­­­НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет “Нейротехнологии и программирование”

Дисциплина “Базы данных”

Лабораторная работа № 3

Выполнил студент

Доровский Егор Сергеевич

Группа № Р3123

Преподаватель: Горбунов Михаил Витальевич

г. Санкт-Петербург

2024

Текст задания

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.

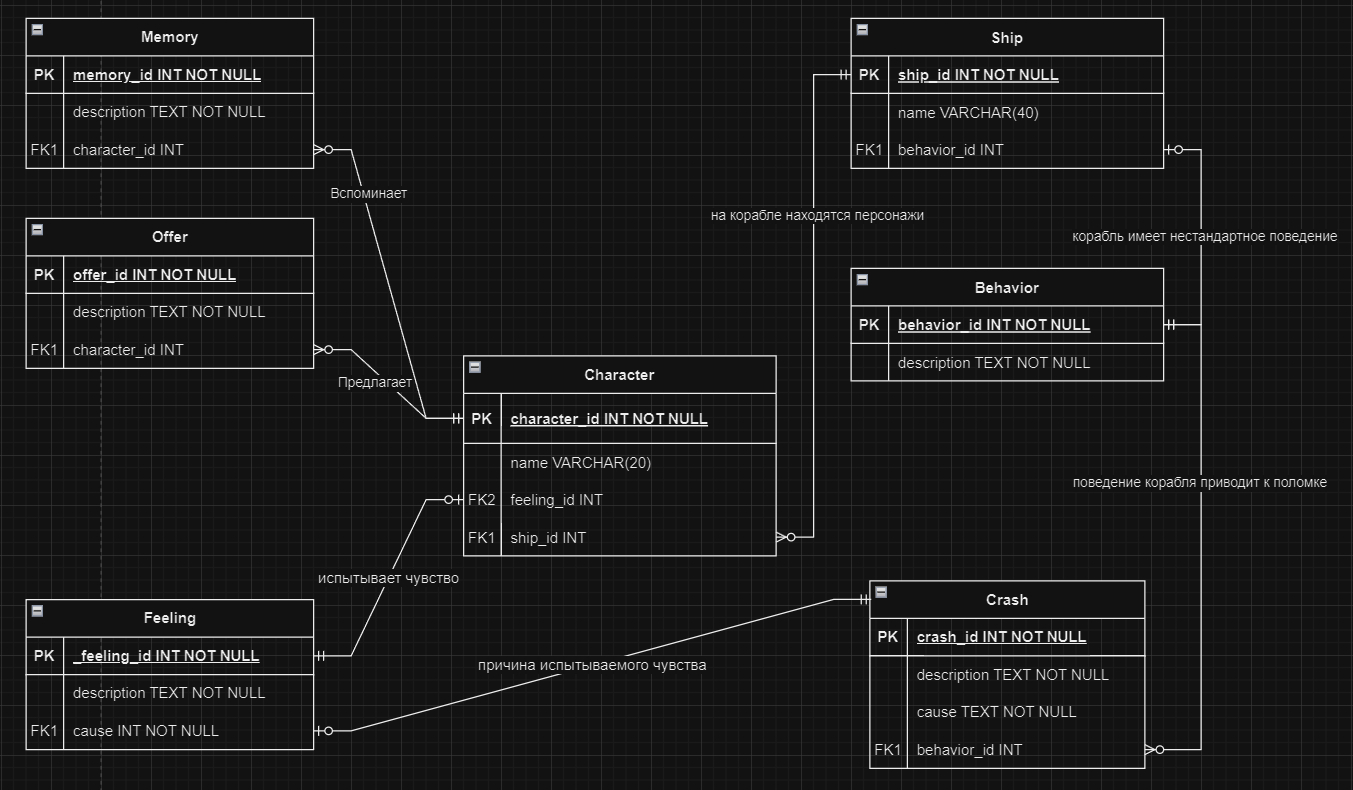
Отчёт по лабораторной работе должен содержать:

1. Текст задания.
2. Исходная, нормализованная и денормализованная модели.
3. Ответы на вопросы, представленные в задании.
4. Функция и триггер на языке PL/pgSQL
5. Выводы по работе.

Темы для подготовки к защите лабораторной работы:

1. Нормализация. Формы
2. Функциональные зависимости. Виды
3. Денормализация
4. Язык PL/pgSQL

Исходная модель



Функциональные зависимости

|  |  |
| --- | --- |
| Character | id -> (name, feeling\_id, ship\_id) |
| Memory | memory\_id -> (description\_character\_id) |
| Offer | offer\_id -> (description, character\_id) |
| Ship | ship\_id -> (name, behavior\_id) |
| Behavior | behavior\_id -> (description) |
| Feeling | feeling\_id -> (description, cause) |
| Crash | сrash\_id -> (description, cause, behavior\_id) |

Нормальные формы

1НФ

Отношение находится в первой нормальной форме (1НФ), если:

* Все его атрибуты являются простыми (т.е. неделимыми на более мелкие компоненты).
* Все используемые домены (области допустимых значений) содержат только скалярные (не имеющие направления) значения.
* В таблице (отношении) не должно быть повторяющихся строк.

Таким образом, моя модель удовлетворяет 1НФ, потому что в ней нет неатомарных атрибутов и повторяющихся строк.

2НФ

Отношение находится во второй нормальной форме, если оно удовлетворяет следующим условиям:

* Отношение находится в 1НФ.
* Каждый неключевой атрибут отношения неприводимо зависит от всего потенциального ключа, а не только от части ключа.

Это утверждение справедливо для моей модели

3НФ

Переменная отношения R находится в 3NF, если:

* Отношение находится в 2НФ.
* Ни один неключевой атрибут R не находится в транзитивной [функциональной зависимости](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) от [потенциального ключа](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%82%D0%B5%D0%BD%D1%86%D0%B8%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BB%D1%8E%D1%87) R.

В моей модели нет транзитивных зависимостей.

НФБК

Отношение находится в НФБК, когда:

* Каждая нетривиальная и неприводимая слева функциональная зависимость обладает потенциальным ключом в качестве детерминанта.

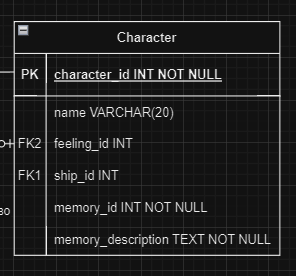
Моя модель удовлетворяет этой форме, потому что во всех отношениях детерминанты – потенциальные ключи

Денормализация

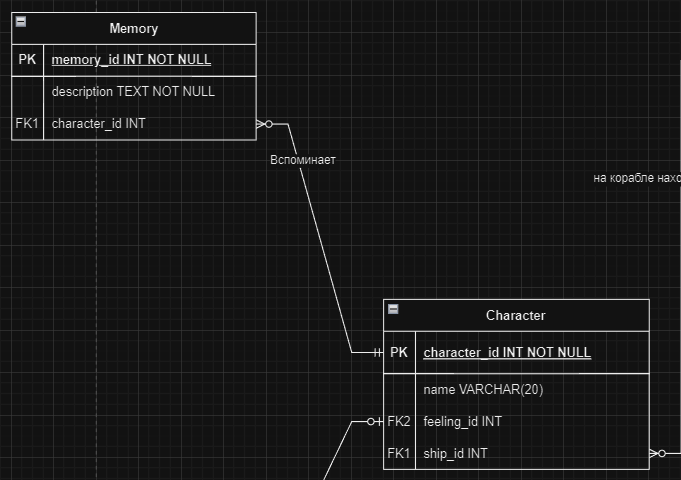
Добавить атрибут name из таблицы Character в таблицу Memory, чтобы иметь доступ к названию имени персонажа без соединения таблиц

Пример транзитивной зависимости

Здесь есть транзитивная зависимость character\_id -> memory\_id



Чтобы избавиться от этой зависимости, вынесем memory\_id и memory\_description в новую таблицу:



Функция

CREATE OR REPLACE FUNCTION update\_character\_memories()

RETURNS TRIGGER AS $$

BEGIN

IF TG\_OP = ‘DELETE’ THEN

UPDATE Character SET memories\_count = (SELECT COUNT(\*) FROM Memory WHERE

Memory.character\_id = OLD.character\_id) WHERE Character.character\_id =

OLD.character\_id;

ELSIF IF TG\_OP = 'INSERT' OR TG\_OP = 'UPDATE' THEN

UPDATE Character SET memories\_count = (SELECT COUNT(\*) FROM Memory WHERE

Memory.character\_id = NEW.character\_id) WHERE Character.character\_id =

NEW.character\_id;

END IF;

RETURN NULL;

END;

$$ LANGUAGE plpgsql;

Триггер

CREATE TRIGGER character\_update\_character\_memories

AFTER INSERT OR UPDATE OR DELETE ON Memory

FOR EACH ROW

EXECUTE FUNCTION update\_character\_memories();

Список литературы:

[Инфологическая модель БД "Учебный Процесс"](https://se.ifmo.ru/documents/10180/733702/%D0%91%D0%94+%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D1%8B%D0%B9+%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81.pdf/2eae3fcd-ea34-4496-924b-6ee4e889a9e5)

[Документация к СУБД PostgreSQL](https://www.postgresql.org/docs/9.6/static/)

[Лекция 1](https://se.ifmo.ru/documents/10180/733702/isbd-2021-1.5.pdf/72944648-dbe7-ced4-38ac-35efe9f87e6f), [Лекция 2](https://se.ifmo.ru/documents/10180/733702/isbd-2021-2.6.pdf/e47a2e01-d445-e017-070f-a300ecdb71a8), [Лекция 3](https://se.ifmo.ru/documents/10180/733702/isbd-2021-3.3.pdf/4d439555-a3cb-883c-705a-bab0fa37687b), [Лекция 4](https://se.ifmo.ru/documents/10180/733702/isbd-2021-4.3.pdf/67de83a2-571f-527c-1ec7-7c9853c86576), [Лекция 5](https://se.ifmo.ru/documents/10180/733702/isbd-2021-5.2.pdf/beadf2a7-db1d-d0c9-c0aa-6abf923c4473), [Лекция 6](https://se.ifmo.ru/documents/10180/733702/isbd-2021-6.2.pdf/e31693c5-499d-3d2d-067c-3e6462d20d57), [Лекция 7](https://se.ifmo.ru/documents/10180/733702/isbd-2021-7.2.pdf/897de389-c846-3483-cd8e-98348c71547b)