_id) VALUES
(1, 1),
(2, 2),
(4, 2);

INSERT INTO human_university (human_id, university_id) VALUES
(1, 1);

INSERT INTO mission_machine (mission_id, machine_id) VALUES
(1, 1),
(2, 2);

INSERT INTO human_machine (human_id, machine_id, part) VALUES
(3, 1, 'construct');

```



## Вывод:

В ходе выполнения лабораторной работы я ознакомился с архитектурой построения баз данных ANSI/SPARC, научился строить инфологические и даталогические модели, работать с СУБД PostgreSQL