Функциональная зависимость - X → Y означает, что если мы имеем два кортежа в R и они совпадают по атрибутам X, то они будут совпадать и по атрибутам Y. Тогда Y атрибут функционально зависит от X.

Таблица person:  
 person.id → person.name  
Таблица action:  
 action.id → action.action\_number  
 action.id → action.person\_id

action.id → action.current\_location\_id

Таблица location:

location.id → location.name

location.id → location.state

Таблица object:

object.id → object.name

object.id → object.location

object.id → object.state

Таблица action\_description:

action\_description.id → action\_description.action\_id

action\_description.id → action\_description.change\_location\_state

action\_description.id → action\_description.change\_object\_state

action\_description.id → action\_description.thought

action\_description.id → action\_description.action\_description

Соответствуют ли отношения в схеме данных Лаб №1 3NF?

Для начала проверим, находятся ли отношения в 1NF.

Критерии 1NF:

1. Нет дубликатов строк

2. Порядок столбцов не имеет значения

3. Порядок строк не имеет значения

4. Атомарность значения (атрибута) в пересечении столбца и строки

Все это следует из определения Отношения

[DDL Скрипт и данные - скрипт лаб #1](https://github.com/KamillaT/db_lab_1_ITMO/blob/main/sql_script_lab_1.sql)

Моя схема соответствует этим критериям и поэтому находится в 1 NF.

Далее проверим, находятся ли отношения в 2NF.

Критерии 2NF:

1. 1NF

2. Все неключевые атрибуты должны быть полностью функционально зависимы от первичного ключа.

Моя схема соответствует этим критериям и поэтому находится в 2NF.

Далее проверим, находятся ли отношения в 3NF.

Критерии 3NF:

1. 1NF

2. 2NF

3. отсутствуют транзитивные функциональные зависимости неключевых атрибутов от ключевых.

Моя схема соответствует этим критериям и поэтому находится в 3NF.

Соответствуют ли отношения в схеме данных Лаб №1 BCNF?

Чтобы схема БД соответствовала BCNF:

Таблица находится в нормальной форме Бойса—Кодда (НФБК), если и только если любая функциональная зависимость между его полями сводится

к полной функциональной зависимости от возможного ключа.

Схема бд полностью соответствует этим требованиям.

Денормализация таблицы:

Если нам часто требуется получать предыдущее состояние объекта, мы можем добавить столбец "state\_history" в таблицу "object" и обновлять его каждый раз, когда состояние объекта изменяется. Тем самым мы упрощаем запрос, чтобы сохранить историю состояния объекта.

Написание триггера и связанной с ним функцией.

Функция и триггер №1.

Язык PL/pgSQL - процедурное расширения SQL, позволяющее пользователям создавать сложные функции, триггеры, условия и циклы, расширяя стандартные возможности SQL за счет процедурного программирования.

Язык PL/pgSQL позволяет использовать SQL внутри процедурных программ.

#триггер1: если на двери вмятина, то Тим не может ничего с ней сделать

Если state двери «заклинило», то action\_description «состояние покоя, Тим ничего не может сделать »

#триггер2: если окно заклинило, то Тим подумал о задней двери

Если object.state «заклинило» и это окно, то action\_description Тима «думать о задней двери»