

**Федеральное государственное автономное образовательное
учреждение
высшего образования
«Национальный исследовательский университет ИТМО»
Факультет Программной Инженерии и Компьютерной Техники**



**Лабораторная работа №3
по дисциплине
Базы данных**

Выполнил Студент группы Р3112
Кобелев Роман Павлович
Преподаватель:
Лисицина Василиса Васильева.

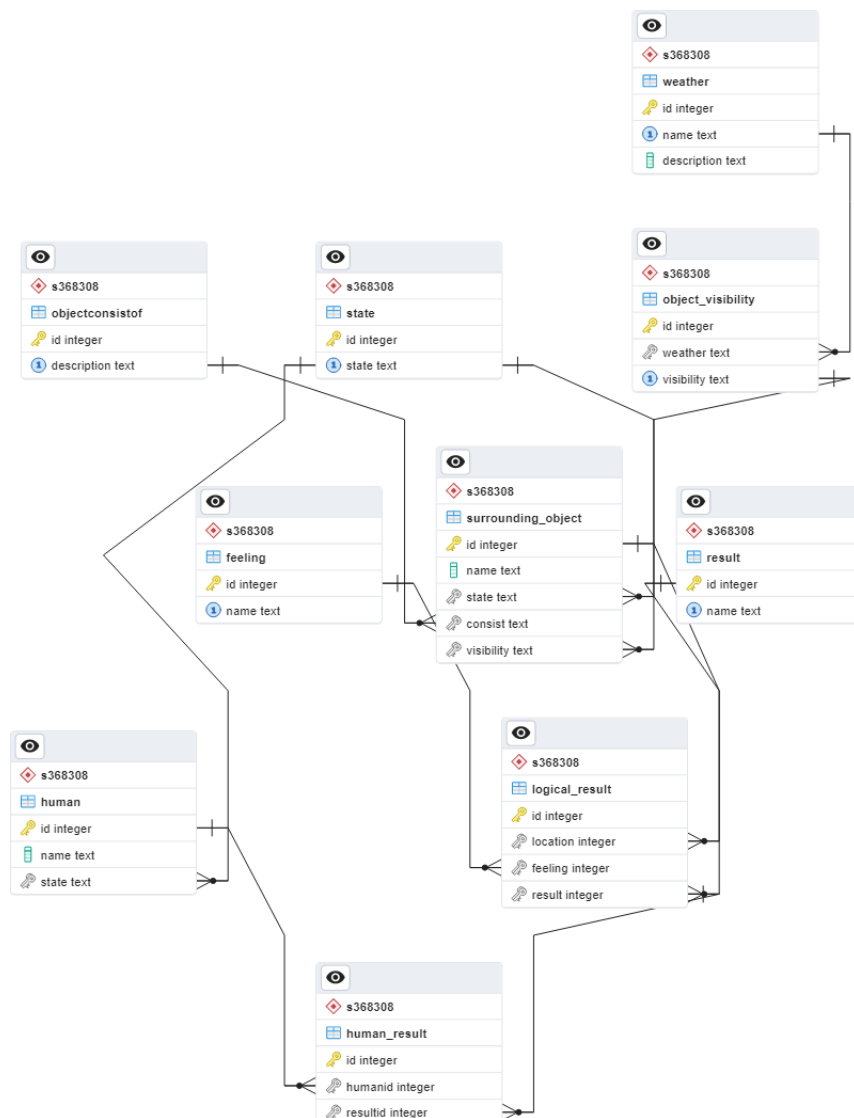
г. Санкт-Петербург
2023г.

1 Задание

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

1. Опишите функциональные зависимости для отношений полученной схемы (минимальное множество);
2. Приведите отношения в 3NF (как минимум). Постройте схему на основе 3NF (как минимум). Постройте схему на основе полученных отношений;
3. Опишите изменения в функциональных зависимостях, произошедшие после преобразования в 3NF (как минимум). Постройте схему на основе 3NF;
4. 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.

2 Исходная модель



human:

- id -> name
- id -> state (FK)

surrounding_object:

- id -> name
- id -> state (FK)
- id -> consist (FK)
- id -> visibility (FK)

feeling:

- id -> name

result:

- id -> name

state:

- id -> name

object_consist_of:

- id -> description

weather:

- id -> name

object_visibility:

- id -> weather (FK)
- id -> visibility

3 Нормализация

- 1НФ: Всё нормально, исправления не требуются. В отношениях базы данных нет групп из 2-х и более элементов.
- 2НФ: Всё нормально, исправления не требуются. В отношениях базы данных соблюдается полная функциональная зависимость. Нет частичных функциональных зависимостей. А также соблюдается 1НФ
- 3НФ: Всё нормально, исправления не требуются. В отношениях базы данных отсутствуют транзитивные функциональные зависимости. А также соблюдаются 1НФ и 2НФ.
- НФБК: Всё нормально, исправления не требуются. В отношениях бд все детерминанты являются потенциальными первичными ключами. Т.е. нет ни одного детерминанта не являющегося ключом.

4 Полезная денормализация

1. Я мог бы соединить таблицы weather и object_visibility, то есть название погоды и уровень видимости, чтобы избежать лишнего JOIN-запроса.
2. Я бы мог соединить таблицы objectconsistof и surrounding_object, то есть названия объектов и из чего они состоят, чтобы избежать лишних JOIN-запросов

5 Функция на языке PL/pgSQL

Триггер, что при добавлении нового набора feeling и result в logical_result, и этот новый result соотноситься с человеком в human_result, то надо выводить этот новый result

dop.sql

```
1  --Триггер, чтопридобавлениииновогонабораfeeling иresult вlogical_result иэтотновый
   ↳ result
2  --соотноситься счеловекомвhuman_result, тоявывожуназваниеresult
3
4  CREATE OR REPLACE FUNCTION print_result_name()
5  RETURNS TRIGGER AS $$
6  DECLARE
7      result_name VARCHAR(255);
8      assoc_exists BOOLEAN;
9  BEGIN
10     SELECT COUNT(*) INTO assoc_exists FROM human_result WHERE humanid = NEW.humanid
        ↳ AND resultid = NEW.resultid;
11
12     IF assoc_exists THEN
13         RAISE NOTICE 'Association already exists for Human ID: % and Result ID: %', NEW.
            ↳ humanid, NEW.resultid;
14     ELSE
15         SELECT name INTO result_name FROM result WHERE id = NEW.resultid;
16         RAISE NOTICE 'Result Name: %', result_name;
17     END IF;
18
19     RETURN NEW;
20 END;
21 $$ LANGUAGE plpgsql;
22
23 CREATE OR REPLACE TRIGGER print_result_name
24 BEFORE INSERT ON human_result
25 FOR EACH ROW
26 EXECUTE FUNCTION print_result_name();
```

6 Выводы

Я ознакомился с такими видами форматирования таблиц, как несколько форм нормализации и денормализация. Узнал о таком понятии, как функциональная зависимость и выучил их виды. Реализовал функцию на языке PL/pgSQL.