

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

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 дисциплине

«Базы данных»

Вариант № 312516

Выполнил:

Студент группы Р3125

Шадрухин Александр Сергеевич

Преподаватель:

Егошин Алексей Васильевич

Санкт-Петербург, 2023

Задание.

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

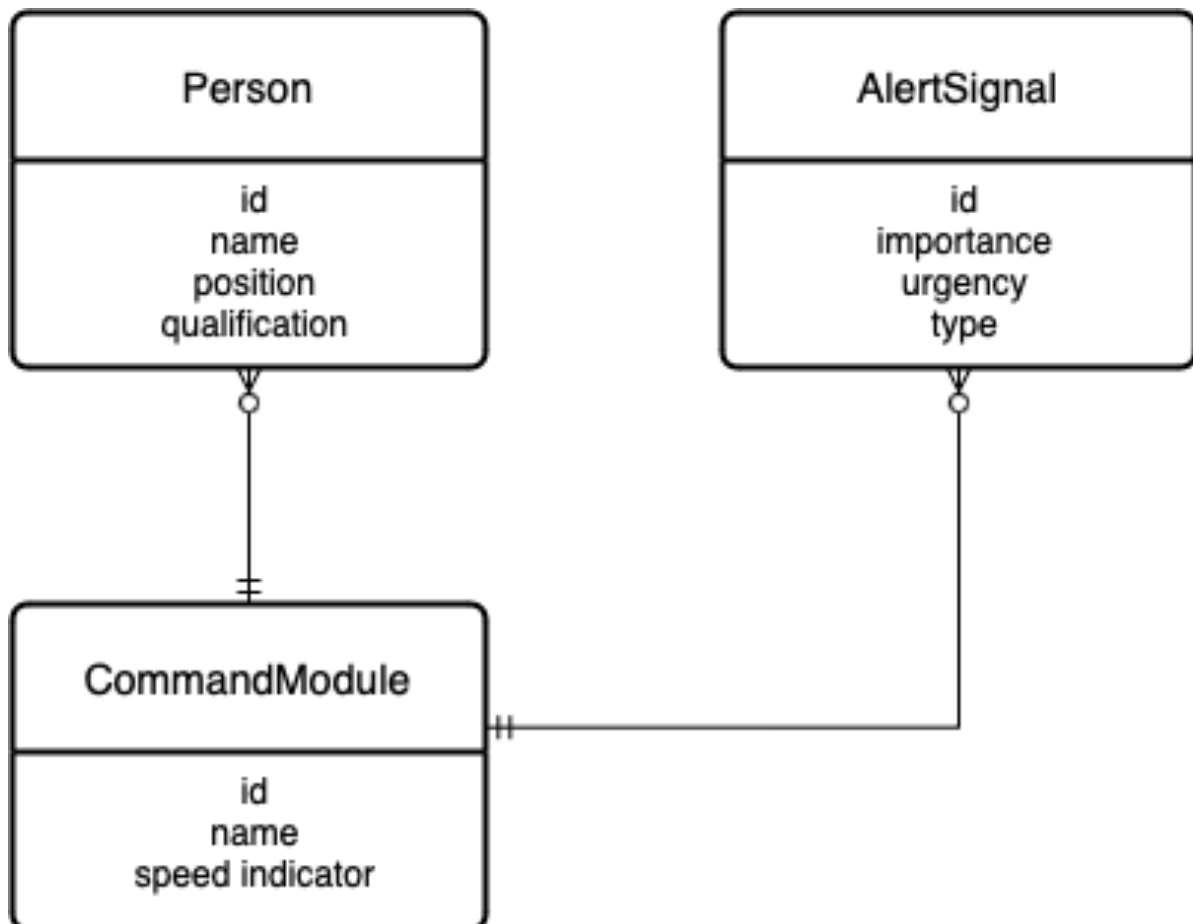
- опишите функциональные зависимости для отношений полученной схемы (минимальное множество);
- приведите отношения в 3NF (как минимум). Постройте схему на основе 3NF (как минимум);
- опишите изменения в функциональных зависимостях, произошедшие после преобразования в 3NF (как минимум). Постройте схему 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 триггер и связанную с ним функцию, относящиеся к вашей предметной области, согласуйте их с преподавателем и реализуйте на языке PL/pgSQL.

Инфологическая модель:



Даталогическая модель:

```
CREATE TABLE s367629.Person ( id SERIAL PRIMARY KEY,
name VARCHAR(31),
position VARCHAR(31), qualification INT, commandModuleId
INT,

FOREIGN KEY (commandModuleId) REFERENCES
CommandModule(id)

);

CREATE TABLE s367629.CommandModule (
```

```

id SERIAL PRIMARY KEY,
name VARCHAR(31),
speedIndicator INT
);

CREATE TABLE s367629.AlertSignal (
id SERIAL PRIMARY KEY,
importance INT,
urgency INT,
type VARCHAR(31),
commandModuleId INT,
FOREIGN KEY (commandModuleId) REFERENCES
CommandModule(id)
);

```

Функциональные зависимости:

person: $id \rightarrow (name, position, qualification, commandModule)$

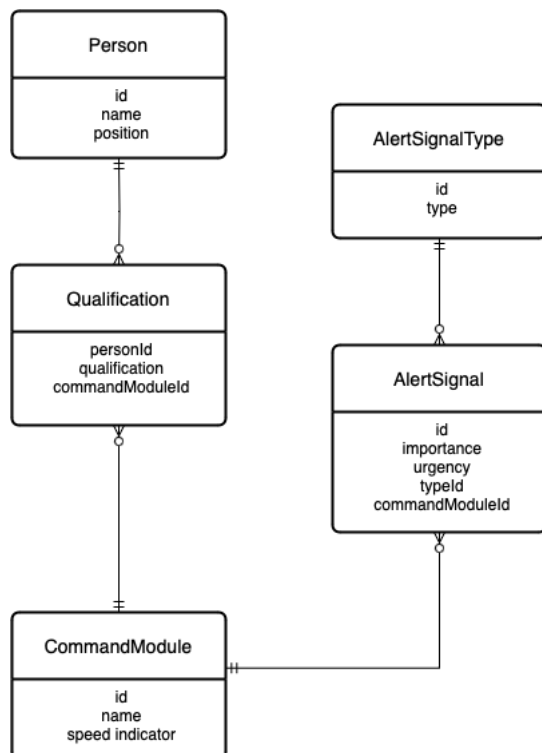
alertSignal: $id \rightarrow (importance, urgency, type, commandModule)$

commandModule: $id \rightarrow ()$

Нормальные формы:

1NF: Отношение находится в 1NF, если все его атрибуты содержат только атомарные значения. Моя модель удовлетворяет 1NF, потому что все атрибуты атомарные 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, когда находится во 2NF и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа. Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы. Моя модель удовлетворяет 3NF, так как все неключевые атрибуты зависят только от первичных ключей, и не содержат транзитивных зависимостей.

BCNF:

Нормальная форма Бойса-Кодда — это частная форма 3NF. Отношение находится в BCNF, когда каждая нетривиальная и неприводимая слева функциональная зависимость обладает потенциальным ключом в качестве детерминанта.

Моя модель удовлетворяет BCNF, потому что все первичные зависимости связаны с первичными и внешними ключами.

Денормализация:

1) Добавление поля "qualification" в таблицу Person: это позволит избежать лишних присоединений (JOIN), когда нужно получить информацию о квалификации конкретного человека.

2) Добавление поля "type signal" в таблицу AlertSignal: это позволит получать информацию о типе каждого сигнала без необходимости выполнения JOIN с таблицей AlertSignalType.

Триггер и функция:

```
CREATE OR REPLACE FUNCTION s367629.insert_alert_signal()
RETURNS TRIGGER AS
$$
BEGIN
    IF (SELECT speedIndicator FROM s367629.CommandModule WHERE
        id = NEW.commandModuleId) <= NEW.qualification THEN
        INSERT INTO s367629.AlertSignal (importance, urgency,
            typeId, commandModuleId)
            VALUES (5, 1, NEW.qualification, NEW.commandModuleId);
    END IF;

    RETURN NULL;
END;
$$
LANGUAGE plpgsql;

CREATE TRIGGER insert_alert_signal_trigger
AFTER INSERT ON s367629.Qualification
FOR EACH ROW
EXECUTE FUNCTION s367629.insert_alert_signal();
```

Вывод:

При выполнении лабораторной работы я познакомился с понятием нормализации и денормализации. Научился определять функциональные зависимости модели, а также анализировать последнюю на соответствие различным нормальным формам. Изучил эффективные способы денормализации схемы базы данных и ситуации, в которых возможно их применение.