

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

Факультет программной инженерии и компьютерной техники

Лабораторная работа № 1

Вариант 31065

Выполнил: Зыков Андрей Алексеевич

Группа: Р3106

Проверил: Вербовой А. А.,

Преподаватель практики факультета ПИиКТ

Санкт - Петербург 2025

Оглавление

Задание.....	3
Описание предметной области	3
Классификация сущностей	3
Основная часть.....	4
Заполнение таблиц тестовыми данными	6
Выводы по работе	8

Задание

Лабораторная работа #1

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
```

Описание предметной области

В поле его сознания мелькнули зазубренные камни четырех крохотных внешних лун - Синопе, Пасифе, Ананке и Карме. Затем, вдвое ближе к Юпитеру, вторая четверка - Элара, Лиситея, Гималия и Леда. Они не заинтересовали его. Его путь лежал к усеянной кратерами Каллисто.

Классификация сущностей

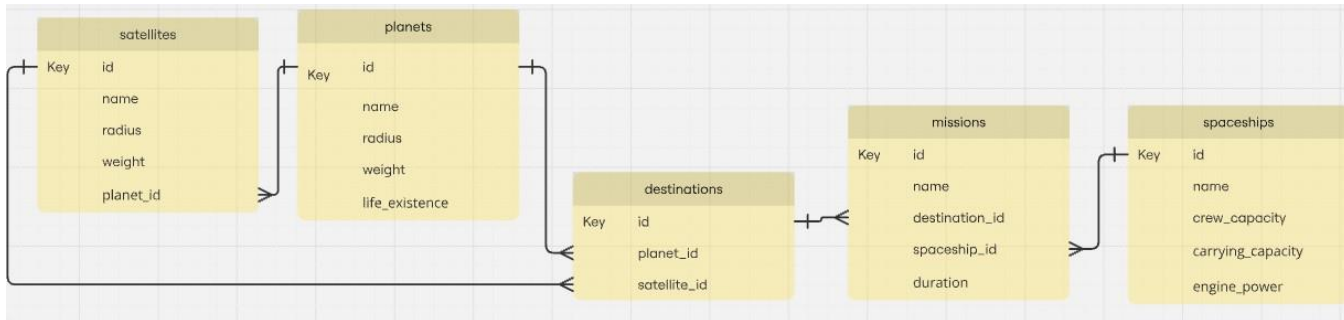
Стержневые: planets, missions

Ассоциативные: destinations

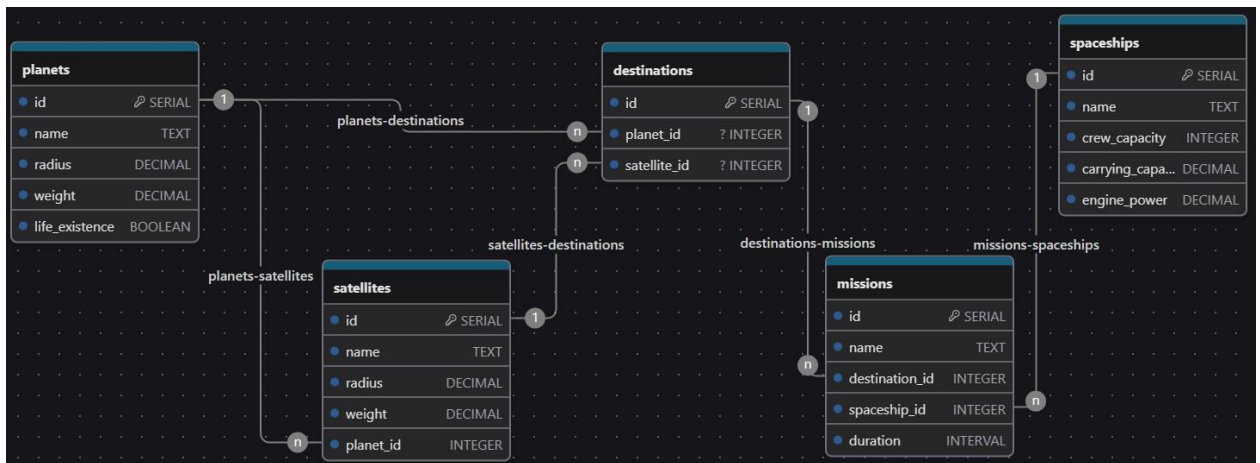
Характеристики: satellites, spaceships

Основная часть

Инфологическая модель



Даталогическая модель



Реализация даталогической модели в postgresSQL

```
CREATE TABLE satellites (  
    id SERIAL PRIMARY KEY NOT NULL UNIQUE,  
    name TEXT NOT NULL,  
    radius DECIMAL NOT NULL,  
    CHECK(radius > 0),  
    weight DECIMAL NOT NULL,  
    CHECK(weight > 0),  
    planet_id INTEGER NOT NULL  
);
```

```

CREATE TABLE planets (
    id SERIAL PRIMARY KEY NOT NULL UNIQUE,
    name TEXT NOT NULL,
    radius DECIMAL NOT NULL,
    CHECK(radius > 0),
    weight DECIMAL NOT NULL,
    CHECK(weight > 0),
    life_existence BOOLEAN NOT NULL
);

```

```

CREATE TABLE destinations (
    id SERIAL PRIMARY KEY NOT NULL UNIQUE,
    planet_id INTEGER,
    satellite_id INTEGER
);

```

```

CREATE TABLE missions (
    id SERIAL PRIMARY KEY NOT NULL UNIQUE,
    name TEXT NOT NULL,
    destination_id INTEGER NOT NULL,
    spaceship_id INTEGER NOT NULL,
    duration INTERVAL NOT NULL
);

```

```

CREATE TABLE spaceships (
    id SERIAL PRIMARY KEY NOT NULL UNIQUE,
    name TEXT NOT NULL,
    crew_capacity INTEGER NOT NULL,
    CHECK(crew_capacity >= 0),
    carrying_capacity DECIMAL NOT NULL,
    CHECK(carrying_capacity >= 0),

```

```
        engine_power DECIMAL NOT NULL,  
        CHECK(engine_power > 0)  
    );
```

```
ALTER TABLE satellites  
ADD FOREIGN KEY(planet_id) REFERENCES planets(id)  
ON UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE destinations  
ADD FOREIGN KEY(planet_id) REFERENCES planets(id)  
ON UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE destinations  
ADD FOREIGN KEY(satellite_id) REFERENCES satellites(id)  
ON UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE missions  
ADD FOREIGN KEY(destination_id) REFERENCES destinations(id)  
ON UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE missions  
ADD FOREIGN KEY(spaceship_id) REFERENCES spaceships(id)  
ON UPDATE CASCADE ON DELETE CASCADE;
```

Заполнение таблиц тестовыми данными

```
INSERT INTO planets (name, radius, weight, life_existence) VALUES  
('Jupiter', 69911000, 18987E23, false);
```

```
INSERT INTO planets (name, radius, weight, life_existence) VALUES  
('Earth', 6378000, 5.9742E24, true);
```

```
INSERT INTO satellites (name, radius, weight, planet_id) VALUES  
('Callisto', 2410.3E3, 1.08E23, 1);
```

```
INSERT INTO satellites (name, radius, weight, planet_id) VALUES  
( 'Moon', 1737.4E3, 7.36E22, 2);
```

```
INSERT INTO satellites (name, radius, weight, planet_id) VALUES  
( 'Sinope', 19E3, 7.5E16, 1);
```

```
INSERT INTO satellites (name, radius, weight, planet_id) VALUES  
( 'Pasiphae', 58E3, 3E17, 1);
```

```
INSERT INTO satellites (name, radius, weight, planet_id) VALUES  
( 'Leda', 20E3, 1.09E16, 1);
```

```
INSERT INTO destinations (planet_id, satellite_id) VALUES (null, 1);
```

```
INSERT INTO spaceships (name, crew_capacity, carrying_capacity,  
engine_power) VALUES ( 'Apollo', 3, 45578.12, 26895);
```

```
INSERT INTO missions (name, destination_id, spaceship_id, duration)  
VALUES ( 'Apollo-14', 1, 1, '3 years 2 months 24 days 6 hours 2 minutes  
36 seconds');
```

Выводы по работе

В ходе выполнения лабораторной работы я научился создавать инфологические и даталогические модели баз данных, познакомился с основами языка SQL и применил знания на практике для создания базы данных по заданным данным.