МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «Санкт-Петербургский национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА №3**

по дисциплине «Информационные системы и базы данных»

Вариант № 1677

***Выполнил:***

Студент группы P3116

Билошицкий Михаил Владимирович

***Преподаватель:***

Горбунов Михаил Витальевич

Санкт-Петербург, 2023

# Содержание

[Содержание 2](#_Toc136102770)

[Текст задания 3](#_Toc136102771)

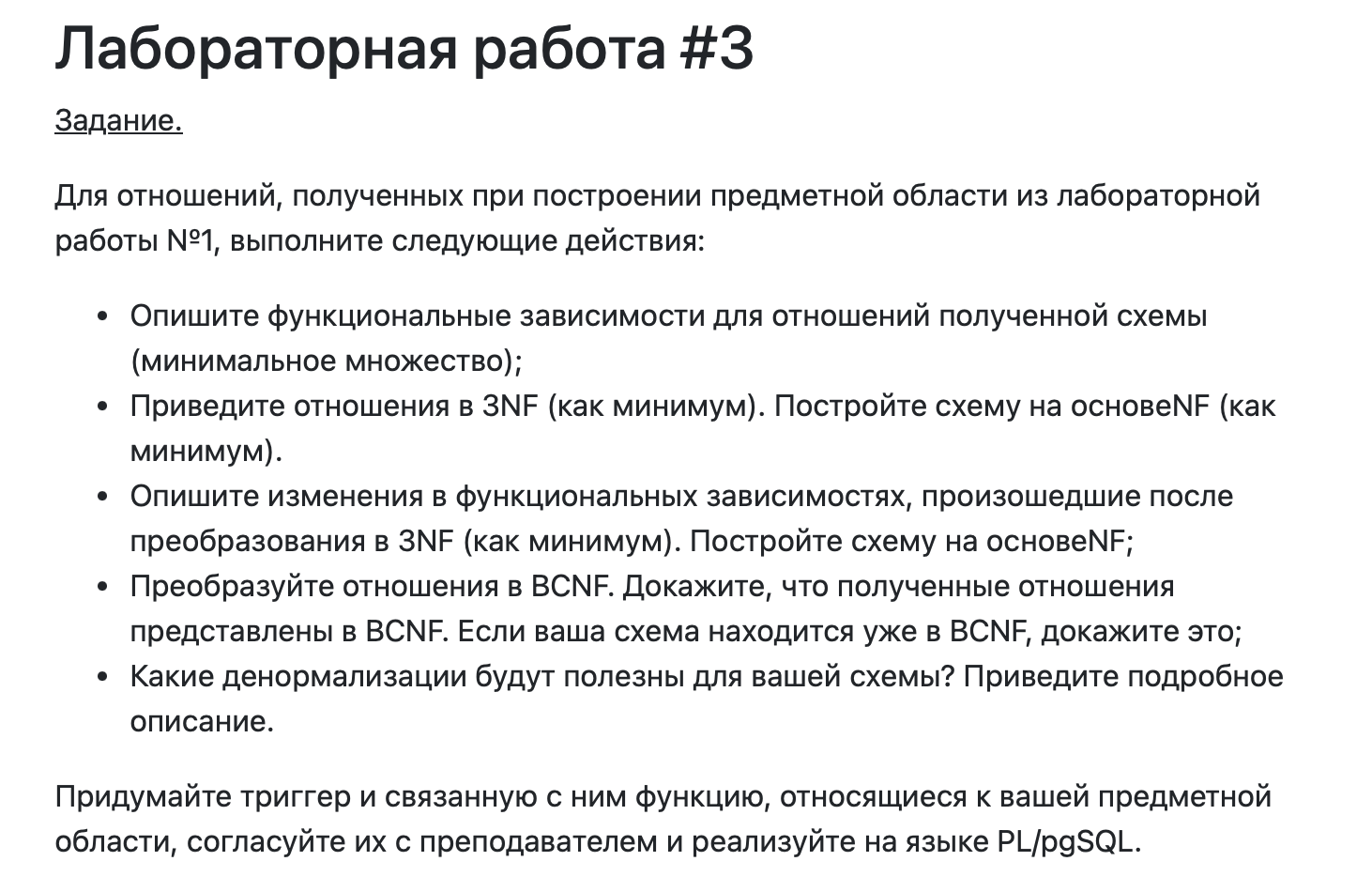
[Исходная, нормализованная и денормализованная модели 4](#_Toc136102772)

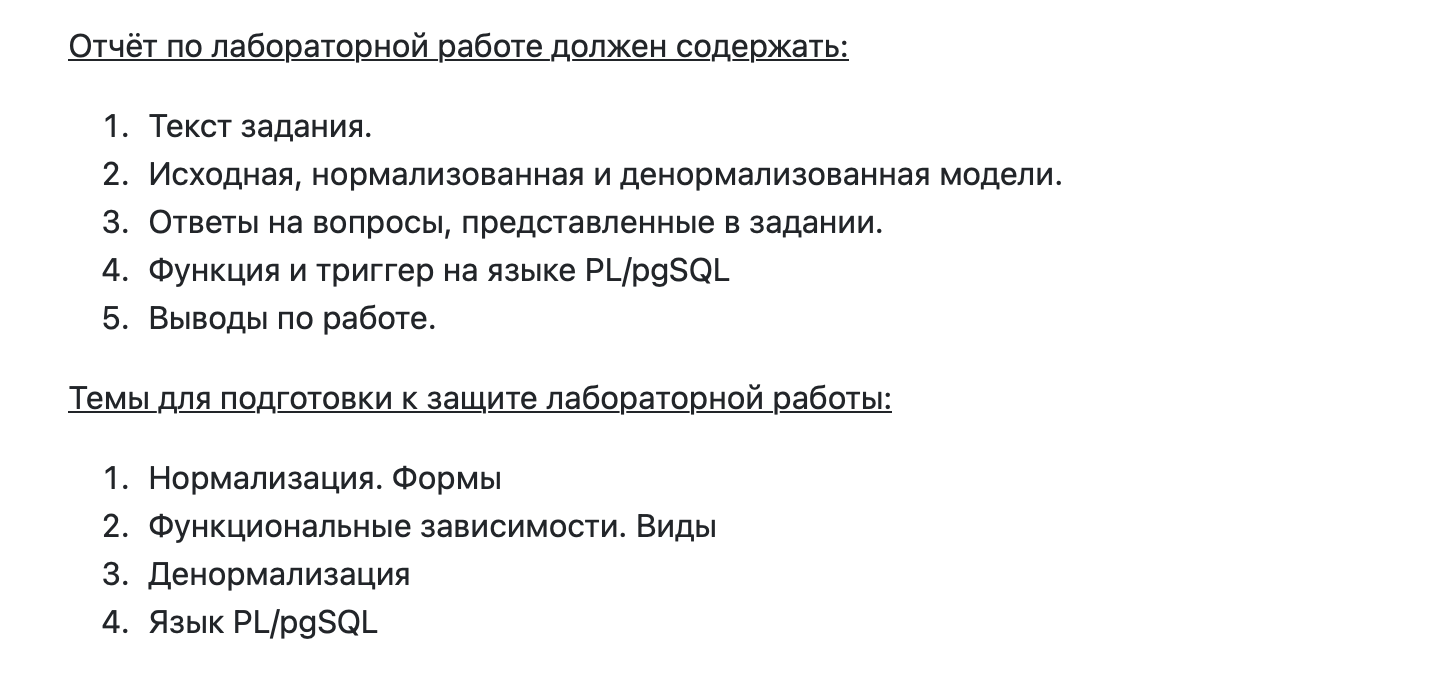
[Ответы на вопросы 5](#_Toc136102773)

[Функция и триггер 9](#_Toc136102774)

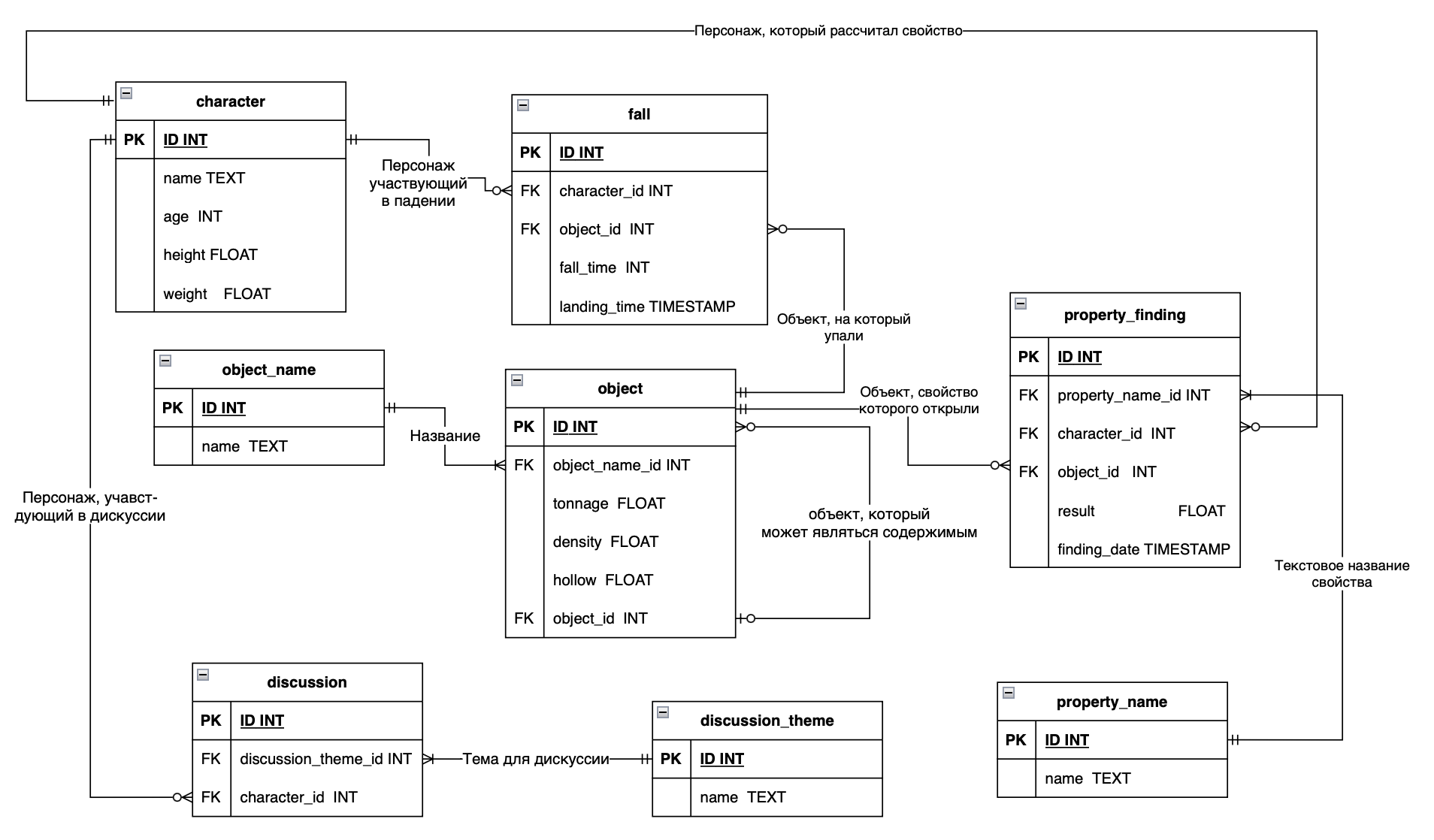
[Вывод 10](#_Toc136102775)

# Текст задания





# Исходная, нормализованная и денормализованная модели



Исходная модель

Функциональные зависимости:

character:

id -> name

id -> age

id -> height

id -> weight

fall (транзитивная зависимость):

id -> object\_id

id -> character\_id

id -> fall\_time

object\_id, character\_id -> landing\_time

object:

id -> object\_name\_id

id -> tonnage

id -> density

id -> hollow

id -> object\_id

property\_finding (транзитивная зависимость):

id -> property\_name\_id

id -> character\_id

id -> object\_id

id -> finding\_date

property\_name\_id, character\_id, object\_id, finding\_date -> result

discussion:

id -> discussion\_theme\_id

id -> character\_id

discussion\_theme:

id -> name

property\_name:

id -> name

object\_name:

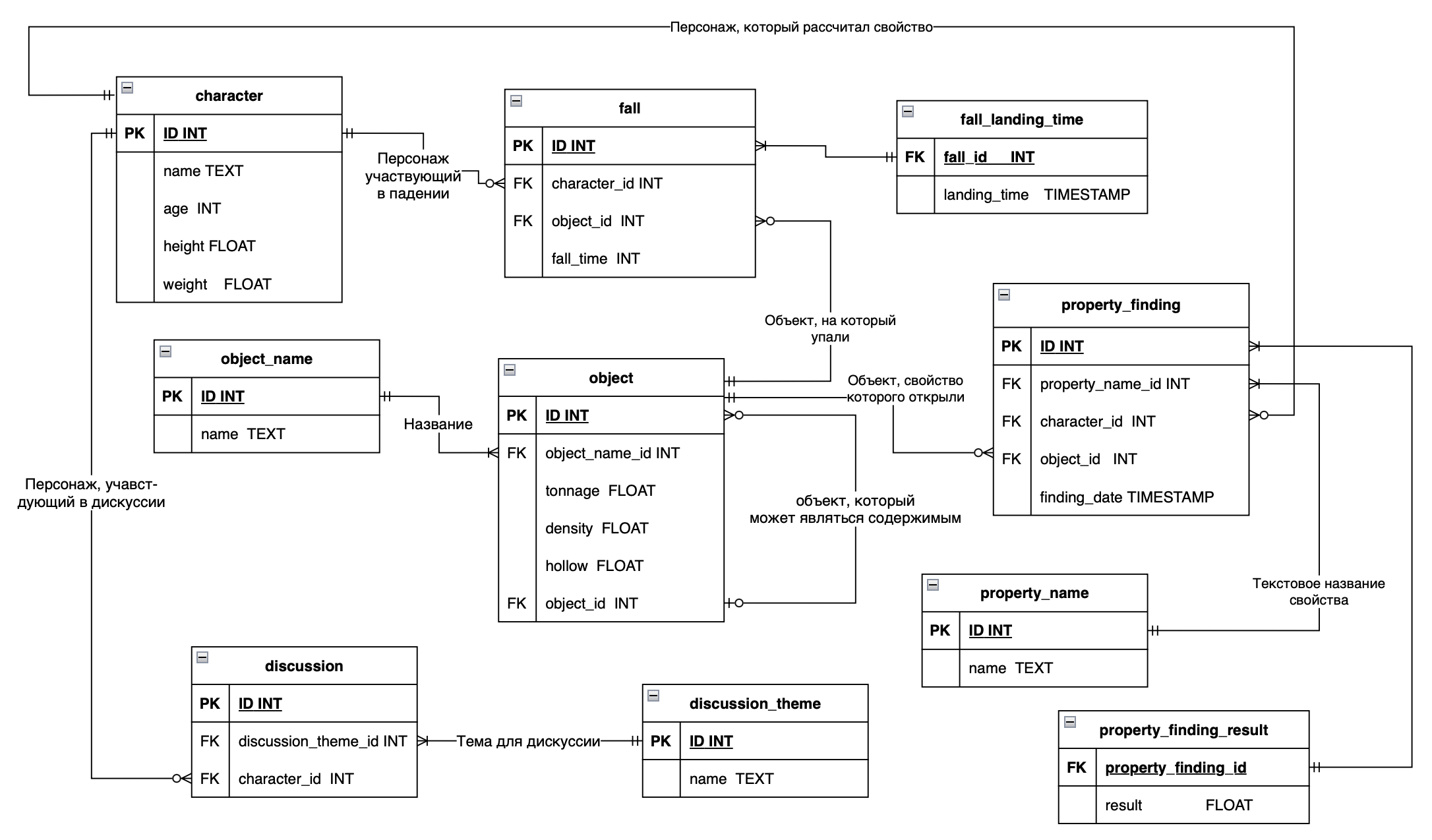
id -> name

# Ответы на вопросы

* Исходная схема соответствует 1НФ, поскольку на пересечении строк и столбцов не встречается нескольких значений.
* Исходная схема соответствует 2НФ, поскольку она соответствует 1НФ и атрибуты, не входящие в PK, находятся в полной функциональной зависимости от PK и нет частичных зависимостей от потенциальных ключей.
* Исходная схема не соответствует 3НФ, поскольку она соответствует 1НФ и 2 НФ, но не все атрибуты, не входящие в PK, не находятся в транзитивной зависимости от PK. Исправим схему и приведем ее к 3НФ:

1. Разделим таблицу fall на таблицы fall и fall\_landing\_time, чтобы избежать транзитивной зависимости.
2. Разделим таблицу property\_finding на таблицы property\_finding и property\_finding\_result, чтобы избежать транзитивной зависимости.

В результате нормализации получим следующую схему с изменениями:



В результате нормализации получим следующие функциональные зависимости:

character:

id -> name

id -> age

id -> height

id -> weight

fall:

id -> object\_id

id -> character\_id

id -> fall\_time

fall\_landing\_time:

fall\_id -> landing\_time

object:

id -> object\_name\_id

id -> tonnage

id -> density

id -> hollow

id -> object\_id

property\_finding:

id -> property\_name\_id

id -> character\_id

id -> object\_id

id -> finding\_date

property\_finding\_result:

property\_finding\_id -> result

discussion:

id -> discussion\_theme\_id

id -> character\_id

discussion\_theme:

id -> name

property\_name:

id -> name

object\_name:

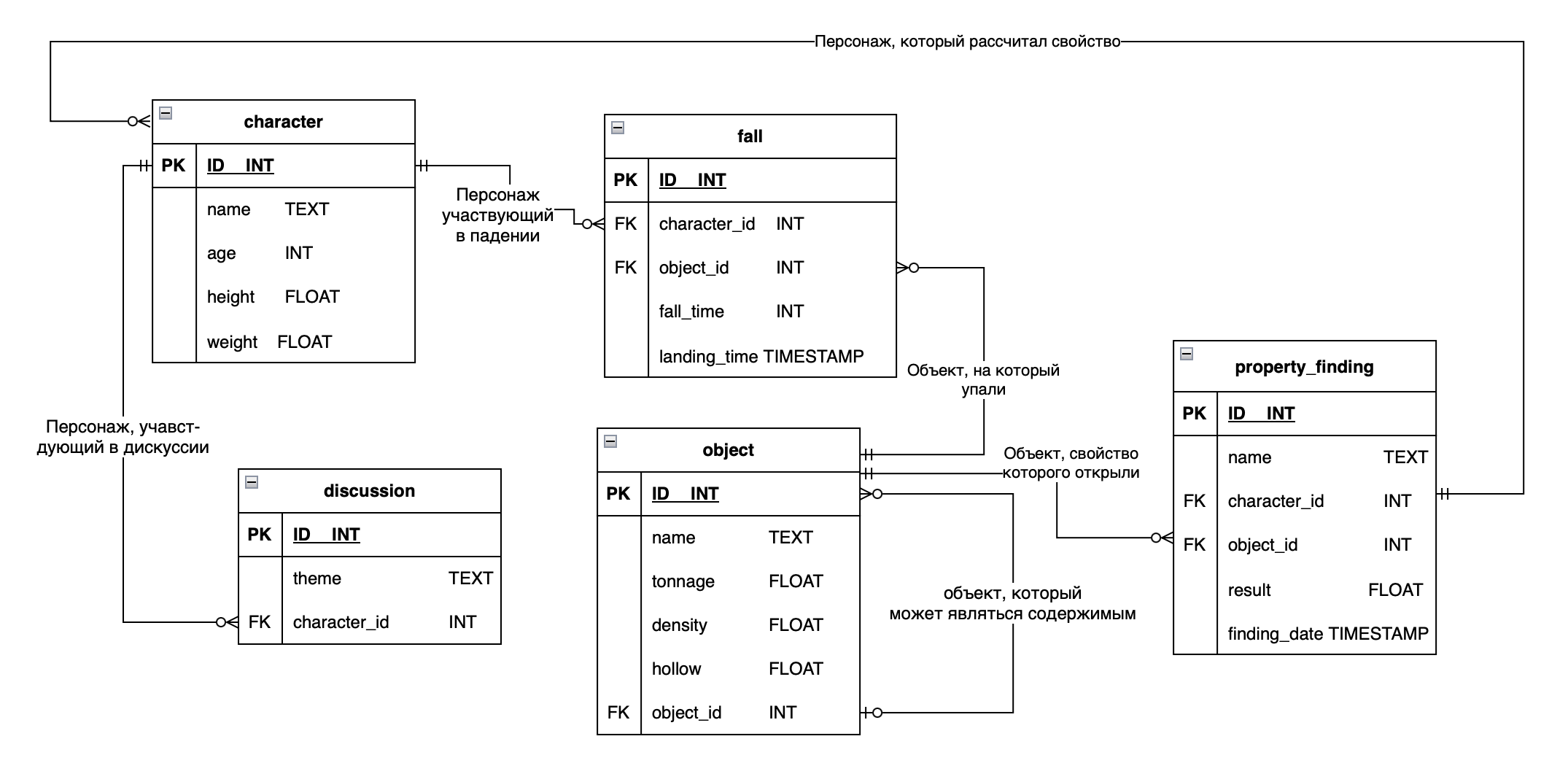
id -> name

* Нормализованная схема находится в BCNF, потому что выполняется условие BCNF – наличие 3НФ + во всех функциональных отношениях детерминанты вынесены как первичные ключи в отдельные таблицы. Столбцов, являющихся детерминантами за исключением первичного ключа, в таблицах не осталось.

Полезные денормализации:

1. Таблицы discussion и discussion\_theme можно объединить во избежание постоянного объединения всех данных таблиц. Но минусом будет являться частая повторяемость данных.
2. Таблицы object и object\_name можно объединить во избежание постоянного объединения всех данных таблиц. Но минусом будет являться частая повторяемость данных.
3. Таблицы property\_finding и property\_name можно объединить во избежание постоянного объединения всех данных таблиц. Но минусом будет являться частая повторяемость данных.
4. Таблицы fall и fall\_landing\_time можно объединить во избежание постоянного объединения всех данных таблиц. Но минусом будет являться наличие транзитивной функциональной зависимости.
5. Таблицы property\_finding и property\_finding\_result можно объединить во избежание постоянного объединения всех данных таблиц. Но минусом будет являться наличие транзитивной функциональной зависимости.

В результате денормализации получается следующая схема:



В результате денормализации получатся следующие отношения:

character:

id -> name

id -> age

id -> height

id -> weight

fall (транзитивная зависимость):

id -> object\_id

id -> character\_id

id -> fall\_time

object\_id, character\_id -> landing\_time

object:

id -> name

id -> tonnage

id -> density

id -> hollow

id -> object\_id

property\_finding (транзитивная зависимость):

id -> property\_name\_id

id -> character\_id

id -> object\_id

id -> finding\_date

property\_name\_id, character\_id, object\_id, finding\_date -> result

discussion:

id -> discussion\_theme

id -> character\_id

# Функция и триггер

CREATE OR REPLACE FUNCTION **property\_finding\_date\_initialize**() RETURNS TRIGGER

LANGUAGE PLPGSQL

AS $$

BEGIN

if (new.finding\_date IS NULL) THEN

new.finding\_date := current\_timestamp AT TIME ZONE 'Europe/Moscow';

end if;

RETURN new;

END $$;

CREATE OR REPLACE TRIGGER **property\_finding\_date\_initialize\_trigger**

AFTER UPDATE OR INSERT ON property\_finding FOR EACH ROW EXECUTE PROCEDURE property\_finding\_date\_initialize();

Мы создаем функцию, которая будет выполняться триггером на добавление или обновление нового свойства, где если дата нахождения свойства будет пустой, то триггер создаст ее.

# Вывод

В рамках данной лабораторной работы были изучены нормальные формы. Проведен анализ модели на соответствие BCNF. Проведена денормализация с целью оптимизации выполнения запросов к таблице. Написана функция на языке PL/pgSQL.