

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

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №1

Вариант 465592

Выполнил:

Горин Семён Дмитриевич

Группа Р3108

Проверила:

Заболотняя Ольга Михайловна

Санкт-Петербург 2025

Содержание

Задание.....	3
Список и классификация сущностей.....	3
Инфологическая модель (Логическая модель)	3
Даталогическая модель (Физическая модель).....	4
Реализация даталогической модели на SQL.....	4
Выводы	6

Задание

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.

Описание предметной области:

Горы еще зябли в тени, когда корабль достиг Шалмирейна. С высоты, на которой они находились, огромная чаша крепости выглядела совсем крохотной. Казалось просто невероятным, что когда-то от этого вот черного как ночь кружка зависели судьбы Земли.

Список и классификация сущностей

Стержневые: Ship, Flight, World, Point_of_interest, Location, Terrain;

Ассоциативные: ship_to_flight;

Инфологическая модель (Логическая модель)

Инфологическая модель представлена на рисунке 1.

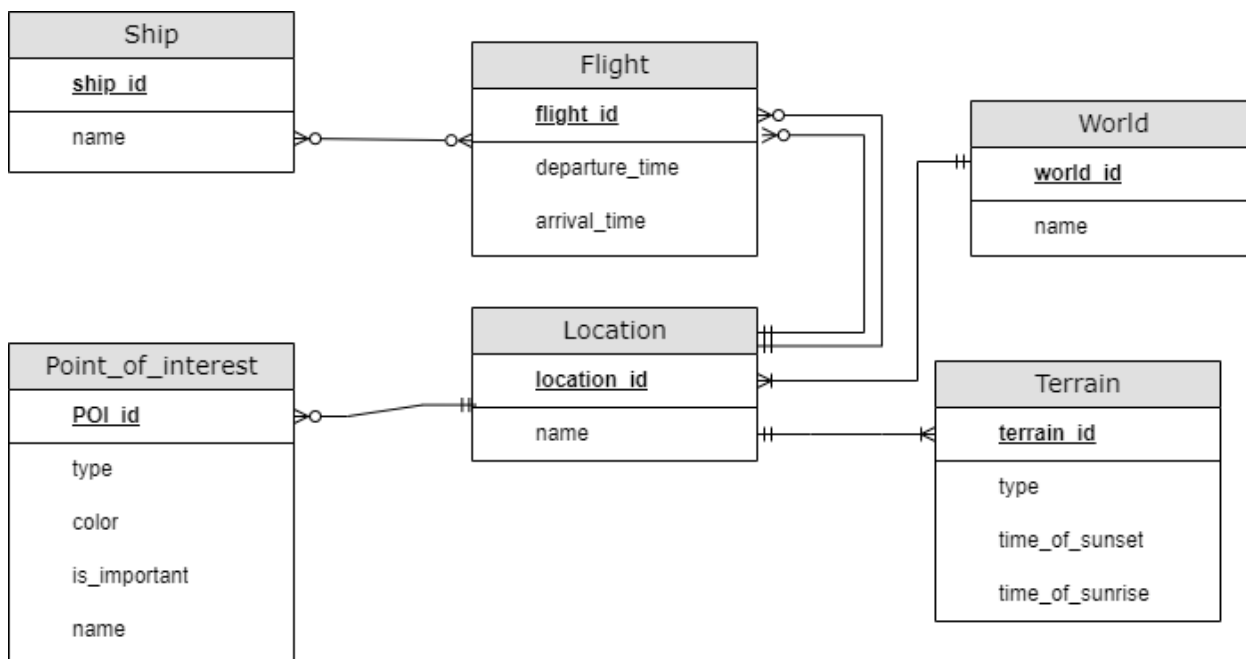


Рисунок 1. Инфологическая модель.

Даталогическая модель (Физическая модель)

Даталогическая модель представлена на рисунке 2.

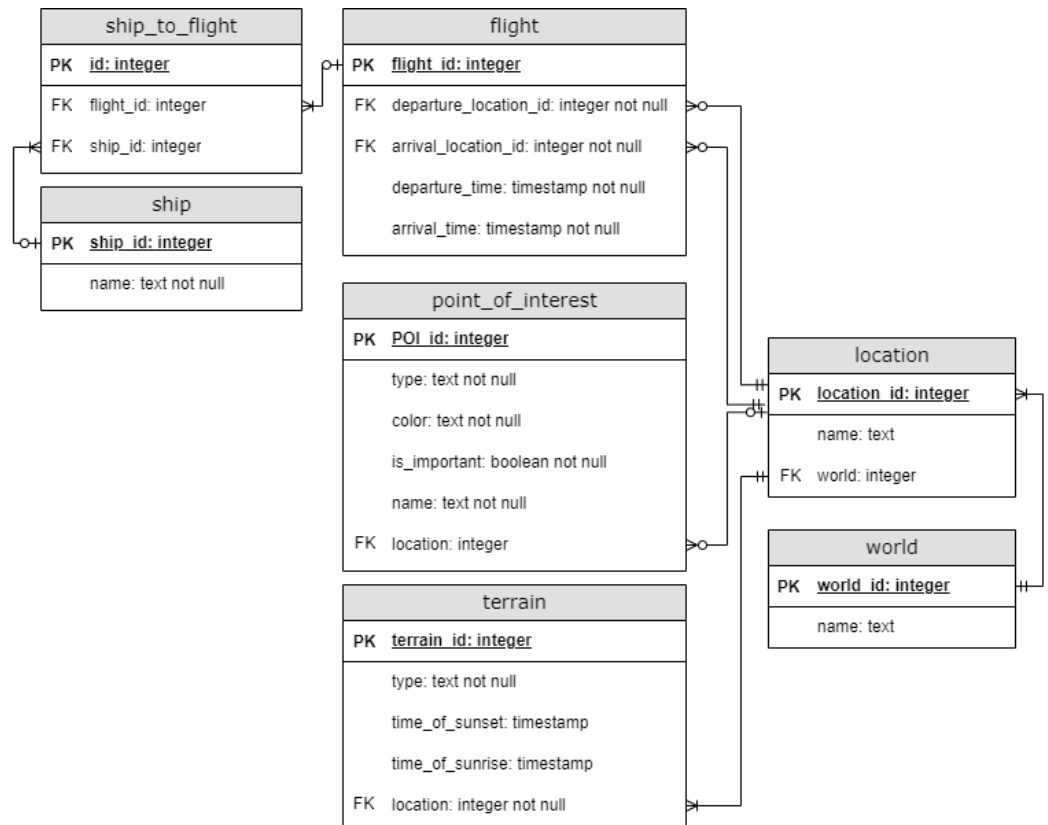


Рисунок 2. Даталогическая модель.

Реализация даталогической модели на SQL

BEGIN;

/ Создание отношений */*

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

```
CREATE TABLE IF NOT EXISTS location
(
location_id SERIAL PRIMARY KEY,
name TEXT,
world_id INT REFERENCES world(world_id)
);
```

```
CREATE TABLE IF NOT EXISTS point_of_interest
(
poi_id SERIAL PRIMARY KEY,
type TEXT NOT NULL,
color TEXT NOT NULL,
is_important BOOLEAN NOT NULL,
name TEXT NOT NULL,
location_id INT REFERENCES location(location_id)
);
```

```
CREATE TABLE IF NOT EXISTS flight
(
```

```

flight_id SERIAL PRIMARY KEY,
departure_location_id INT REFERENCES location(location_id) NOT NULL,
arrival_location_id INT REFERENCES location(location_id) NOT NULL,
departure_time TIMESTAMP NOT NULL,
arrival_time TIMESTAMP NOT NULL
);

CREATE TABLE IF NOT EXISTS ship
(
ship_id SERIAL PRIMARY KEY,
name TEXT NOT NULL
);

CREATE TABLE IF NOT EXISTS ship_to_flight
(
id SERIAL PRIMARY KEY,
flight_id INT REFERENCES flight(flight_id),
ship_id INT REFERENCES ship(ship_id)
);

CREATE TABLE IF NOT EXISTS terrain
(
terrain_id SERIAL PRIMARY KEY,
type TEXT NOT NULL,
time_of_sunset TIMESTAMP,
time_of_sunrise TIMESTAMP,
location_id INT REFERENCES location(location_id) NOT NULL
);

/* Добавление сущностей */

INSERT INTO world(name)
VALUES('Арда');

INSERT INTO location(name, world_id)
VALUES('Шалмирейн', 1);

INSERT INTO location(name, world_id)
VALUES('Валинор', 1);

INSERT INTO flight(departure_location_id, arrival_location_id,
departure_time, arrival_time)
VALUES
(
(SELECT location_id FROM location WHERE name = 'Валинор'),
(SELECT location_id FROM location WHERE name = 'Шалмирейн'),
'1000-10-10 11:30:30',
'1000-11-10 10:30:30'
);

INSERT INTO ship(name)
VALUES
(
'Галка'
);

INSERT INTO terrain(type, time_of_sunrise, time_of_sunset, location_id)
VALUES
(
'Горы',
'0001-01-01 06:00:00',
'0001-01-01 19:00:00',
1

```

```

);
INSERT INTO point_of_interest (type, color, is_important, name)
VALUES
(
    'Крепость',
    'Черный',
    TRUE,
    'Барад - Дур'
);

INSERT INTO ship_to_flight (flight_id, ship_id)
VALUES
(
    1,
    1
);

END;

```

Выводы

В ходе лабораторной работы я попробовал придумать структуру базы данных и представить ее на инфологической и даталогической модели. Также, я изучил основные команды для управления таблицами в SQL.