

Нормализация таблиц реляционной базы данных



Понятие нормализации

Нормализация - это формальный аппарат ограничений на формирование таблиц, который позволяет устранить дублирование, обеспечивает непротиворечивость хранимых данных и уменьшает трудозатраты на ведение базы данных.

Цели нормализации

- Обеспечить быстрый доступ к данным
- Исключить ненужное повторение данных, которое может являться причиной ошибок при вводе, а также привести к нерациональному использованию дискового пространства.
- Обеспечить целостность данных, т.е. чтобы при изменении одних объектов автоматически происходило соответствующее изменение связанных с ними объектов.

Окончательная цель нормализации- получение такого проекта базы данных, в котором каждый факт появляется лишь в одном месте

Нормальные формы таблиц

Первая нормальная форма (**First Normal Form — 1NF**)

Вторая нормальная форма (**Second Normal Form — 2NF**)

Третья нормальная форма (**Third Normal Form — 3NF**)

Нормальная форма Бойса-Кодда (**Brice-Codd Normal Form — BCNF**)

Четвертая нормальная форма (**Fourth Normal Form - 4NF**)

Пятая нормальная форма - нормальная форма проекции-соединения
(**Fifth Normal Form - 5NF**, или **PJ/NF**)

Каждая последующая форма должна отвечать требованиям предыдущей плюс некоторые дополнительные требования

При описании нормальных форм используются следующие понятия:

- функциональная зависимость между полями
- полная функциональная зависимость между полями
- многозначная функциональная зависимость между полями
- транзитивная функциональная зависимость между полями
- взаимная независимость между полями

Функциональной зависимостью между полями **A** и **B** называется зависимость, при которой каждому значению **A** в любой момент времени соответствует единственное значение **B** из всех возможных.

Полной функциональной зависимостью между *составным* полем **A** и полем **B** называется зависимость, при которой поле **B** зависит функционально от всего поля **A** и не зависит функционально от любого подмножества поля **A**.

Многозначная функциональная зависимость между полями определяется следующим образом: поле **A** многозначно определяет поле **B**, если для каждого значения поля **A** существует «хорошо определенное множество» соответствующих значений поля **B**.

Транзитивная функциональная зависимость между полями **A** и **C** существует в том случае, если поле **C** функционально зависит от поля **B**, а поле **B** функционально зависит от поля **A**, при этом *не существует* функциональной зависимости поля **A** от поля **B**.

Взаимная независимость между полями определяется следующим образом: несколько полей взаимно независимы, если ни одно из них не является функционально зависимым от другого.

Первая нормальная форма (1НФ – 1NF)

Таблица находится в **первой нормальной форме** тогда и только тогда, когда ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица, находящаяся в 1НФ, должна отвечать следующим **требованиям**:

- *Таблица не должна иметь повторяющихся записей*
- *В таблице должны отсутствовать повторяющиеся группы полей*

Любая таблица в реляционной базе данных автоматически находится в первой нормальной форме (по определению).

В такой таблице не должно содержаться полей, которые можно было бы разделить на несколько полей.

Приведение таблицы к 1НФ

- 1) Определить поле, которое можно назначить первичным ключом. Если такого поля нет, то добавить уникальное ключевое поле
- 2) Определить группы повторяющихся полей
- 3) Вынести группы повторяющихся полей в отдельные таблицы, в основной таблице остается только одно поле для организации связи между таблицами
- 4) Назначить первичные ключи в новых таблицах
- 5) Определить тип отношения между таблицами

Пример приведения таблицы к 1НФ

Пусть дана однотабличная БД «Заказы»:

Дата заказа
ФИО клиента
Телефон клиента
Номер счета
Наименование товара
Цена товара
Фирма производитель
Адрес фирмы производителя
Количество товара
Оплата товара

**Таблица не удовлетворяет
требованиям 1НФ**

- ✓ Не определён первичный ключ
- ✓ Содержит повторяющиеся группы полей:
ФИО клиента и Телефон клиента

Приведение БД к 1НФ

Дата заказа
ФИО клиента
Контактный телефон
Номер счета
Наименование товара
Цена
Фирма производитель
Адрес фирмы производителя
Количество товара
Оплата

Таблица «Заказы»

Код заказа
Дата заказа
Количество товара
Оплата

Таблица «Клиенты»

Код клиента
ФИО клиента
Контактный телефон
Номер счета

Таблица «Товары»

Код товара
Наименование товара
Цена

Таблица «Производители»

Код производителя
Фирма производитель
Адрес фирмы производителя

Определяем связи



Вторая нормальная форма (2НФ – 2NF)

Таблица находится во **второй нормальной форме**, если она удовлетворяет определению 1НФ и все ее поля, не входящие в первичный ключ, связаны **полной функциональной зависимостью** с первичным ключом (каждый неключевой атрибут зависит от всего составного ключа и не зависит от его частей – нет частичнозависимых полей).

Для приведения таблицы к 2НФ необходимо:

1. Вынести все частично-зависимые поля в отдельную таблицу
2. Определить ключевое поле
3. Установить отношения между таблицами

Пример: таблица «Проекты»

Код проекта
Название
Основная цель
Продолжительность
Код руководителя проекта
Фамилия руководителя
Телефон руководителя

Составной ключ

Код проекта
Код руководителя проекта

Есть ли в таблице частично-зависимые поля?

Название
Основная цель
Продолжительность

Код проекта

Код руководителя

Фамилия руководителя
Телефон руководителя

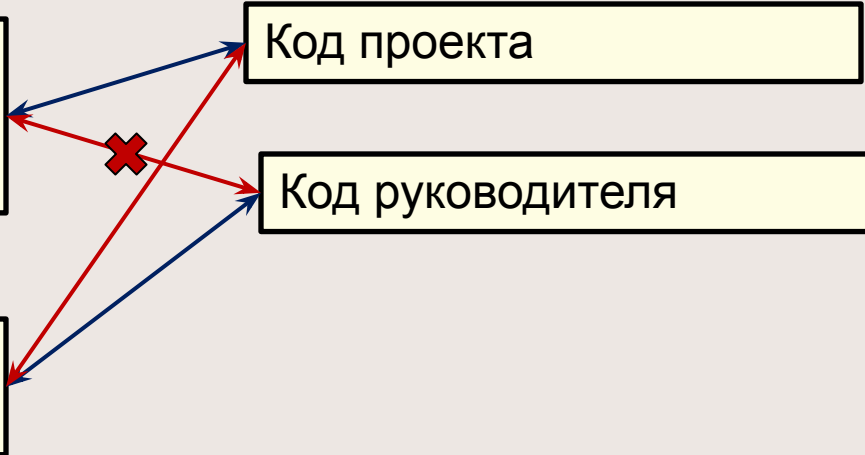


Таблица «Руководители»

Таблица «Проекты»

Код проекта
Название
Основная цель
Продолжительность
Код руководителя

M

1

Код руководителя
Фамилия руководителя
Телефон руководителя



Третья нормальная форма (3НФ – 3NF)

Таблица находится в **третьей нормальной форме**, если она удовлетворяет определению 2НФ и ни один из ее атрибутов **функционально независим** от любого другого атрибута.

Таблица в 3НФ:

- должна отвечать требованиям 2НФ
- все не ключевые атрибуты взаимно независимы (нет полей, не зависящих от ключа)

Алгоритм приведения к 3НФ:

1. Вынести все поля, не зависящие от ключа, в отдельную таблицу
2. Определить ключевые поля
3. Установить отношения

Приведение таблицы к 3НФ

Таблица «Заказы»

Код заказа

Фамилия покупателя

Дата продажи

Код менеджера

Фамилия менеджера

Таблица не находится в 3НФ:

поле **Фамилия менеджера**

зависит от поля **Код менеджера**

Таблица «Заказы»

Код заказа

Фамилия покупателя

Дата продажи

Код менеджера

Таблица «Менеджеры»

Код менеджера

Фамилия менеджера

М

1

ЗАДАНИЕ:

Дана таблица **Предметы** (*Код предмета*, Название, Цикл, Объем часов, Преподаватели)
Необходимо последовательно привести её к **1НФ**, **2НФ** и **3НФ**.

Приведение таблицы к 1 НФ

ПРЕДМЕТ (*Код предмета*, Название, Цикл, Объем часов)

ПРЕПОДАВАТЕЛЬ (*Код преподавателя*, ФИО, Должность, Контакты, Код предмета)

1НФ

ПРЕДМЕТ (*Код предмета*, Название, Цикл, Объем часов)

ПРЕПОДАВАТЕЛЬ (*Код преподавателя*, ФИО, Должность, Оклад, Контакты)

ИЗУЧЕНИЕ (*Код предмета*, *Код преподавателя*)

Приведение таблицы к 2НФ

2НФ

ПРЕДМЕТ (*Код предмета*, Название, Объем часов, Код цикла)

ЦИКЛ (*Код цикла*, Название цикла)

ПРЕПОДАВАТЕЛЬ (*Код преподавателя*, ФИО, Должность, Оклад, Контакты)

ИЗУЧЕНИЕ (*Код предмета*, Код преподавателя)

Приведение таблицы к 3НФ

3НФ

ПРЕДМЕТ (*Код предмета*, Название, Объем часов, Код цикла)

ЦИКЛ (*Код цикла*, Название цикла)

ПРЕПОДАВАТЕЛЬ (*Код преподавателя*, ФИО, Код должности, Контакты)

ДОЛЖНОСТЬ (*Код должности*, Название должности, Оклад)

ИЗУЧЕНИЕ (*Код предмета*, Код преподавателя)