

*Факультет программной инженерии и компьютерной техники*

*Дисциплина «Базы данных»*

**Отчет**

**По лабораторной работе №3**

**Вариант №876596**

Выполнил:

Саранча Павел Александрович

Группа: Р3109

Преподаватель:

Райла Мартин

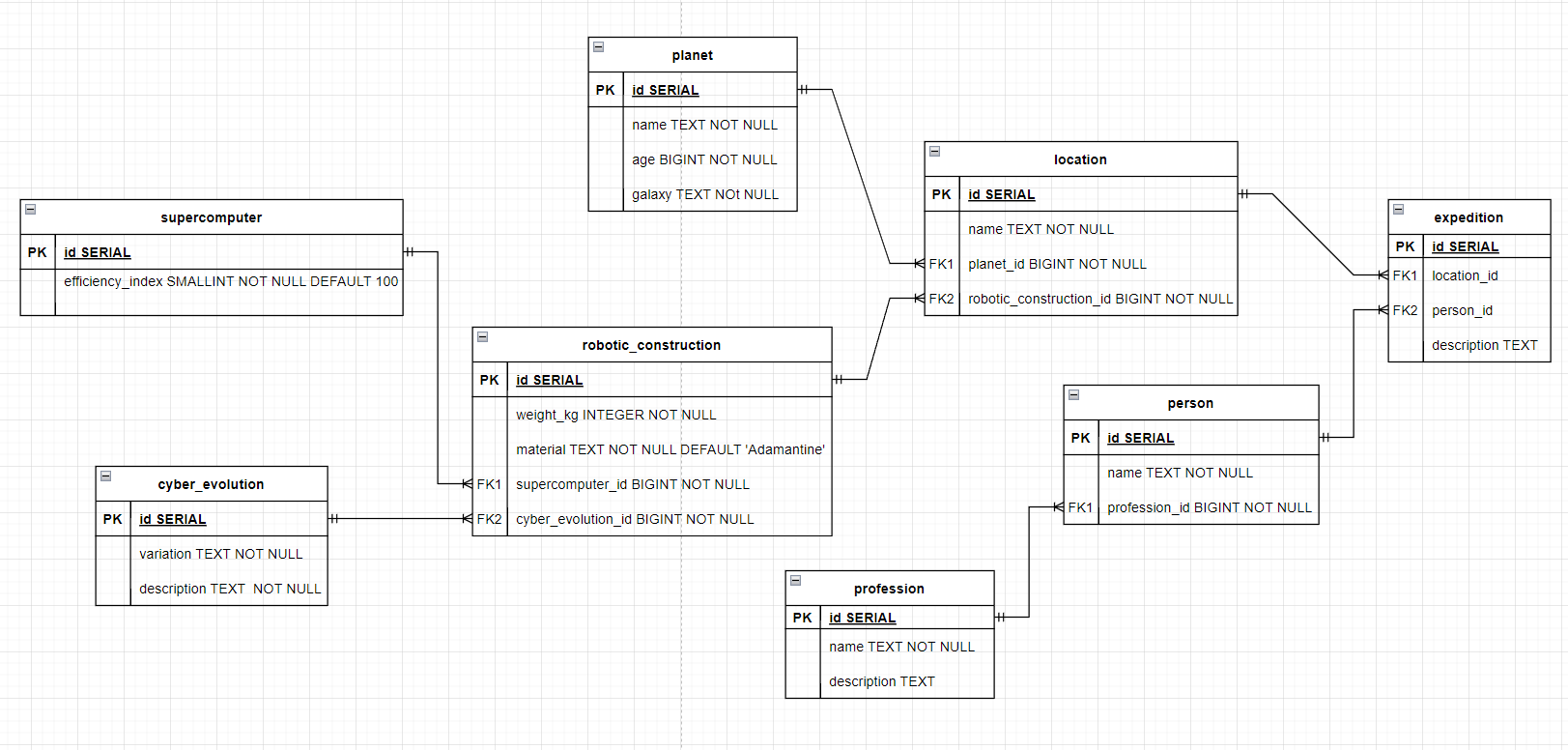
Г. Санкт-Петербург, 2024 г.

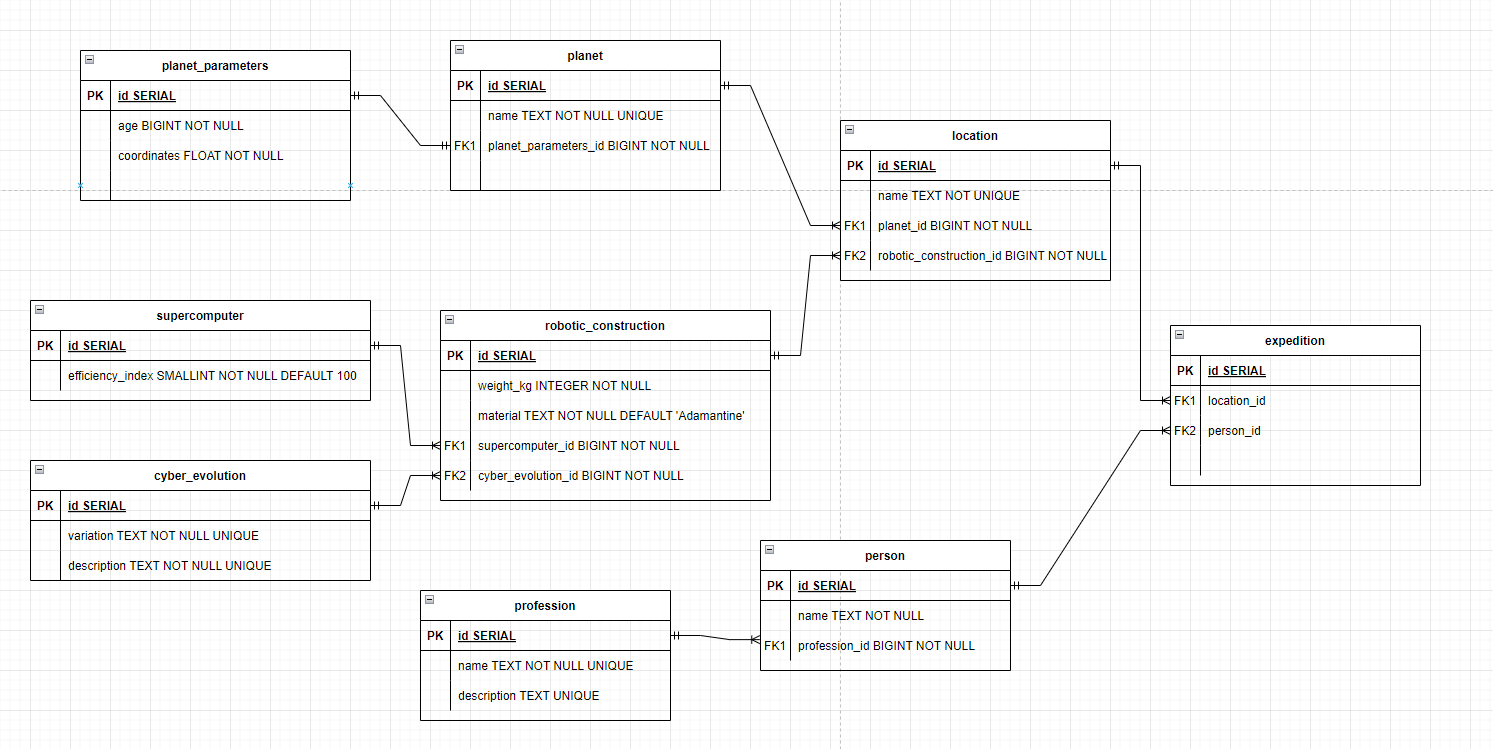
# 1. Текст задания:

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





# 2. Функциональные зависимости

|  |
| --- |
| person: id → (name, profession\_id)    profession: id → (name, description)    cyber\_evolution: id → (variation, description)    supercomputer: id → (efficiency\_index)    robotic\_construction: id → (weight\_kg, material, supercomputer\_id, cyber\_evolution\_id)    location: id → (name, planet\_id, robotic\_construction\_id)    planet: id → (name, planet\_parameters\_id)  planet\_parameters: id → (age, coordinates)    expedition: id → (location\_id, person\_id) |

# 3. Нормальные формы

|  |
| --- |
| 1NF: Отношение находится в 1NF, если все его атрибуты содержат только атомарные значения. Моя модель удовлетворяет 1NF, так как все атрибуты атомарны, и нет повторяющихся групп.  Таблицы:  1) person(name, profession\_id) в 1НФ так как нет повторов строк, имена могут повторяться, в строках только одно значение  2) profession(name, description) в 1НФ так как нет повторов строк, name и description с модификатором UNIQUE, в строках только одно значение.  3) cyber\_evolution(variation, description) в 1НФ так как нет повторов строк, variation и description с модификатором UNIQUE, в строках только одно значение.  4) supercomputer(efficiency\_index) в 1НФ так на каждый id есть единственный индекс эффективности.  5) robotic\_construction(weight\_kg, material, supercomputer\_id, cyber\_evolution\_id) в 1НФ, так как нет повторов строк, в каждой строке только одно значение для каждого атрибута.  6) location(name, planet\_id, robotic\_construction\_id) в 1НФ, так как нет повторов строк, в каждой строке только одно значение для каждого атрибута, name с модификатором UNIQUE.  7) planet(name, planet\_parameters\_id) в 1НФ, так как нет повторов строк, в каждой строке только одно значение для каждого атрибута, name с модификатором UNIQUE.  8) planet\_parameters(age, coordinates) в 1НФ, так как нет повторов строк, в каждой строке только одно значение для каждого атрибута.  9) expedition(location\_id, person\_id) в 1НФ, так как нет повторов строк, в каждой строке только одно значение для каждого атрибута. |
| 2NF: Отношение находится во 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.  Таблицы:   1. person(name, profession\_id) в 2НФ так как есть первичный ключ, он не составной и неключевые атрибуты name и profession\_id зависят только от него. 2. profession(name, description) в 2НФ так как есть первичный ключ и он не составной и неключевые атрибуты name и description зависят только от него. 3. cyber\_evolution(variation, description) в 2НФ так как есть первичный ключ и он не составной и неключевые атрибуты variation и description зависят только от него.   4) supercomputer(efficiency\_index) в 2НФ так как есть первичный ключ и он не составной и неключевой атрибут efficiency\_index зависит только от него  5) robotic\_construction(weight\_kg, material, supercomputer\_id, cyber\_evolution\_id) в 2НФ так как есть первичный ключ и он не составной и неключевые атрибуты weight\_kg , material, supercomputer\_id и cyber\_evolution\_id зависят только от него.  6) location(name, planet\_id, robotic\_construction\_id) в 2НФ так как есть первичный ключ и он не составной и неключевые атрибуты name, planet\_id,  robotic\_construction\_id зависят только от него.  7) planet(name, planet\_parameters\_id) в 2НФ так как есть первичный ключ и он не составной, значит отпадает проблема с зависимостью от всего первичного ключа целиком.  8) planet\_parameters(age, coordinates) в 2НФ так как есть первичный ключ и он не составной и неключевые атрибуты зависят только от него.  9) expedition(location\_id, person\_id) в 2НФ так как есть первичный ключ и он не составной и неключевые атрибуты зависят только от него. |
| 3NF: Отношение находится в 3NF, если оно находится во 2NF и не содержит транзитивных зависимостей. Моя модель удовлетворяет 3NF, так как все неключевые атрибуты зависят только от первичных ключей, и не содержат транзитивных зависимостей.  Таблицы:   1. person(name, profession\_id) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу. 2. profession(name, description) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.   3) cyber\_evolution(variation, description) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.  4) supercomputer(efficiency\_index) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.  5) robotic\_construction(weight\_kg, material, supercomputer\_id, cyber\_evolution\_id) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.  6) location(name, planet\_id, robotic\_construction\_id) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.  7) planet(name, planet\_parameters\_id) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.  8) planet\_parameters(age, coordinates) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу.  9) expedition(location\_id, person\_id) в 3НФ так как нет транзитивных зависимостей, то есть зависимостей не относящихся к первичному ключу. |

# 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.  Ключевые атрибуты не должны зависеть от неключевых  Таблицы:   1. person(name, profession\_id) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа. 2. profession(name, description) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.   3) cyber\_evolution(variation, description) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.  4) supercomputer(efficiency\_index) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.  5) robotic\_construction(weight\_kg, material, supercomputer\_id, cyber\_evolution\_id) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.  6) location(name, planet\_id, robotic\_construction\_id) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.  7) planet(name, planet\_parameters\_id) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.  8) planet\_parameters(age, coordinates) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа.  9) expedition(location\_id, person\_id) в BCNF так как первичный ключ никак не зависит от других атрибутов, атрибуты зависят от первичного ключа. |

# 4. Денормализация

|  |
| --- |
| **Объединение связанных таблиц**: В некоторых случаях, объединение таблиц может уменьшить количество операций JOIN и ускорить обработку запросов. Например, можно рассмотреть объединение таблиц person и profession, если часто запрашиваются данные о человеке и его профессии одновременно.  **Добавление избыточных атрибутов**: В некоторых случаях добавление избыточных атрибутов может улучшить производительность запросов. Например, если часто запрашивается количество людей определенной профессии, то можно добавить атрибут person\_count в таблицу profession. Это позволит избежать операций подсчета при каждом запросе, однако необходимо будет обновлять этот атрибут при добавлении или удалении людей. |

# Вывод

При выполнении лабораторной работы я познакомился с понятием нормализации и денормализации. Научился определять функциональные зависимости модели, а также анализировать последнюю на соответствие различным нормальным формам. Изучил эффективные способы денормализации схемы базы данных и ситуации, в которых возможно их применение.