Санкт-Петербургский политехнический университет им. Петра Великого

Кафедра компьютерных систем и программных технологий

**Отчёт по лабораторной работе №1**

**Дисциплина**: Базы данных

# Разработка структуры и нормализация БД.

Выполнил студент гр. 43501/3 Анисимов А.А.

Преподаватель: Мяснов А. В.

Санкт-Петербург

2016

## Цели работы

Познакомить студентов с основами проектирования схемы БД, способами нормализации отношений в БД.

## Программа работы

1. Представить SQL-схему БД, соответствующую заданию (должно получиться не менее 7 таблиц)
2. Привести схему БД к  3НФ
3. Согласовать с преподавателем схему БД. Обосновать соответствие схемы  3НФ.
4. Продемонстрировать результаты преподавателю

**Задание: Банковское хранилище.**

Хранит информацию о клиентах, сейфовых ячейках, аренде ячеек, стоимости аренды, стоимости хранимых предметов, доступе клиентов к ячейкам (условия доступа могут быть сложными - только один клиент, обязательно несколько клиентов, ...), история каждой ячейки.

**3. Выполнение работы**

В соответствии с заданием была составлена схема БД, представленная на Рис. 1.

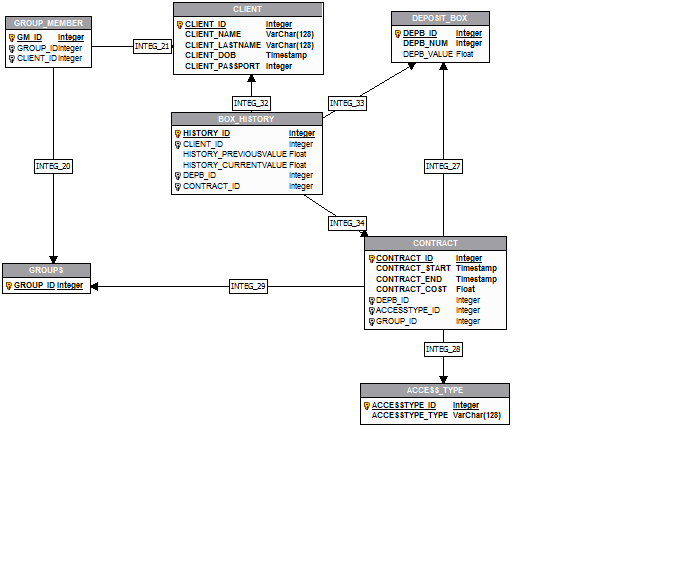


Рис. 1. Схема создаваемой БД.

Схема базы данных содержит следующие таблицы:

* CLIENT – таблица, содержащая информацию о клиентах
* DEPOSIT\_BOX – таблица содержащая информацию о ячейках
* CONTRACT – таблица контрактов (у одного клиента может быть несколько контрактов, контракт может быть заключен с несколькими клиентами)
* BOX\_HISTOTY – таблица, содержащая историю ячейки
* GROUP – таблица, содержащая информацию о группах.
* GROUP\_MEMBER – связывающая таблица, для огрганизации связи многие ко многим таблиц GROUP и CLIENT.
* ACCESS\_TYPE – таблица содержащая информацию по типам доступа к ячейке

Данная схема приведена к 3НФ.

Первая нормальная форма требует, чтобы на любом пересечении строки и столбца находилось единственное значение, которое должно быть атомарным (неделимым). В таблице, удовлетворяющей 1НФ, не должно быть повторяющихся групп.

Вторая нормальная форма основана на понятии полной функциональной зависимости. Атрибут В называется полностью функционально зависимым от атрибута А, если атрибут В функционально зависит от полного значения атрибута А и не зависит от какого-либо подмножества атрибута А. Отношение находится во 2НФ, если оно находится в 1НФ и каждый его атрибут, не входящий в состав первичного ключа, функционально полно зависит от первичного ключа. Другими словами, второе правило нормализации требует, чтобы любой не ключевой столбец зависел от всего первичного ключа, а не от его отдельных компонентов. Это правило относится к случаю, когда первичный ключ образован из нескольких столбцов.

Третья нормальная форма основана на понятии транзитивной зависимости. Если для атрибутов А, В и С некоторого отношения существуют зависимости С от В и В от А, то говорят, что атрибут С транзитивно зависит от атрибута А через атрибут В. Отношение находится в 3НФ, если оно находится в 1НФ и 2НФ, и в нем не существует транзитивных зависимостей неключевых атрибутов от первичного ключа. Другими словами, третья нормальная форма требует, чтобы ни один неключевой столбец не зависел бы от другого неключевого столбца. Любой неключевой столбец должен зависеть только от столбца первичного ключа.

**Выводы.**

**Нормальная форма** — свойство [отношения](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BD%D0%BE%D1%88%D0%B5%D0%BD%D0%B8%D0%B5_(%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C)) в [реляционной модели данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), характеризующее его с точки зрения избыточности, потенциально приводящей к логически ошибочным результатам выборки или изменения данных. Нормальная форма определяется как совокупность требований, которым должно удовлетворять отношение.

Процесс преобразования отношений [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) к виду, отвечающему нормальным формам, называется **нормализацией**.

Нормализация предназначена для приведения структуры БД к виду, обеспечивающему минимальную логическую избыточность, и не имеет целью уменьшение или увеличение производительности работы или же уменьшение или увеличение физического объёма базы данных. Конечной целью нормализации является уменьшение потенциальной противоречивости хранимой в базе данных информации.

В результате работы была получена БД приведенная к 3НФ. Она хороша тем, что позволяет избежать дублирования данных. Почти каждый атрибут, от которого зависят другие, является ключом.

3НФ позволяет быстрее выполнять операции вставки, обновления, удаления. Но за эти плюсы мы платим небольшой (по сравнению с 1НФ и 2НФ) скоростью выборки данных.

**Преимущества нормализации**

• Лучшая общая организация базы данных.

• Сокращение числа ненужных повторений данных.

• Согласованность данных внутри базы данных.

• Более гибкая структура базы данных.

• Эффективные возможности обеспечения безопасности и надежности базы данных.

**Недостатки нормализации**

Хотя большинство успешно работающих баз данных в некоторой степени нормализованы, нормализация имеет один существенный недостаток: замедление работы базы данных. Выполнение запроса или транзакции предполагает использование центрального процессора компьютера, памяти и операций ввода-вывода. Попросту говоря, в нормализованной базе данных для выполнения транзакций или запросов более интенсивно используется центральный процессор, требуется больше памяти и большее число операций ввода-вывода, чем в ненормализованной. В нормализованной базе данных требуется находить соответствующие таблицы и связывать данные для того, чтобы извлечь нужную информацию или обработать ее.