**Лабораторная работа №6. (УСРС)**

**Нормализация отношений**

1. **Первая нормальная форма**

* Устраните повторяющиеся группы в отдельных таблицах.
* Создайте отдельную таблицу для каждого набора связанных данных.
* Идентифицируйте каждый набор связанных данных с помощью первичного ключа.

Например, 1НФ:

В сущности «Клиент» было одно поле FullName, состоящее из трех полей имя, фамилия, отчество (рис. 1):

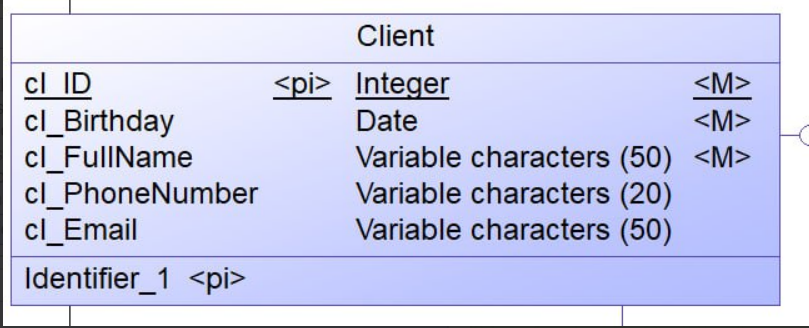


Рис. 1 — Поле Fullname

Поле FullName необходимо разбить на три отдельных поля FirstName, MiddleName, LastName (рис. 2):

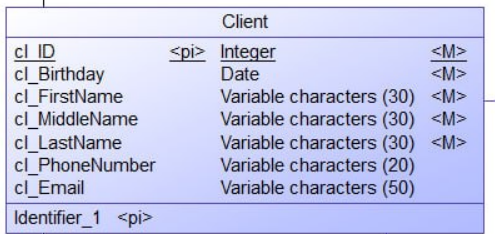


Рис. 2 — поля FirstName, MiddleName, LastName

1. **Вторая нормальная форма**

Данная схема базы данных соответствует второй нормальной форме (2NF) нормализации базы данных.

* Созданы отдельные таблицы для наборов значений, относящихся к нескольким записям.
* Связаны эти таблицы с помощью внешнего ключа.

1. **Третья нормальная форма.**

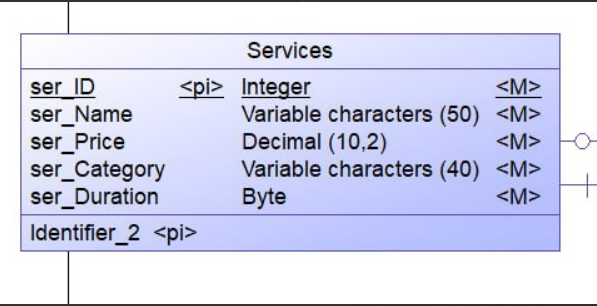
****

Рис. 3 — До нормализации по третьей форме

В нормализованной версии создается отдельная таблица Category для хранения категорий услуг, а поле категории удаляется из таблицы Services. Вместо этого в таблицу Services добавляется внешний ключ cat\_id, ссылающийся на поле id в таблице category (рис. 4).

Эта нормализация уменьшает избыточность, сохраняя категории в отдельной таблице и связывая их с помощью внешнего ключа, что повышает целостность данных, уменьшает пространство для хранения и позволяет эффективно запрашивать и поддерживать данные.

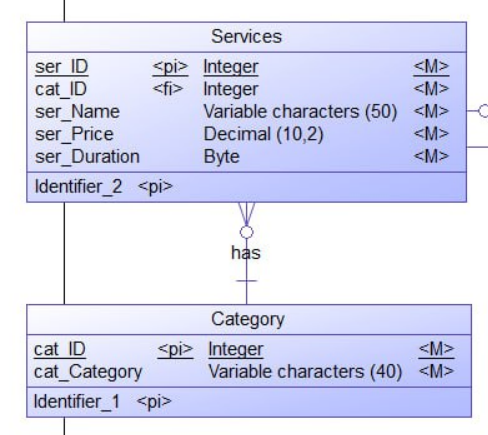


Рис. 4 — После нормализации по третьей форме

Данная схема базы данных соответствует третьей нормальной форме (3NF) нормализации базы данных.

Таблицы организованы таким образом, что исключается избыточность и гарантируется, что каждая таблица содержит только атомарные значения без каких-либо повторяющихся групп или транзитивных зависимостей. Шаги нормализации, применяемые к таблицам, следующие:

Поле category в Services было отделено в собственную таблицу Category, которая имеет идентификатор первичного ключа и атрибут category. Это устраняет транзитивную зависимость в исходной таблице сервисов, где категория зависела от идентификатора первичного ключа.

Отношения внешнего ключа были установлены между таблицами с использованием соответствующих ограничений первичного ключа и внешнего ключа, что обеспечивает ссылочную целостность и предотвращает несоответствия в данных.

Ни один непервичный атрибут в любой таблице не зависит от любого другого непервичного атрибута, что позволяет избежать транзитивных зависимостей.

В целом данная схема базы данных хорошо организована и нормализована, придерживаясь принципов третьей нормальной формы (3NF) нормализации базы данных.