

# Нормализация баз данных

---

ПРИМЕРЫ

Пример. Дана таблица БД

Нормализации баз данных					
Таблица 13 – Исходное отношение					
Код сотрудника	ФИО	Должность	Номер отдела	Наименование отдела	Квалификация
7513	Иванов И.И.	Программист	128	Отдел проектирования	C, Java
9842	Сергеева С.С.	Администратор БД	42	Финансовый отдел	DB2
6651	Петров П.П.	Программист	128	Отдел проектирования	VB, Java
9006	Николаев Н.Н.	Системный администратор	128	Отдел проектирования	Windows, Linux

# Приведем к 1 НФ

## Нормализация баз данных - 1НФ

Отношение находится **в первой нормальной форме (1НФ)**, когда каждая строка содержит только одно значение для каждого атрибута (столбца), то есть все атрибуты отношения имеют единственное значение (являются **атомарными**).

В столбце **Квалификация** ненормализованной таблицы 13 содержатся списки значений (С, Java и т. д.).

Чтобы привести схему к 1НФ, необходимо разместить в этом столбце атомарные значения.

Самый простой способ заключается в выделении по одной строке на каждый элемент квалификации (таблица 14).

## Нормализация баз данных - 1НФ

Таблица 14

Код сотрудника	ФИО	Должность	Номер отдела	Наименование отдела	Квалификация
7513	Иванов И.И.	Программист	128	Отдел проектирования	С
7513	Иванов И.И.	Программист	128	Отдел проектирования	Java
9842	Сергеева С.С.	Администратор БД	42	Финансовый отдел	DB2
6651	Петров П.П.	Программист	128	Отдел проектирования	VB
6651	Петров П.П.	Программист	128	Отдел проектирования	Java
9006	Николаев Н.Н.	Системный администратор	128	Отдел проектирования	Windows
9006	Николаев Н.Н.	Системный администратор	128	Отдел проектирования	Linux

# Приведем к 2 НФ

## Нормализация баз данных - 2НФ

Таблица **Квалификации\_сотрудников** (таблица 14) находится в 1НФ, но не удовлетворяет 2НФ. Первичный ключ должен уникальным образом идентифицировать каждую строку. Единственным вариантом является использование в качестве первичного ключа комбинации **Код сотрудника** и **Квалификация**.

Это порождает схему:

**Квалификации\_сотрудников** (**Код сотрудника**, **ФИО**, **Должность**, **Номер отдела**, **Наименование отдела**, **Квалификация**).

2НФ

## Нормализация баз данных - 2НФ

Одной из имеющихся здесь функциональных зависимостей будет: **Код сотрудника, Квалификация → ФИО, Должность, Номер отдела, Наименование отдела**. Но, кроме того, мы также имеем: **Код сотрудника → ФИО, Должность, Номер отдела, Наименование отдела**. Другими словами, можно определить имя, должность и отдел, используя только код сотрудника. Это значит, что указанные атрибуты функционально зависимы только от части первичного ключа, а не от всего первичного ключа. Следовательно, указанная схема не находится в 2НФ.

2НФ

## Нормализация баз данных - 2НФ

Для приведения этой схемы в 2НФ необходимо декомпозировать исходное отношение на два, в которых все не ключевые атрибуты будут полностью функционально зависеть от ключа: **сотрудники** (Код сотрудника, ФИО, Должность, Номер отдела, Наименование отдела) и **Квалификации сотрудников** (Код сотрудника, Квалификация) (таблицы 15-16).

## Нормализация баз данных - 2НФ

Таблица 15

Код сотрудника	ФИО	Должность	Номер отдела	Наименование отдела
7513	Иванов И.И.	Программист	128	Отдел проектирования
9842	Сергеева С.С.	Администратор БД	42	Финансовый отдел
6651	Петров П.П.	Программист	128	Отдел проектирования
9006	Николаев Н.Н.	Системный администратор	128	Отдел проектирования

Таблица 16

Код сотрудника	Квалификация
7513	C
7513	Java
9842	DB2
6651	VB
6651	Java
9006	Windows
9006	Linux

2НФ

# Приведем к 3 НФ

## Нормализация баз данных - 3НФ

Отношение находится **в третьей нормальной форме (3НФ)**, если оно находится во 2НФ и ни один из его не ключевых атрибутов не связан функциональной зависимостью с любым другим не ключевым атрибутом. Атрибуты, зависящие от других не ключевых атрибутов, нормализуются путем перемещения зависимого атрибута и атрибута, от которого он зависит, в новое отношение.

Формально, для приведения схемы в 3НФ необходимо исключить все транзитивные зависимости. Схема отношения **сотрудники** (таблица 15) содержит следующие функциональные зависимости: **Код сотрудника** → **ФИО, Должность, Номер отдела, Наименование отдела** и **Номер отдела** → **Наименование отдела**.

## Нормализация баз данных - 3НФ

Первичным ключом является **Код сотрудника**, и все атрибуты полностью функционально зависимы от него (первичный ключ определяется единственным атрибутом). При этом **Номер отдела** ключом не является.

Функциональная зависимость **Код сотрудника** → **Наименование отдела** является транзитивной, поскольку содержит промежуточный шаг (зависимость **Номер отдела** → **Наименование отдела**). Для приведения в 3НФ необходимо исключить эту транзитивную зависимость, декомпозируя отношение на два: **сотрудники** (**Код сотрудника, ФИО, Должность, Номер отдела**) и **отделы** (**Номер отдела, Наименование отдела**) (таблицы 17-18).

## Нормализация баз данных - ЗНФ

### Таблица 17

Код сотрудника	ФИО	Должность	Номер отдела
7513	Иванов И.И.	Программист	128
9842	Сергеева С.С.	Администратор БД	42
6651	Петров П.П.	Программист	128
9006	Николаев Н.Н.	Системный администратор	128

ЗНФ

### Таблица 18

Номер отдела	Наименование отдела
42	Финансовый отдел
128	Отдел проектирования



## Нормализация баз данных - БКНФ

**Нормальная форма Бойса-Кодда (БКНФ)** является развитием ЗНФ и требует, чтобы в отношении были только такие функциональные зависимости, левая часть которых является потенциальным ключом отношения. **Потенциальный ключ** представляет собой атрибут (или множество атрибутов), который может быть использован для данного отношения в качестве первичного ключа. Фактически первичный ключ - это один из потенциальных ключей, назначенный в качестве первичного. **Детерминантом** называется левая часть функциональной зависимости. Отношение находится в БКНФ тогда и только тогда, когда каждый детерминант отношения является потенциальным ключом.

## Нормализация баз данных - БКНФ

**Алгоритм приведения ненормализованных схем в ЗНФ показан на рисунке 15.**

**На практике построение ЗНФ в большинстве случаев является достаточным и приведением к ней процесс построения реляционной БД заканчивается.**

## Алгоритм приведения ненормализованных схем в 3НФ

ненормализованная  
схема

1. Расцепить схему на схемы без повторяющихся групп
2. Объявить один или более атрибутов главным ключом (самый малый ключ)

1НФ

1. Для схем, где ключи имеют более одного атрибута, проверить, что каждый неключевой атрибут полностью функционально зависим от всего ключа
2. Если это не так, расцепить схему

2НФ

1. Проверить, что все неключевые атрибуты являются взаимно-независимыми
2. Исключить избыточные атрибуты или расцепить схему

3НФ

3НФ

## Нормализация баз данных - БКНФ

Запомнить правила нормализации помогает изречение:

**«Нормализация - это ключ, целый ключ и ничего, кроме ключа».**

Нормальные формы высших порядков (4НФ и 5НФ) представляют большой интерес для теоретических исследований, чем для практики проектирования БД. В них учитываются многозначные зависимости между атрибутами. *Полной декомпозицией отношений* называют такую совокупность произвольного числа его проекций, соединение которых позволяет получить исходное отношение.