

Министерство высшего образования и науки Российской Федерации  
Национальный научно-исследовательский университет ИТМО  
Факультет программной инженерии и компьютерной техники

**ЛАБОРАТОРНАЯ РАБОТА №1**

по дисциплине

**«БАЗЫ ДАННЫХ».**

Вариант №726.

Выполнил:

Петров Вячеслав Маркович,

Студент группы Р3108.

Преподаватель:

Афанасьев Дмитрий Борисович

Санкт-Петербург, 2024

## Оглавление

Текст задания.....	3
Описание предметной области.....	3
Список сущностей и их классификация .....	4
Инфологическая модель .....	4
Даталогическая модель .....	5
Реализация даталогической модели на SQL.....	5
Выводы по работе .....	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.

## Описание предметной области

Во всех коммюнике и документах Агентство по астронавтике старательно избегало слова "выручка", поскольку оно подразумевало какую-то неудачу или аварию: общепринятой формулой было "возвращение". Если случится что-либо действительно серьезное, спасти людей, конечно, не удастся: полтора миллиарда, километров от Земли - расстояние нешуточное.

# Список сущностей и их классификация

Стрежневые:

- Sphere - name;
- Organization – name, location;
- Field – name, description;
- Distance – kilometres, is\_significant, from\_what.

Характеристические:

- Incident – description, organization, sphere, is\_seriously, will\_people\_be\_saved, distance;
- Action\_with\_word – name, unpleasant\_word, is\_everywhere;  
Unpleasant\_word – name, reason, replacement;
- Reason – argumentation, concept.

Ассоциативные:

- Field\_Action – field, action;
- Action\_Owner – organization, field;
- Sphere\_Organization – sphere, organization, is\_famous.

## Инфологическая модель

Ниже представлена инфологическая модель (Рисунок 1), разработанная на основе задания.

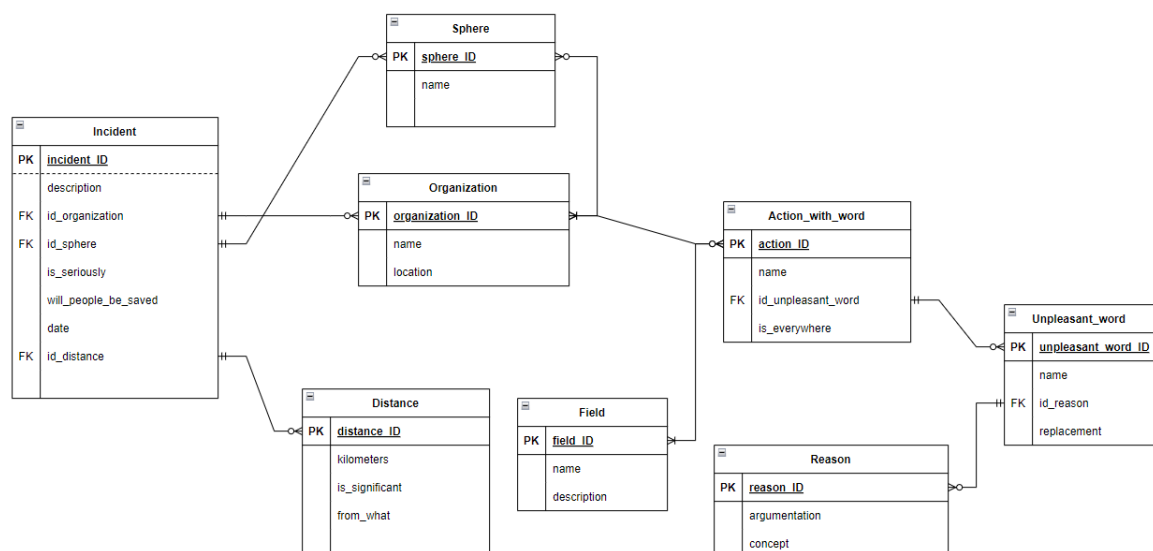


Рисунок 1. Инфологическая модель.

## Даталогическая модель

Ниже представлена даталогическая модель (Рисунок 2), разработанная на основе задания.

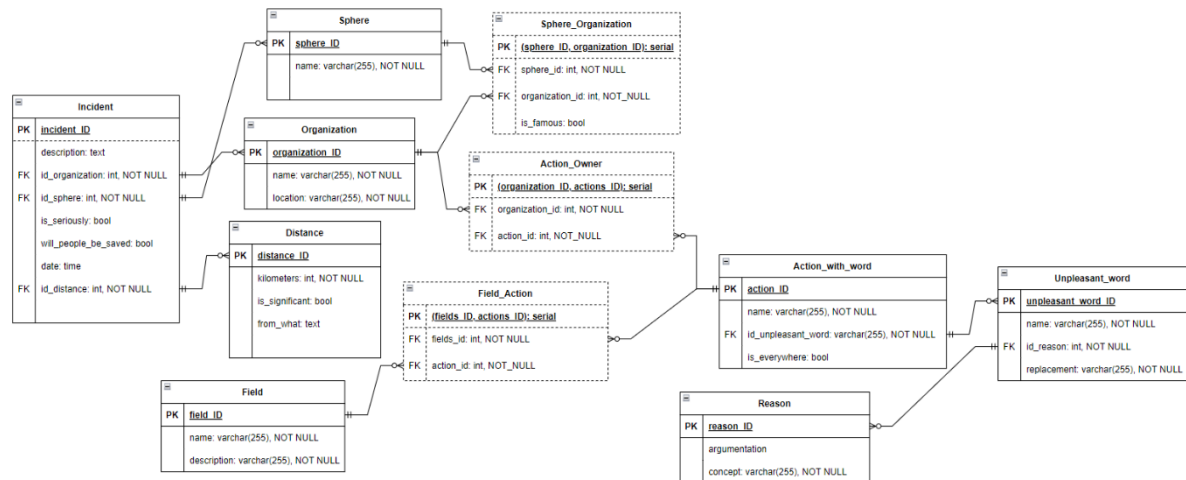


Рисунок 2. Даталогическая модель.

## Реализация даталогической модели на SQL

BEGIN;

```
CREATE TABLE IF NOT EXISTS sphere
```

```
(
    sphere_ID SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    PRIMARY KEY (sphere_ID, name)
);
```

```
CREATE TABLE IF NOT EXISTS organization
```

```
(
    organization_ID SERIAL PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    location VARCHAR(255) NOT NULL,
    PRIMARY KEY (organization_ID, name)
);
```

```
CREATE TABLE IF NOT EXISTS sphere_organization
```

```
(
    sphere_id INT REFERENCES sphere (sphere_ID),
    organization_id INT REFERENCES organization (organization_ID),
    is_famous BOOLEAN NOT NULL,
    PRIMARY KEY (sphere_id, organization_id)
);
```

```
CREATE TABLE IF NOT EXISTS distance
```

```
(
    distance_ID SERIAL PRIMARY KEY,
```

```

kilometers INT CHECK (kilometers >= 0) NOT NULL,
is_significant BOOLEAN NOT NULL,
from_what TEXT NOT NULL
);
CREATE TABLE IF NOT EXISTS incident
(
incident_ID SERIAL PRIMARY KEY,
description VARCHAR(255) NOT NULL DEFAULT 'кое-где',
id_organization INT NOT NULL REFERENCES organization (organization_ID),
id_sphere INT NOT NULL REFERENCES sphere (sphere_ID),
is_seriously BOOLEAN NOT NULL,
will_people_be_saved BOOLEAN NOT NULL,
date TIME NOT NULL,
id_distance INT NOT NULL REFERENCES distance (distance_ID)
);
CREATE TABLE IF NOT EXISTS reason
(
reason_ID SERIAL PRIMARY KEY,
argumentation TEXT NOT NULL,
concept VARCHAR(255) NOT NULL
);
CREATE TABLE IF NOT EXISTS unpleasant_word
(
unpleasant_word_ID SERIAL PRIMARY KEY,
name VARCHAR(255) NOT NULL,
id_reason INT REFERENCES reason (reason_ID),
replacement VARCHAR(50) NOT NULL
);
CREATE TABLE IF NOT EXISTS action_with_word
(
action_ID SERIAL PRIMARY KEY,
name VARCHAR(255) NOT NULL,
id_unpleasant_word INT REFERENCES unpleasant_word (unpleasant_word_ID),
is_everywhere BOOLEAN NOT NULL
);
CREATE TABLE IF NOT EXISTS field
(
field_ID SERIAL PRIMARY KEY,
name TEXT NOT NULL,
description TEXT NOT NULL,
PRIMARY KEY (field_ID, name)
);
CREATE TABLE IF NOT EXISTS action_owner
(
organization_id INT REFERENCES organization (organization_ID),
action_id INT REFERENCES action_with_word (action_ID),
PRIMARY KEY (organization_id, action_id)
);

```

```

CREATE TABLE IF NOT EXISTS field_action
(
    field_id INT REFERENCES field (field_ID),
    action_id INT REFERENCES action_with_word (action_ID),
    PRIMARY KEY (field_id, action_id)
);
INSERT INTO sphere(name)
VALUES ('космонавтика');
INSERT INTO organization(name, location)
VALUES ('агенство', 'США');
INSERT INTO field(name, description)
VALUES ('документы', 'зафиксированная на материальном носителе информация');
INSERT INTO field(name, description)
VALUES ('коммюнике', 'официальное сообщение');
INSERT INTO reason(argumentation, concept)
VALUES ('подразумевает', 'неудача');
INSERT INTO reason(argumentation, concept)
VALUES ('подразумевает', 'авария');
INSERT INTO unpleasant_word(name, id_reason, replacement)
VALUES ('выручка', 1, 'возвращение');
INSERT INTO unpleasant_word(name, id_reason, replacement)
VALUES ('выручка', 2, 'возвращение');
INSERT INTO action_with_word(name, id_unpleasant_word, is_everywhere)
VALUES ('избегать', 1, 'TRUE');
INSERT INTO action_with_word(name, id_unpleasant_word, is_everywhere)
VALUES ('избегать', 2, 'TRUE');
INSERT INTO distance(kilometers, is_significant, from_what)
VALUES (1500000, 'TRUE', 'Земля');
INSERT INTO incident(description, id_organization, id_sphere, is_seriously,
will_people_be_saved, id_distance)
VALUES (default, 1, 1, 'TRUE', 'FALSE', 1);
INSERT INTO sphere_organization(sphere_id, organization_id, is_famous)
VALUES (1, 1, 'TRUE');
INSERT INTO field_action(field_id, action_id)
VALUES (1, 1);
INSERT INTO field_action(field_id, action_id)
VALUES (2, 1);
INSERT INTO field_action(field_id, action_id)
VALUES (1, 2);
INSERT INTO field_action(field_id, action_id)
VALUES (2, 2);
INSERT INTO action_owner(organization_id, action_id)
VALUES (1, 1);
INSERT INTO action_owner(organization_id, action_id)
VALUES (1, 2);

END;

```

## Выводы по работе

В ходе данной лабораторной работы я потренировался продумывать структуру базы данных и представлять её в виде диаграмм (инфологической и даталогической моделей), а также изучил основные команды для создания и удаления таблиц в SQL.