Санкт-Петербургский Национальный Исследовательский Университет ИТМО

Факультет Программной Инженерии и Компьютерной Техники



Вариант №492541

Лабораторная работа №1

По дисциплине

Базы Данных

Выполнил студент группы P3117:

Изаак Герман Константинович

Преподаватель:

Чупанов Аликылыч Алибекович

Санкт-Петербург 2025 г.

1. **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.
6. **Описание предметной области**

**Описание предметной области, по которой должна быть построена доменная модель:**

|  |
| --- |
| И тут внизу, прямо под Боуменом, появилось нечто совершенно новое, чего раньше не было, потому что проглядеть это было невозможно. По океану раскаленного газа плыли мириады светящихся бусинок, от которых исходило жемчужное сияние; каждые несколько секунд оно то вспыхивало, то гасло. Все бусинки двигались в одном направлении, словно стая лососей, идущая на нерест вверх по течению реки; порой они отклонялись то вправо, то влево, так что пути их пересекались, но ни разу не коснулись друг друга. |

На некоторой локации описывается поведение объектов, обладающих различными характеристиками: яркостью, скоростью и направлением движения. Эти объекты взаимодействуют между собой, собираются в группы. Их поведение может быть цикличным. Движение объектов происходит с определённой закономерностью, и их характеристики могут изменяться во времени.

1. **Список сущностей и их классификация**

Стержневые:

* Fish – локация, группа, вес, размер
* Location – название, координаты, активность
* Cycle – тип, частота, продолжительность
* Group – название, траектория, ассоциация, количество участников

Характеристические:

* Trajectory – время начала, время окончания, локация отправления, локация прибытия
* State – яркость, скорость, время фиксации состояния, объект, температура

Ассоциативные:

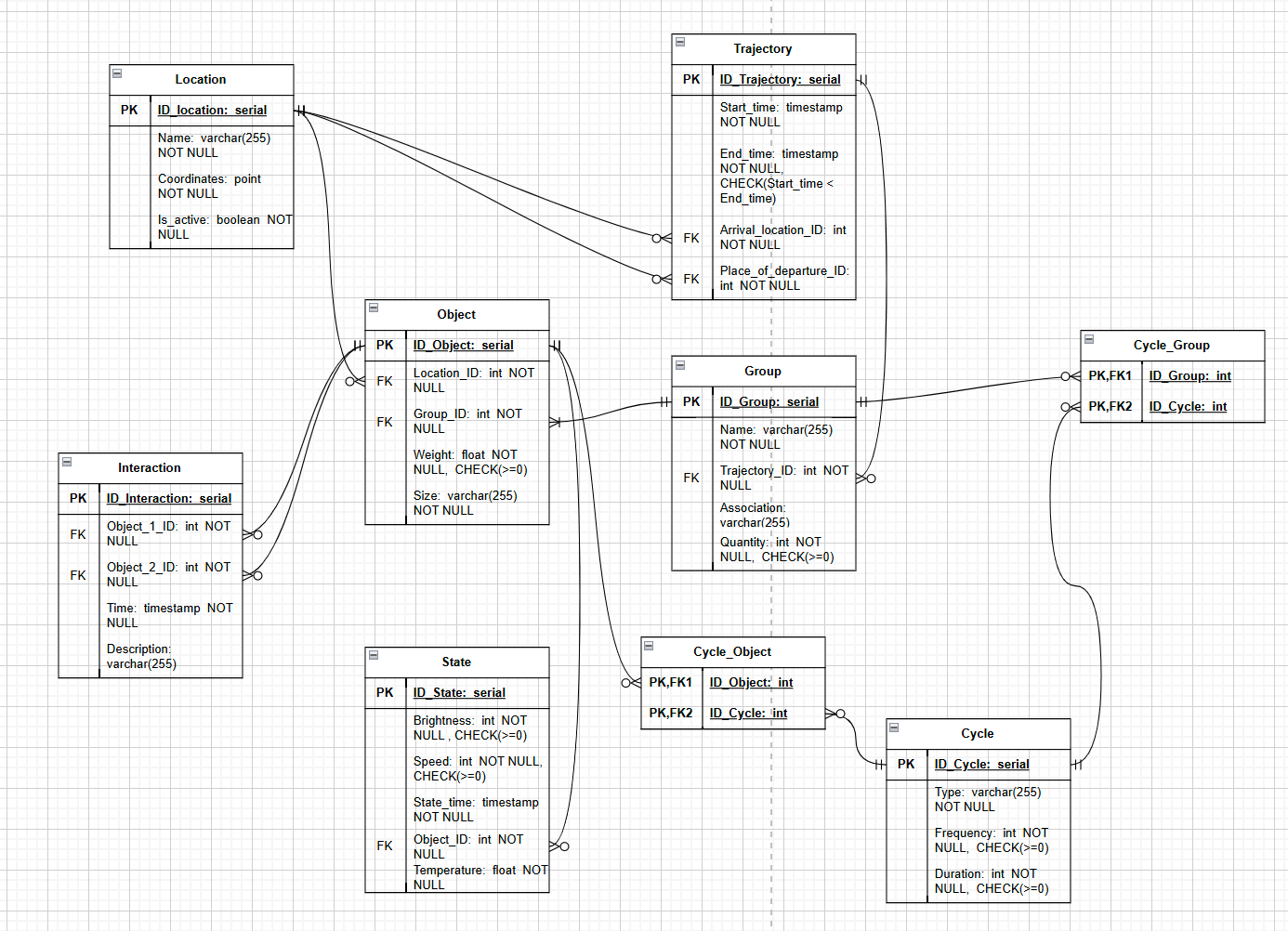
* Interaction – время взаимодействия, описание, первый объект взаимодействия, второй объект
* Cycle\_Group – связь между группой и циклом
* Cycle\_Fish – связь между объектом и циклом

1. **Инфологическая модель**

**Изображение выглядит как текст, диаграмма, линия, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.**

1. **Даталогическая модель**

** **

1. **Реализация даталогической модели на SQL**

CREATE TABLE IF NOT EXISTS location (  
 id\_location SERIAL PRIMARY KEY,  
 name VARCHAR(255) NOT NULL,  
 coordinates POINT NOT NULL,  
 is\_active BOOLEAN NOT NULL  
);  
  
CREATE TABLE IF NOT EXISTS trajectory (  
 id\_trajectory SERIAL PRIMARY KEY,  
 start\_time TIMESTAMP NOT NULL,  
 end\_time TIMESTAMP NOT NULL,  
 arrival\_location\_id INT NOT NULL,  
 place\_of\_departure\_id INT NOT NULL,  
 FOREIGN KEY (arrival\_location\_id) REFERENCES location(id\_location) ON DELETE CASCADE,  
 FOREIGN KEY (place\_of\_departure\_id) REFERENCES location(id\_location) ON DELETE CASCADE,  
  
 CHECK (start\_time < end\_time)  
);  
  
CREATE TABLE IF NOT EXISTS groups (  
 id\_group SERIAL PRIMARY KEY,  
 name VARCHAR(255) NOT NULL,  
 association VARCHAR(255),  
 quantity INT NOT NULL,  
 trajectory\_id INT NOT NULL,  
 FOREIGN KEY (trajectory\_id) REFERENCES trajectory(id\_trajectory) ON DELETE CASCADE,  
  
 CHECK (quantity >= 0)  
);  
  
CREATE TABLE IF NOT EXISTS fish (  
 id\_fish SERIAL PRIMARY KEY,  
 weight FLOAT NOT NULL,  
 size VARCHAR(255) NOT NULL,  
 location\_id INT NOT NULL,  
 group\_id INT NOT NULL,  
 FOREIGN KEY (location\_id) REFERENCES location(id\_location) ON DELETE CASCADE,  
 FOREIGN KEY (group\_id) REFERENCES groups(id\_group) ON DELETE CASCADE,  
  
 CHECK (weight >= 0)  
);  
  
CREATE TABLE IF NOT EXISTS interaction (  
 id\_interaction SERIAL PRIMARY KEY,  
 time TIMESTAMP NOT NULL,  
 description VARCHAR(255),  
 fish\_1\_id INT NOT NULL,  
 fish\_2\_id INT NOT NULL,  
 FOREIGN KEY (fish\_1\_id) REFERENCES fish(id\_fish) ON DELETE CASCADE,  
 FOREIGN KEY (fish\_2\_id) REFERENCES fish(id\_fish) ON DELETE CASCADE  
);  
  
CREATE TABLE IF NOT EXISTS state (  
 id\_tate SERIAL PRIMARY KEY,  
 brightness INT NOT NULL,  
 speed INT NOT NULL,  
 state\_time TIMESTAMP NOT NULL,  
 temperature FLOAT NOT NULL,  
 fish\_id INT NOT NULL,  
 FOREIGN KEY (fish\_id) REFERENCES fish(id\_fish) ON DELETE CASCADE,  
  
 CHECK (brightness >= 0),  
 CHECK (speed >= 0)  
);  
  
CREATE TABLE IF NOT EXISTS cycle (  
 id\_cycle SERIAL PRIMARY KEY,  
 type VARCHAR(255) NOT NULL,  
 frequency INT NOT NULL ,  
 duration INT NOT NULL,  
  
 CHECK (frequency >= 0),  
 CHECK (duration >= 0)  
);  
  
CREATE TABLE IF NOT EXISTS cycle\_group (  
 id\_group INT NOT NULL,  
 id\_cycle INT NOT NULL,  
 PRIMARY KEY (id\_cycle, id\_group),  
 FOREIGN KEY (id\_group) REFERENCES groups(id\_group) ON DELETE CASCADE,  
 FOREIGN KEY (id\_cycle) REFERENCES cycle(id\_cycle) ON DELETE CASCADE  
);  
  
CREATE TABLE IF NOT EXISTS cycle\_fish (  
 id\_fish INT NOT NULL,  
 id\_cycle INT NOT NULL,  
 PRIMARY KEY (id\_fish, id\_cycle),  
 FOREIGN KEY (id\_fish) REFERENCES fish(id\_fish) ON DELETE CASCADE,  
 FOREIGN KEY (id\_cycle) REFERENCES cycle(id\_cycle) ON DELETE CASCADE  
);  
  
--test data  
  
INSERT INTO location (name, coordinates, is\_active)  
VALUES  
 ('Central Park', *POINT*(40.785091, -73), TRUE),  
 ('Times Square', *POINT*(22.3424, -7.3), TRUE),  
 ('Brooklyn Bridge', *POINT*(1.232, 35.56456), TRUE),  
 ('Empire State Building', *POINT*(-31.666, 228.337), FALSE);  
  
INSERT INTO trajectory (start\_time, end\_time, arrival\_location\_id, place\_of\_departure\_id)  
VALUES  
 ('1998-02-25 08:12:52', '2022-02-24 4:00:00', 1, 2),  
 ('2001-09-11 12:45:01', '2025-02-26 11:00:00', 3, 1),  
 ('2025-02-27 4:20:00', '2025-02-27 16:00:00', 3, 2);  
  
INSERT INTO groups (name, association, quantity, trajectory\_id)  
VALUES  
 ('Explorers', 'Tourist', 10, 1),  
 ('Photographers', NULL, 5, 2),  
 ('Historians', 'Research', 8, 3);  
  
  
INSERT INTO fish (weight, size, location\_id, group\_id)  
VALUES  
 (5.5, 'Small', 1, 1),  
 (12.0, 'Medium', 2, 1),  
 (8.3, 'Large', 3, 2),  
 (4.0, 'Small', 2, 3);  
  
INSERT INTO interaction (time, description, fish\_1\_id, fish\_2\_id)  
VALUES  
 ('2024-02-25 09:30:00', 'fish 1 and 2 interacted near Central Park', 1, 2),  
 ('2025-02-26 10:15:00', 'fish 3 observed fish 4 near Brooklyn Bridge', 3, 4);  
  
INSERT INTO state (brightness, speed, state\_time, temperature, fish\_id)  
VALUES  
 (80, 10, '2005-05-5 14:38:018', 22.5, 1),  
 (90, 15, '2019-07-12 12:37:32', 21.0, 2),  
 (70, 8, '2015-08-13 15:30:07', -19.0, 3);  
  
INSERT INTO cycle (type, frequency, duration)  
VALUES  
 ('flashing', 24, 60),  
 ('torsion', 7, 420),  
 ('sound', 30, 1800);  
  
INSERT INTO cycle\_group (id\_group, id\_cycle)  
VALUES  
 (1, 1),  
 (2, 2),  
 (3, 3);  
  
INSERT INTO cycle\_fish (id\_fish, id\_cycle)  
VALUES  
 (1, 1),  
 (2, 1),  
 (3, 2),  
 (4, 3);  
  
--SELECT \* FROM fish ORDER BY size DESC LIMIT 2;  
--hEeEfRc7ld0Q22xT

1. **Вывод**

При выполнении лабораторной работы я познакомился с принципом проектирования «Top – Down», научился составлять инфологическую и даталогическую модели сущностей, на основе которых реализовал базу данных с использованием языка SQL. В процессе работы я освоил создание таблиц, установление связей между ними, применение ограничений и выполнение запросов для наполнения и проверки целостности данных. Полученные знания и навыки позволили мне успешно спроектировать и реализовать базу данных, отвечающую требованиям предметной области.