Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

**Отчет**

**по лабораторной работе №3**

по дисциплине «Базы данных»

Выполнил: Тахватулин М. В., группа Р3107

Преподаватель: Бострикова Д. К.

г. Санкт-Петербург

~ 2023 ~

**Оглавление**

[**Задание** 2](#_Toc136346042)

[**Триггер** 5](#_Toc136346043)

[**Вывод** 7](#_Toc136346044)

# **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). Постройте схему на основе полученных отношений;

• 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;

• какие денормализации будут полезны для вашей схемы? Приведите подробное описание;

Изображение выглядит как текст, диаграмма, снимок экрана, Параллельный

Автоматически созданное описание

**1NF** – выполнено

Все атрибуты простые.

**2NF** – выполнено

1НФ и каждый не ключевой атрибут зависит от первичного ключа.

Effect:

Name зависит от id

Eyes\_effect:

Description зависит от effect\_id и от eyes\_id

Eyes:

И effect\_id, и color зависят от id

Human:

Name, sex, location, eyes\_id – все зависит от id

Pitecantrop:

Name, sex, location\_id, eyes\_id, emotion\_id – все зависит от id

Feel\_emotions:

Description зависит от emotion\_id и от pitecantrop\_id

Emotion:

Name зависит от id

Location:

Name, lighting\_id, coords – все зависит от id

Lighting:

И name, и lightlvl зависят от id

**3NF** – выполнено

2НФ и отсутствует транзитивная зависимость

**BCNF** – выполнено

3НФ и часть составного первичного ключа не зависит от неключевого столбца

Денормализация – думаю она не нужна

# **Триггер**

Функция:

CREATE FUNCTION scarefunc() RETURNS trigger AS $$

DECLARE

cave\_id integer = 1;

scare\_id integer = 1;

BEGIN

CASE WHEN NEW.location\_id = cave\_id THEN

INSERT INTO feel\_emotions (pitecantrop\_id, emotion\_id)

VALUES (NEW.id, scare\_id);

RETURN NEW;

ELSE DELETE FROM feel\_emotions WHERE pitecantrop.id = NEW.id AND emotion.id = scare\_id;

Return NEW;

End Case;

END;

$$LANGUAGE plpgsql;

Триггер:

create trigger pitec\_trigger

after insert or update of location\_id on pitecantrop

for each row execute procedure scarefunc();

# **Вывод**

В этой лабораторной работе мы изучили нормальные формы и нормализовали базу данных из первой лабораторной работы. Так же мы написали функцию и триггер на языке PL/pgSQL.