**Федеральное государственное автономное образовательное**

**учреждение высшего образования**

**«Национальный исследовательский университет ИТМО»**

Факультет программной инженерии и компьютерной техники

Базы данных

**Лабораторная работа №3**

Вариант - 2277

**Студент**: Рощин Константин Эдуардович

**Группа:** P3122

**Преподаватель**: Горбунов Михаил Витальевич

САНКТ-ПЕТЕРБУРГ

2024

# **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 NF (как минимум).
* Опишите изменения в функциональных зависимостях, произошедшие после преобразования в 3NF (как минимум). Постройте схему на основе NF;
* 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.

**Текст**: Полночь на Юпитере! Флойд будет вспоминать это волшебное зрелище всю жизнь. А сейчас можно им наслаждаться, спокойно и беззаботно, поскольку случиться уже ничего не может. По крайней мере, он сделал для этого все, что было в его силах.

**Исходная** **Даталогическая модель**

Изображение выглядит как текст, линия, диаграмма, снимок экрана

Автоматически созданное описание

**Исходные функциональные зависимости**

Person:

* Id → person\_name, memory\_id

Memory:

* Id → storage

View:

* Id → location\_id, memory\_id, description

Location:

* Id → location\_name, curtime

Action:

* Action\_type\_id, person\_id → action\_time

Action\_type:

* Id → activity

Emotion\_state:

* Id → action\_id, emotions

Эти функциональные зависимости находятся в:

1. В 1НФ так как отсутствуют группы значений в одном поле в связи с уникальностью первичного ключа (ID, либо action\_type\_id, person\_id)
2. В 2НФ так как все атрибуты кроме первичного ключа полностью зависят от первичного ключа (ID, либо action\_type\_id, person\_id)
3. В 3НФ так как все атрибуты кроме первичного ключа не образуют транзитивных зависимостей
4. В НФБК так как все функциональные зависимости зависят от первичного ключа (в Action, два PK: action\_type\_id, person\_id они не зависят от неключевого атрибута – action\_time)

**Денормализация**

1) Можно денормализовать 1НФ у таблиц Action\_type и Emotion\_state. В таблице Action\_type можно будет сразу видеть какие эмоции вызывает действие. Чтобы узнать эмоции, которые вызывает какое-либо действие, не нужно будет использовать join для связи таблиц, так как всё будет в одной таблице, но возможны аномалии.

Новая таблица Action\_type:

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

2) То же самое можно сделать с таблицами View и Location, их можно соединить в одну таблицу View

Новая таблица View:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

**Функция и триггер**

Триггер location\_time\_trigger применяется перед обновлением записей таблицы location, атрибут curtime обновляется автоматически в зависимости от указанного часового пояса.

--функция

CREATE OR REPLACE FUNCTION update\_location\_time\_utc()

RETURNS TRIGGER AS $$

DECLARE

    timezone TEXT :='UTC';

BEGIN

    NEW.curtime := CURRENT\_TIMESTAMP AT TIME ZONE timezone;

    RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER location\_time\_trigger

BEFORE UPDATE ON Location

FOR EACH ROW EXECUTE FUNCTION update\_location\_time\_utc();

**Вывод**

В ходе этой лабораторной работы я разобрался с функциональными зависимостями, узнал про виды нормальных форм и денормализацию. Так же узнал, что такое триггер и функции на языке PL/pgSQL и как их создавать.