

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

**Отчёт по лабораторной работе 3**

Предмет: БД

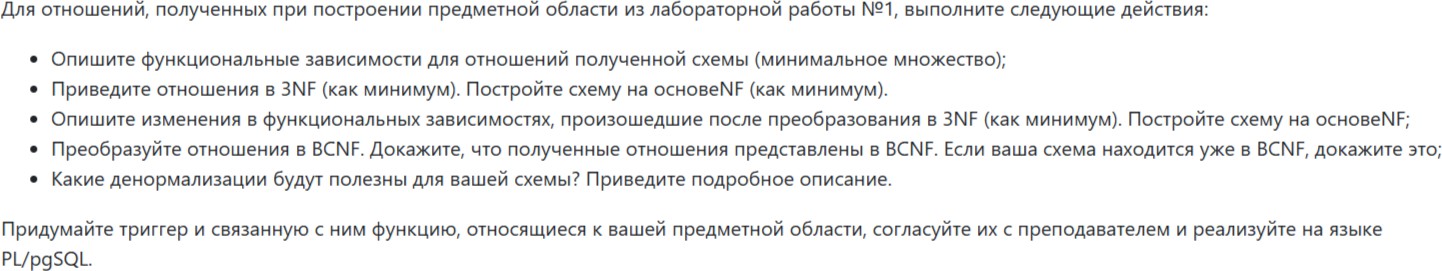
Варинат: 19191919

Выполнил**:** студент группы Р3115 Суворов Станислав Денисович

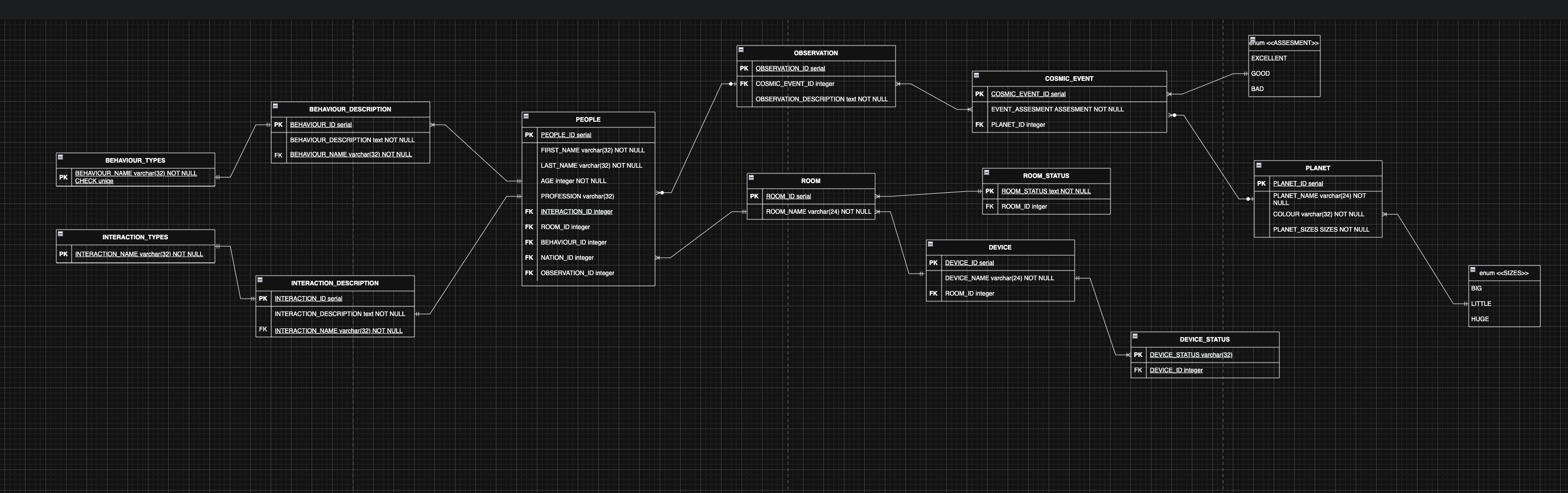
Проверил**:** Райла Мартин

2025

**Задание**

****

**Даталогическая модель**

****

**Описание функциональных зависимостей**

* **PEOPLE\_ID → FIRST\_NAME**
* **PEOPLE\_ID → LAST\_NAME**
* **PEOPLE\_ID → AGE**
* **PEOPLE\_ID → PROFESSION**
* ​
* **PEOPLE\_ID → INTERACTION\_ID**
* **PEOPLE\_ID → ROOM\_ID**
* **PEOPLE\_ID → BEHAVIOUR\_ID**
* **PEOPLE\_ID → NATION\_ID**
* **PEOPLE\_ID → OBSERVATION\_ID**

**People**

1НФ: Атрибуты атомарны, не содержат составных значений, также содержит первичный ключ, сущность находится в 1НФ.

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK, и таблица уже находится в 1НФ => таблица находится в 2НФ.

3НФ: Отсутсивуют транзитивные зависимости => таблица находится в 3НФ.

НФБК: Соблюдается тк, в нашей таблице суперключом является PEOPLE\_ID – от него идут все функциональные зависимости, таблица находится в НФБК.

**Behaviour\_Types**

PK Behaviour\_name

1НФ: Атрибут атомарен, не содержит составных значений, первичный ключ есть. Таблица находится в 1НФ

2НФ: нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK.

Находится в 2НФ.

3НФ: Тк не содержит неключевых атрибутов значит нет транзитивных зависимостей. Находится в 3НФ.

НФБК: Такая же причина

**INTERACTION\_TYPES**

PK INTERACTION\_NAME

1НФ: Атрибут атомарен, не содержит составных значений, первичный ключ есть. Таблица находится в 1НФ

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной зависимости от PK. Находится в 2НФ

3НФ: Тк не содержит неключевых атрибутов значит нет транзитивных зависимостей. Находится в 3НФ.

НФБК: Такая же причина

Behaviours\_Description

Функциональные зависимости

* BEHAVIOUR\_ID → BEHAVIOUR\_DESCRIPTION
* BEHAVIOUR\_ID → BEHAVIOUR\_NAME

1НФ: Атрибуты атомарны, не содержат составных значений, есть первичный ключ. Находится в 1НФ.

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK Находится в 2НФ.

3НФ: Нет связи между атрибутами, каждый зависит только от ID. Следовательно транзитивных зависимостей – нет. Таблица в 3НФ

НФБК: Соблюдается тк. нет зависимостей где ID не ключ.

**INTERACTION\_DESCRIPTION**

* INTERACTION\_ID → INTERACTION \_DESCRIPTION
* INTERACTION\_ID → INTERACTION \_NAME

1НФ: Атрибуты атомарны, не содержат составных значений, есть первичный ключ. Находится в 1НФ.

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK Находится в 2НФ.

3НФ: Нет связи между атрибутами, каждый зависит только от ID. Следовательно транзитивных зависимостей – нет. Таблица в 3НФ

НФБК: Соблюдается тк. нет зависимостей где ID не ключ.

OBSERVATION

* OBSERVATION\_ID → COSMIC\_EVENT\_ID
* OBSERVATION\_ID → OBSERVATION\_DESCRIPTION
* **1НФ:** Атрибуты атомарны. Не содержат составных значений, есть PK. Находится в 1НФ**.**
* **2НФ:** PK простой. Неключевые атрибуты полностью зависят от него. **.** Находится в 2НФ.
* **3НФ:** Неключевые атрибуты зависят только от PK. Нет зависимостей между неключевыми атрибутами. Следовательно, транзитивных зависимостей нет. Таблица в 3НФ.
* **НФБК:** Соблюдается тк. нет зависимостей где ID не ключ.

Cosmic\_Event

* COSMIC\_EVENT\_ID → COSMIC\_EVENT\_ASSESSMENT
* COSMIC\_EVENT\_ID → PLANET\_ID
* COSMIC\_EVENT\_ID → PLANET\_ID → COSMIC\_EVENT\_ASSESSMENT

1НФ: Атрибуты атомарны. Не содержат составных значений, есть PK. Находится в 1НФ 2НФ: PK простой. Неключевые атрибуты полностью зависят от него. **.** Находится в 2НФ. 3НФ: Есть транзитивная зависимость.

НФБК: не в 3НФ соответственно не НФБК

ROOM

1. ROOM\_ID → ROOM\_NAME

1НФ: Атрибуты атомарны. Не содержат составных значений, есть PK. Находится в 1НФ 2НФ: PK простой. Неключевые атрибуты полностью зависят от него. **.** Находится в 2НФ. 3НФ: Отсутсвуют транзитивные зависимости.

НФБК: Соблюдается тк. нет зависимостей где ID не ключ.

ROOM\_STATUS

1. ROOM\_STATUS → ROOM\_ID

1НФ: Атрибуты атомарны. Не содержат составных значений, есть PK. Находится в 1НФ 2НФ: PK простой. Неключевые атрибуты полностью зависят от него. **.** Находится в 2НФ. 3НФ: Отсутсвуют транзитивные зависимости.

НФБК: Соблюдается тк. нет зависимостей где ID не ключ.

PLANET

* PLANET\_ID → PLANET\_NAME
* PLANET\_ID → COLOUR
* PLANET\_ID → PLANET\_SIZES

1НФ: Все атрибуты являются атомарными. Они содержат одиночные, неделимые значения. enum также является атомарным

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK.

3НФ: Отсутствуют транзитивные зависимости.

НФБК: Соблюдается тк. нет зависимостей где ID не ключ.

Device

* DEVICE\_ID → DEVICE\_NAME
* DEVICE\_ID → ROOM\_ID
* DEVICE\_ID → ROOM\_ID → DEVICE\_NAME

1НФ: Все атрибуты являются атомарными. Они содержат одиночные, неделимые значения. enum также является атомарным

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK.

3НФ: Есть транзитивная зависимость. НФБК: не в 3НФ соответственно не НФБК

Device\_Status

* + DEVICE\_STATUS → DEVICE\_ID

1НФ: Все атрибуты являются атомарными. Они содержат одиночные, неделимые значения. enum также является атомарным

2НФ: Нет составного PK, поэтому неключевые атрибуты находятся в полной завсисимости от PK.

3НФ: Отсутствуют транзитивные зависимости.

НФБК: Соблюдается тк. нет зависимостей где ID не ключ.

**Денормализация** В целом мы можем ускорить запросы к таблице DEVICE Путем включения DEVICE\_NAME в DEVICE

Не нужен будет join

**Триггер**

**Сложный триггер созданный по мотивам задания из ЕГЭ по информатике про игру в камушки: Триггер проводит учет победителя**

**CREATE TABLE GameSessions (**

**game\_session\_id SERIAL PRIMARY KEY,**

**game\_winning INTEGER,**

**current\_game\_status VARCHAR(32),**

**number\_of\_stone INTEGER**

**);**

**CREATE TABLE Moves (**

**move\_id SERIAL PRIMARY KEY,**

**game\_session\_id INTEGER REFERENCES GameSessions(game\_session\_id),**

**player\_name VARCHAR(32),**

**add\_of\_stones INTEGER,**

**stones\_after\_move INTEGER**

**);**

**CREATE OR REPLACE FUNCTION check\_game\_winner()**

**RETURNS TRIGGER AS $$**

**DECLARE**

**game\_winning\_threshold INTEGER;**

**current\_status VARCHAR(32);**

**curr\_number\_of\_stone INTEGER;**

**total\_stones INTEGER;**

**BEGIN**

**SELECT game\_winning, current\_game\_status, number\_of\_stone**

**INTO game\_winning\_threshold, current\_status, curr\_number\_of\_stone**

**FROM GameSessions**

**WHERE game\_session\_id = NEW.game\_session\_id;**

**SELECT COALESCE(SUM(add\_of\_stones), 0)**

**INTO total\_stones**

**FROM Moves**

**WHERE game\_session\_id = NEW.game\_session\_id;**

**total\_stones := total\_stones + NEW.add\_of\_stones;**

**NEW.stones\_after\_move := total\_stones;**

**IF current\_status = 'We play' AND total\_stones >= game\_winning\_threshold THEN**

**UPDATE GameSessions**

**SET current\_game\_status =**

**CASE NEW.player\_name**

**WHEN 'Petya' THEN 'Petya\_win'**

**WHEN 'Vanya' THEN 'Vanya\_win'**

**ELSE 'Error'**

**END**

**WHERE game\_session\_id = NEW.game\_session\_id;**

**END IF;**

**RETURN NEW;**

**END;**

**$$ LANGUAGE plpgsql;**

**CREATE TRIGGER trg\_check\_game\_winner**

**BEFORE INSERT ON Moves**

**FOR EACH ROW**

**EXECUTE FUNCTION check\_game\_winner();**

**Вывод**

Я изучил нормализацию бд, проанализировал отношения моей модели, определил их НФ. Написал триггер.