Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа №3

по дисциплине «Базы данных»

Вариант №3160

Выполнил:

Дядев Владислав Александрович

Группа P3131

Проверил:

Вербовой А. А.

Санкт-Петербург

2025

Содержание

[Задание 3](#_Toc195033816)

[Порядок выполнения 4](#_Toc195033817)

[Функциональные зависимости 4](#_Toc195033818)

[Нормальные формы 4](#_Toc195033819)

[Денормализация 5](#_Toc195033820)

[Триггер 6](#_Toc195033821)

[Вывод 7](#_Toc195033822)

# **Задание**

**Вариант №3160**

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

# **Порядок выполнения**

## **Функциональные зависимости**

* **sensient\_creature:** (id) -> (type, description)
* **emotion:** (id) -> (type, intensity, creature\_id)
* **event:** (id) -> (title, date, description)
* **natural\_object:** (id) -> (type, characteristic, effect)
* **thought:** (id) -> (content, complexity, creature\_id, reason\_id)
* **reason:** (id) -> (type)
* **reason\_emotion:** (id) -> (emotion\_id)
* **reason\_event:** (id) -> (event\_id)
* **reason\_object:** (id) -> (object\_id)
* **setting:** (id) -> (object\_id, event\_id, effect\_type)
* **participation:** (id) -> (creature\_id, event\_id, role)
* **reaction:** (id) -> (action\_id, emotion\_id)
* **action:** (id) -> (description, thought\_id)

## **Нормальные формы**

* **1NF:** Отношение находится в 1NF, если все его атрибуты содержат только атомарные значения и отсутствуют повторяющиеся группы. Мои отношения удовлетворяют 1NF, так как все атрибуты атомарны, и нет повторяющихся групп.
* **2NF:** Отношение находится в 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 в 2NF и не содержит транзитивных зависимостей. Моя модель не удовлетворяет 3NF, так как все неключевые атрибуты напрямую зависят от первичных ключей.
* **BCNF:** Отношение находится в BCNF, если для каждой функциональной зависимости X -> Y, X является суперключом. Моя модель удовлетворяет BCNF, так как для всех функциональных зависимостей X является суперключом.

## **Денормализация**

* **Добавление избыточных атрибутов:** в некоторых случаях можно улучшить производительность благодаря добавлению избыточных атрибутов. Например, можно добавить атрибут **emotion\_count** в **sensient\_creature**, чтобы получать количество чувств, которые существо испытывает прямо сейчас.
* **Объединение связанных таблиц:** в некоторых случаях объединение таблиц может уменьшить количество операций JOIN, то есть уменьшить время обработки запросов. В моей схеме можно рассмотреть объединение таблиц **thought** и **reason**, если часто необходимо узнать тип причины (**reason.type**), вызывающей мысль, без обращения к таблице причин.

Изображение выглядит как диаграмма, текст, План, линия

Контент, сгенерированный ИИ, может содержать ошибки.

## **Триггер**

Когда у существа появляется эмоция с высокой интенсивностью (параметр “H”), то автоматически создается событие, которое фиксирует факт сильного всплеска эмоций и связывает это событие с нужным существом. Реализация на языке SQL представлена в репозитории на github:

<https://github.com/Alvas07/ITMO/blob/main/2%20DataBases/Lab3/trigger.sql>

# **Вывод**

В ходе данной лабораторной работы я изучил понятия нормализации и денормализации. Научился работать с функциями, триггерами и писать собственные.