**Федеральное государственное бюджетное образовательное учреждение высшего образования «Санкт-Петербургский государственный университет»**

**Экономический факультет**

**Базы данных**

**Лабораторная работа №1**

Студент:

Кустарев Александр Павлович Б01-э

Преподаватель:

Иванова Виктория Валерьевна

Оценка: \_\_\_\_\_\_\_\_\_\_

Подпись преподавателя: \_\_\_\_\_\_\_\_\_\_

Санкт-Петербург 2024

### 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.

Каждый студент должен использовать свою схему при работе над лабораторной работой №1

### Описание предметной области.

Существуют страны, к которым принадлежат космические миссии.

В разных странах есть департаменты внешних взаимодействий.

Существуют космические корабли, которые принадлежат космическим миссиям разных стран (не обязательно только одной миссии), такие как Дискавери и Цянь.

На космических кораблях служат экипажи из людей, которые принадлежат к разным странам

(не обязательно к тем, к которым принадлежит корабль).

На земле есть телескопы, которыми могут пользоваться разные космические мисси и которые видят некоторые космические корабли.

Список сущностей и их классификация

Стержневые:

* Departament
* Country
* Spacemission
* Spaceship
* Telescope
* Crew\_member
* Scientist

Ассоциативные:

* Spaceship\_spacemission
* Telescope\_spacemission
* Telescope\_scientist

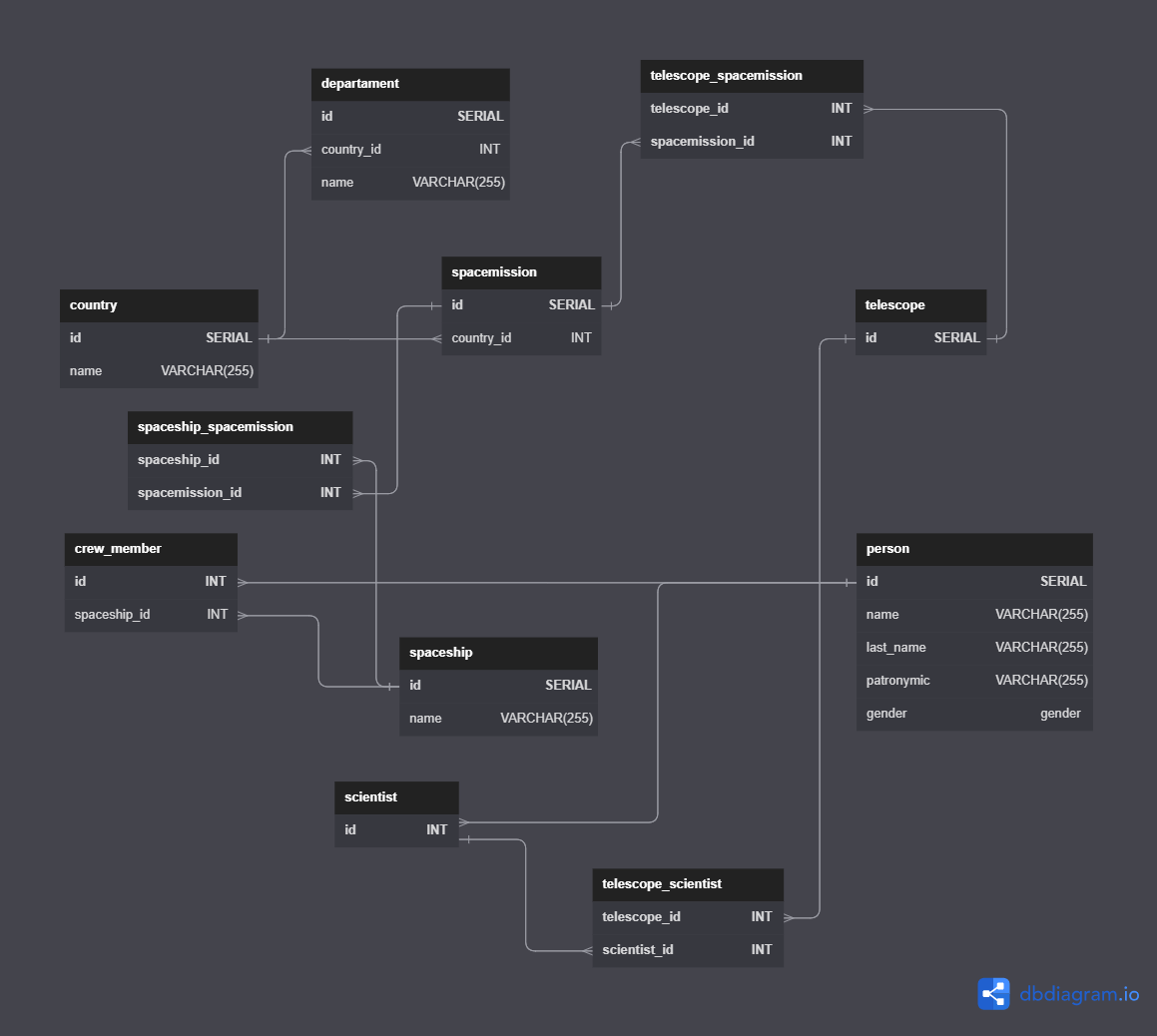
Характеристические:

* Person

### Инфологическая модель



### Даталогическая модель



### Реализация даталогической модели на SQL

CREATE TABLE country

(

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL

);

CREATE TABLE departament

(

id SERIAL PRIMARY KEY,

country\_id INT NOT NULL,

name VARCHAR(255) NOT NULL

);

ALTER TABLE departament ADD CONSTRAINT fk\_country

FOREIGN KEY(country\_id)

REFERENCES country(id) ON DELETE CASCADE;

CREATE TABLE spacemission

(

id SERIAL PRIMARY KEY,

country\_id INT NOT NULL

);

ALTER TABLE spacemission ADD CONSTRAINT fk\_country\_space

FOREIGN KEY(country\_id)

REFERENCES country(id) ON DELETE CASCADE;

CREATE TABLE telescope

(

id SERIAL PRIMARY KEY

);

CREATE TABLE telescope\_spacemission

(

telescope\_id INT NOT NULL,

spacemission\_id INT NOT NULL,

PRIMARY KEY(telescope\_id, spacemission\_id),

CONSTRAINT fk\_telescope

FOREIGN KEY(telescope\_id)

REFERENCES telescope(id) ON DELETE CASCADE,

CONSTRAINT fk\_spacemission

FOREIGN KEY(spacemission\_id)

REFERENCES spacemission(id) ON DELETE CASCADE

);

CREATE TABLE spaceship

(

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL

);

CREATE TABLE spaceship\_spacemission

(

spaceship\_id INT NOT NULL,

spacemission\_id INT NOT NULL,

PRIMARY KEY(spaceship\_id, spacemission\_id),

CONSTRAINT fk\_spaceship

FOREIGN KEY(spaceship\_id)

REFERENCES spaceship(id) ON DELETE CASCADE,

CONSTRAINT fk\_spacemission

FOREIGN KEY(spacemission\_id)

REFERENCES spacemission(id) ON DELETE CASCADE

);

CREATE TYPE gender AS ENUM ('MALE', 'FEMALE');

CREATE TABLE person

(

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

last\_name VARCHAR(255) NOT NULL,

patronymic VARCHAR(255),

gender gender

);

CREATE TABLE crew\_member

(

id INT PRIMARY KEY,

spaceship\_id INT NOT NULL,

CONSTRAINT fk\_person

FOREIGN KEY(id)

REFERENCES person(id) ON DELETE CASCADE,

CONSTRAINT fk\_spaceship

FOREIGN KEY(spaceship\_id)

REFERENCES spaceship(id) ON DELETE CASCADE

);

CREATE TABLE scientist

(

id INT PRIMARY KEY,

CONSTRAINT fk\_person

FOREIGN KEY(id)

REFERENCES person(id) ON DELETE CASCADE

);

CREATE TABLE telescope\_scientist

(

telescope\_id INT NOT NULL,

scientist\_id INT NOT NULL,

PRIMARY KEY(telescope\_id, scientist\_id),

CONSTRAINT fk\_telescope

FOREIGN KEY(telescope\_id)

REFERENCES telescope(id) ON DELETE CASCADE,

CONSTRAINT fk\_scientist

FOREIGN KEY(scientist\_id)

REFERENCES scientist(id) ON DELETE CASCADE

);

### Заполнение таблиц запросами SQL

INSERT INTO country (name) VALUES ('USA');

INSERT INTO country (name) VALUES ('CHINA');

INSERT INTO country (name) VALUES ('RUSSIA');

INSERT INTO departament (country\_id, name)

VALUES ((SELECT id FROM country WHERE name='USA'), 'GOSDEP');

INSERT INTO departament (country\_id, name)

VALUES ((SELECT id FROM country WHERE name='CHINA'), 'PINYIN');

INSERT INTO departament (country\_id, name)

VALUES ((SELECT id FROM country WHERE name='RUSSIA'), 'ROSCOSMOS');

INSERT INTO spacemission (country\_id)

VALUES ((SELECT id FROM country WHERE name='USA'));

INSERT INTO spacemission (country\_id)

VALUES ((SELECT id FROM country WHERE name='CHINA'));

INSERT INTO spacemission (country\_id)

VALUES ((SELECT id FROM country WHERE name='RUSSIA'));

INSERT INTO telescope DEFAULT VALUES;

INSERT INTO telescope DEFAULT VALUES;

INSERT INTO telescope DEFAULT VALUES;

INSERT INTO telescope\_spacemission (telescope\_id, spacemission\_id)

VALUES(

(SELECT t.id FROM telescope as t WHERE id=1),

(SELECT s.id FROM spacemission as s WHERE id=1)

);

INSERT INTO telescope\_spacemission (telescope\_id, spacemission\_id)

VALUES(

(SELECT t.id FROM telescope as t WHERE id=2),

(SELECT s.id FROM spacemission as s WHERE id=2)

);

INSERT INTO telescope\_spacemission (telescope\_id, spacemission\_id)

VALUES(

(SELECT t.id FROM telescope as t WHERE id=3),

(SELECT s.id FROM spacemission as s WHERE id=3)

);

INSERT INTO spaceship (name) VALUES ('OREON');

INSERT INTO spaceship (name) VALUES ('СHINA\_LYOT');

INSERT INTO spaceship (name) VALUES ('RUSSO\_LYOT');

INSERT INTO spaceship\_spacemission (spaceship\_id, spacemission\_id)

VALUES (

(SELECT s.id FROM spaceship as s WHERE id=1),

(SELECT s.id FROM spacemission as s WHERE id=1)

);

INSERT INTO spaceship\_spacemission (spaceship\_id, spacemission\_id)

VALUES (

(SELECT s.id FROM spaceship as s WHERE id=2),

(SELECT s.id FROM spacemission as s WHERE id=2)

);

INSERT INTO spaceship\_spacemission (spaceship\_id, spacemission\_id)

VALUES (

(SELECT s.id FROM spaceship as s WHERE id=3),

(SELECT s.id FROM spacemission as s WHERE id=3)

);

INSERT INTO person(name, last\_name, patronymic, gender)

VALUES ('Erik', 'Karapetyan', 'Akopovich', 'FEMALE');

INSERT INTO person(name, last\_name, patronymic, gender)

VALUES ('Ivan', 'Kustarev', 'Pavlovich', 'MALE');

INSERT INTO person(name, last\_name, patronymic, gender)

VALUES ('Pavel', 'Danilov', 'Alexandrovich', 'MALE');

INSERT INTO person(name, last\_name, patronymic, gender)

VALUES ('Konstantin', 'Zinchenko', 'Petrovich', 'MALE');

INSERT INTO crew\_member(id, spaceship\_id) VALUES(

(SELECT p.id FROM person as p WHERE id=1),

(SELECT s.id FROM spaceship as s WHERE id=1)

);

INSERT INTO crew\_member(id, spaceship\_id) VALUES(

(SELECT p.id FROM person as p WHERE id=2),

(SELECT s.id FROM spaceship as s WHERE id=2)

);

INSERT INTO crew\_member(id, spaceship\_id) VALUES(

(SELECT p.id FROM person as p WHERE id=3),

(SELECT s.id FROM spaceship as s WHERE id=3)

);

INSERT INTO scientist(id) VALUES(

(SELECT p.id FROM person as p WHERE id=4)

);

INSERT INTO telescope\_scientist(telescope\_id, scientist\_id) VALUES(

(SELECT t.id FROM telescope as t WHERE id=1),

(SELECT s.id FROM scientist as s WHERE id=4)

);

### Выводы:

В ходе выполнения данной лабораторной работы я изучил и реализовал на практике процесс проектирования БД: создание инфологической и даталогической модели по предметной области и реализация даталогической модели SQL-запросами к PSQL-базе. Также я изучил различные виды сущностей и их связей, ограничения целостности, связанные с сущностями, и их реализацию.