

Федеральное государственное автономное
образовательное учреждение высшего образования
«Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики

Факультет программной инженерии и компьютерной техники

Лабораторная работа №3

по базам данных

Выполнила: Гафурова Фарангиз

Фуркатовна

Группа: Р3120

Принял: Николаев Владимир Вячеславович

г. Санкт-Петербург, 2024г

1. Текст задания

Для отношений, полученных при построении предметной области из лабораторной работы №1, выполните следующие действия:

- опишите функциональные зависимости для отношений полученной схемы (минимальное множество);
- приведите отношения в 3NF (как минимум). Постройте схему на основе 3NF (как минимум). Постройте схему на основе полученных отношений;
- опишите изменения в функциональных зависимостях, произошедшие после преобразования в 3NF (как минимум). Постройте схему на основе 3NF;
- преобразуйте отношения в BCNF. Докажите, что полученные отношения представлены в BCNF;

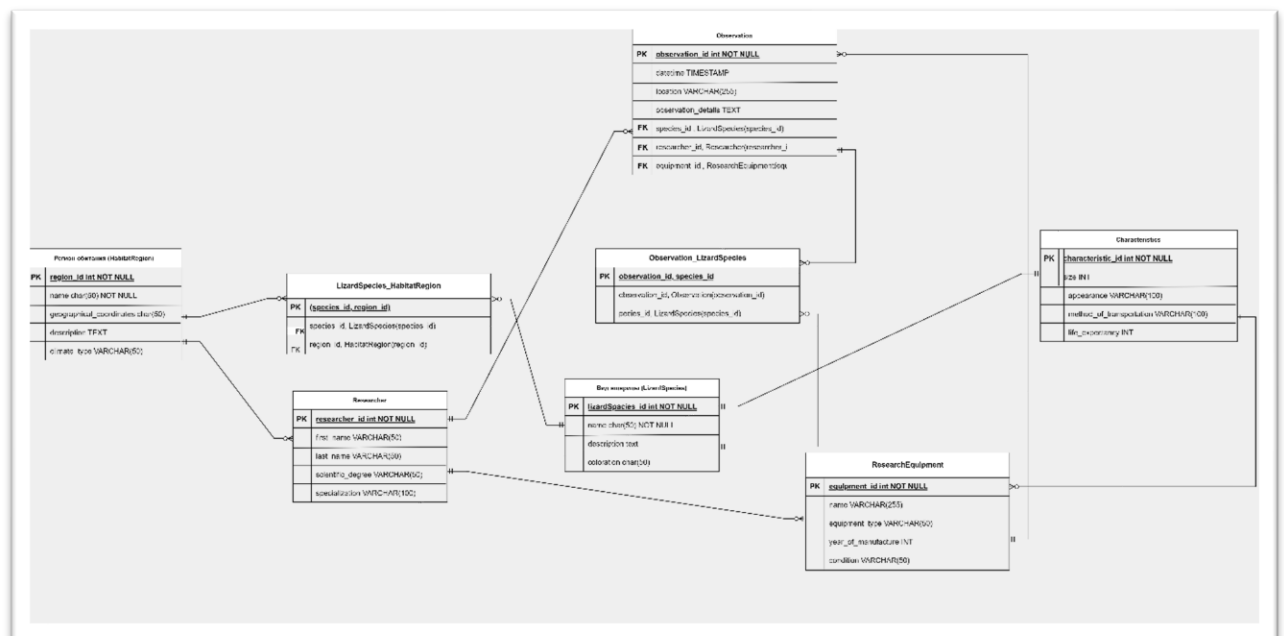
Если ваша схема находится уже в BCNF, докажите это.

Какие денормализации будут полезны для вашей схемы? Приведите подробное описание;

Придумайте функцию, связанную с вашей предметной областью, согласуйте ее с преподавателем и реализуйте на языке PL/pgSQL.

Описание предметной области

Он объяснил, что из шести тысяч видов ящериц, существующих в мире, чуть больше дюжины передвигаются в вертикальном положении. И только четыре вида из них обитают в Латинской Америке. И, судя по окраске, данная ящерица может принадлежать только к одному виду из этих четырех.



2. Функциональные зависимости

LizardSpecies:

id → name

id → description

id → coloration

name → id

HabitatRegion:

id → name

id → geographical_coordinates

id → description

id → climate_type

name → id

Researcher:

id → first_name, last_name, scientific_degree, specialization

ResearchEquipment:

equipment_id → name

equipment_id → equipment_type

equipment_id → year_of_manufacture

equipment_id → condition

equipment_id → researcher_id

Characteristics:

id → size

id → appearance

id → method_of_transportation

id → life_expectancy

Observation:

id → datetime

id → location

id → observation_details

id → species_id

id → researcher_id

id → equipment_id

3. Приведение отношений в 3NF и описание изменений

1NF: Отношение находится в 1NF, если все его атрибуты содержат только атомарные значения. Моя модель удовлетворяет 1NF, так как все атрибуты атомарны, и нет повторяющихся групп.

2NF: Отношение находится во 2NF, если оно находится в 1NF и все его неключевые атрибуты полностью функционально зависят от первичного ключа. Моя модель удовлетворяет 2NF, так как все неключевые атрибуты полностью функционально зависят от первичных ключей.

3NF: Отношение находится в 3NF, если оно находится во 2NF и не содержит транзитивных зависимостей. Моя модель удовлетворяет 3NF, так как все неключевые атрибуты зависят только от первичных ключей, и не содержат транзитивных зависимостей.

4. Преобразование в BCNF и доказательство

Отношение находится в BCNF, если для каждой функциональной зависимости $X \rightarrow Y$, X является суперключом. Моя модель удовлетворяет BCNF, так как для всех функциональных зависимостей X является суперключом.

5. Денормализация

В качестве денормализации можно увеличить скорость выполнения некоторых запросов к нашей БД за счет сокращения количества сущностей, а именно – объединение таблиц `Characteristics` и `LizardSpecies` в одну `LizardSpecies`.

Это уменьшит количество соединений в запросах, но в новой таблице появятся избыточные данные, поэтому будет лучше оставить всё как есть.

6. Функция на языке PL/pgSQL

```
CREATE OR REPLACE FUNCTION check_equipment_condition_and_age() RETURNS TRIGGER AS
$$
DECLARE
    equipment_condition VARCHAR(50);
    equipment_age INT;
    current_year INT := EXTRACT(YEAR FROM CURRENT_DATE);
BEGIN
    -- Извлекаем состояние и год выпуска оборудования из таблицы ResearchEquipment
    SELECT condition, year_of_manufacture INTO equipment_condition, equipment_age
    FROM ResearchEquipment
    WHERE equipment_id = NEW.equipment_id;

    -- Вычисляем возраст оборудования
    equipment_age := current_year - equipment_age;

    -- Проверяем состояние оборудования
    IF equipment_condition NOT IN ('Excellent', 'Good') THEN
        RAISE EXCEPTION 'Equipment in unsuitable condition: %', equipment_condition;
    END IF;

    -- Проверяем возраст оборудования
    IF equipment_age > 5 THEN
        RAISE EXCEPTION 'Equipment is too old: % years', equipment_age;
    END IF;

    RETURN NEW;
END;
```

```
$$ LANGUAGE plpgsql;  
  
-- Создание триггера  
CREATE TRIGGER equipment_condition_check  
BEFORE INSERT OR UPDATE ON Observation  
FOR EACH ROW EXECUTE FUNCTION check_equipment_condition_and_age();
```

7. Вывод

По завершении лабораторной работы я научилась нормализации до 3NF и BCNF, разработке функциональных зависимостей, а также созданию триггеров и функций на PL/pgSQL для обеспечения целостности данных.