



Проектирование реляционных БД

План

1. Реляционные объекты данных.
2. Нормализация отношений:
 - обзор нормальных форм;
 - декомпозиция без потерь;
 - первая, вторая и третья нормальные формы;
 - итоговая схема процедуры нормализации.
3. Целостность реляционных данных
 - потенциальные, первичные и альтернативные ключи;
 - внешние ключи. Правило ссылочной целостности.
 - типы связей. Правила внешних ключей.
 - Null-значения. Правило целостности объектов.

Реляционная система

(relation - отношение)

- ✓ была предложена сотрудником фирмы IBM *Эдгаром Коддом* в 1962 г.
- ✓ это система, основанная на следующих принципах:
 - данные представлены пользователям только в виде таблиц;
 - пользователю предоставляются операторы, генерирующие новые таблицы из основных.

Реляционные объекты



- **Отношением** называется вся таблица.
- **Атрибут (поле)** - столбец таблицы.
- **Степень отношения** – количество его атрибутов.
- **Кортеж (запись)** - заполненная строка таблицы.
- **Кардинальное число** – количество кортежей.
- **Первичный ключ** – это атрибут (или атрибуты), значения которого уникально идентифицируют кортежи.
- **Домен (тип данных)** – это общая совокупность значений для конкретного атрибута.

Свойства отношений

- Нет одинаковых кортежей.

Следствие: в отношении всегда существует первичный ключ.

- Кортежи неупорядочены.
- Атрибуты неупорядочены.
- Все значения атрибутов неделимы.

Обозначение отношения

ИмяОтношения (ИмяАтрибута1,
ИмяАтрибута2, ..., ИмяАтрибутаN),

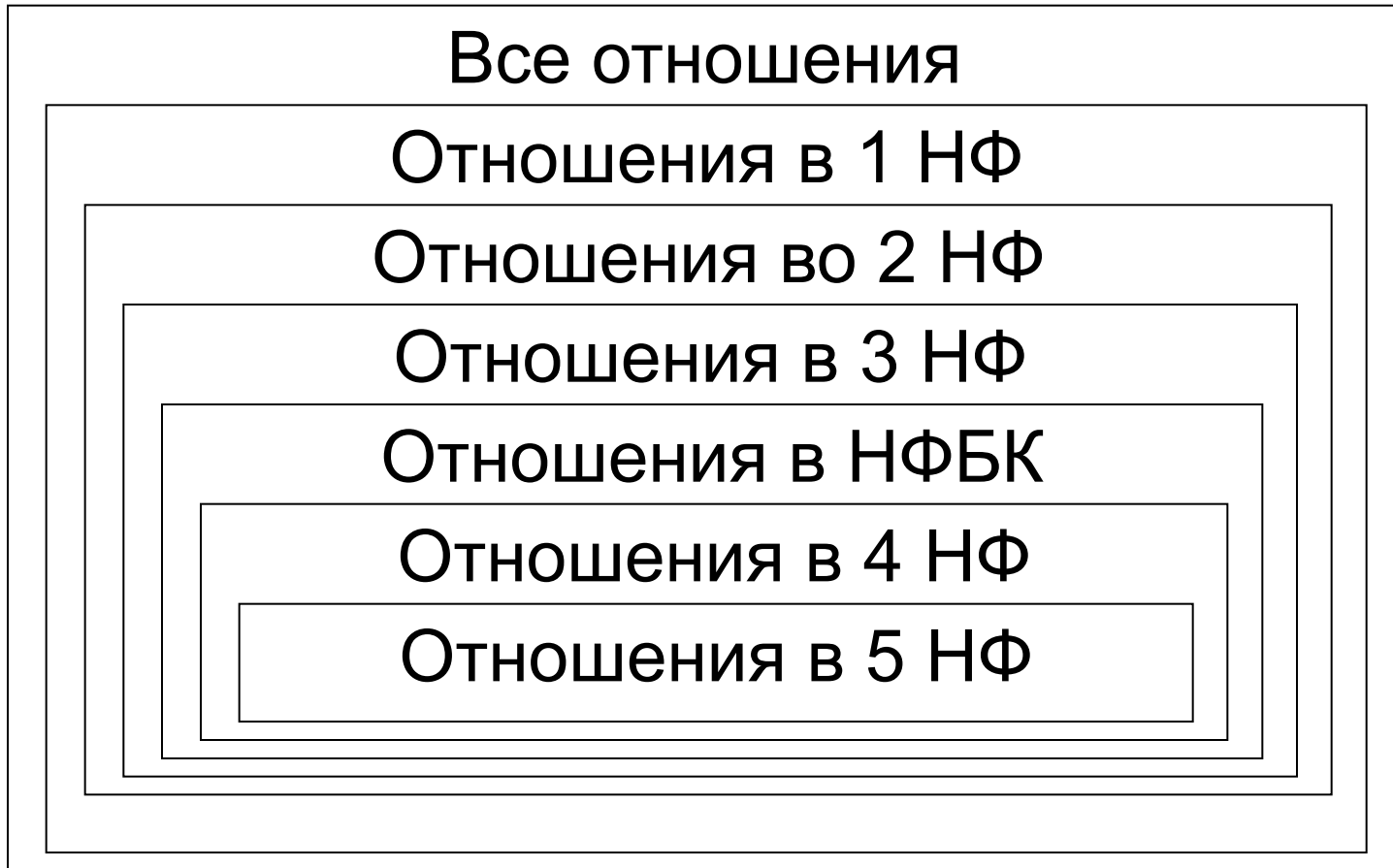
где подчеркнутый атрибут -
первичный ключ, N – степень
отношения.

Например,

Студенты (№, фамилия, имя,
отчество, адрес, телефон).

Проектирование реляционных баз данных

- **Нормализация** – это процесс реструктуризации базы, направленный на устранение избыточности данных.
- Отношение находится **в некоторой НФ**, если оно удовлетворяет заданному набору условий.



Все условия, необходимые для некоторой НФ, должны выполняться и для всех последующих НФ!

- 1-3 НФ определил Э. Кодд
- НФБК – НФ Бойса-Кодда
- 4 и 5 НФ - Р. Фейджин

Если отношение не находится ни во 2НФ, ни в 3НФ, существует избыточность, которая приводит к нарушению целостности данных.

ДЕКОМПОЗИЦИЯ - разбиение отношения на другие, более мелкие отношения (с меньшим количеством атрибутов).

Декомпозиция должна быть обратимой.

Пример: БД «Факультет»

Состоит из таблиц:

- «Студенты»;
- «Группы»;
- «Преподаватели»;
- «Дисциплины»;
- «Занятия».

Студенты (НомерЗачетнойКнижки,
Фамилия, Имя, Группа, Адрес, Телефон)

Декомпозиция

1 вариант:

Студенты1 (НомерЗачетнойКнижки, Фамилия, Имя)
Студенты2 (Фамилия, Группа, Адрес, Телефон).

Декомпозиция
не обратима!

2 вариант:

Студенты3 (НомерЗачетнойКнижки, Фамилия, Имя)
Студенты4 (НомерЗачетнойКнижки, Группа, Адрес,
Телефон).

Декомпозиция
обратима!

Нормальные формы

- **Отношение находится в 1НФ тогда и только тогда, когда значения всех атрибутов (полей) неделимы.**

Пример: ФИО – не одно поле, а три!

- **Отношение находится во 2НФ тогда и только тогда, когда оно находится в 1НФ и каждый неключевой атрибут (поле) зависит от всего первичного ключа.**

Пример:


- Успеваемость** (НомерЗачетнойКнижки, Фамилия, Имя, Отчество, Дисциплина, Оценка)
- **Студенты** (НомерЗачетнойКнижки, Фамилия, Имя, Отчество)
 - **Успеваемость1** (НомерЗачетнойКнижки, Дисциплина, Оценка).

- Отношение находится в 3НФ тогда и только тогда, когда оно находится во 2НФ и между неключевыми атрибутами (полями) отсутствует взаимозависимость.

Пример:

Студенты (НомерЗачетнойКнижки, Фамилия, Имя, Группа, СтаростаГруппы)

- **Студенты1** (НомерЗачетнойКнижки, Фамилия, Имя, Отчество, Группа)
- **Группы** (Группа, СтаростаГруппы)

- 
- Если отношение не находится в 3 НФ, существует избыточность, которая приводит к **аномалиям обновления**, т. е. нарушению целостности при вставке, удалении или изменении данных.

Итоговая схема процедуры нормализации

Цели:

- Исключение *избыточности*.
- Устранение *аномалий* обновления.
- Проектирование макета данных, *соответствующего реальности*.
- Упрощение процесса наложения ограничений *целостности*.

Этапы нормализации до 3НФ

- Каждый атрибут первоначального отношения разбить на множество атрибутов таким образом, чтобы все значения стали **неделимыми** (1 НФ).
- Отношение в 1 НФ разбить на другие отношения **для исключения зависимостей неключевых атрибутов от части первичного ключа** (2 НФ).
- Отношения во 2 НФ следует разбить на другие отношения **для исключения зависимостей между неключевыми атрибутами** (3 НФ).

Целостность реляционных БД

Потенциальный ключ K отношения – это подмножество атрибутов, обладающее следующими свойствами:

- **уникальность** (нет двух различных кортежей с одинаковыми значениями K);
- **неизбыточность** (никакое подмножество K не обладает свойством уникальности).



Примеры:

- **Студенты** (НомерЗачетнойКнижки,
Фамилия, Имя, Группа, Адрес, Телефон)
- **Занятия** (Группа, Дисциплина,
Преподаватель, ВидЗанятия)

- Среди потенциальных ключей один назначают **первичным (Primary key)**, остальные – **альтернативные**.

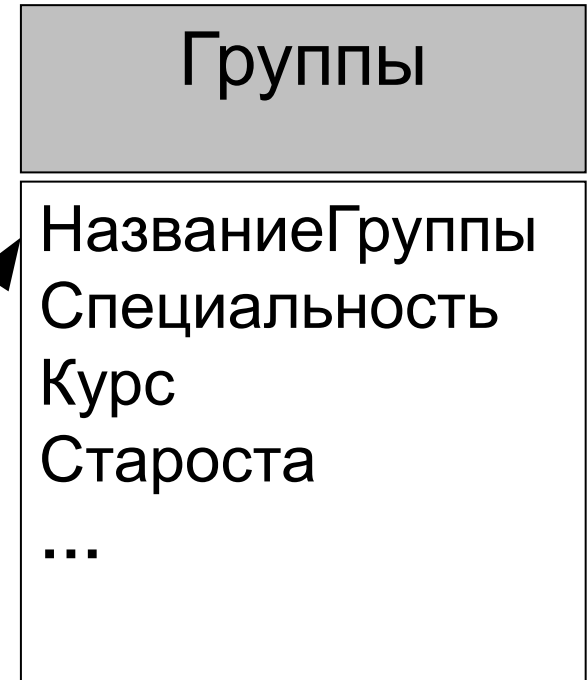
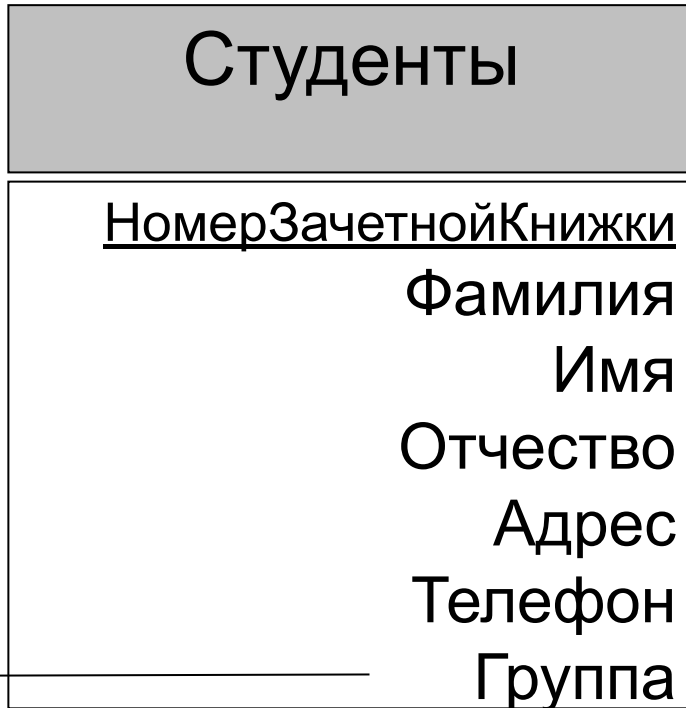
Пример:

Дисциплина (Код, Название,
Кол-во_Лекц, Кол-во_Пр, Общ_Объем)

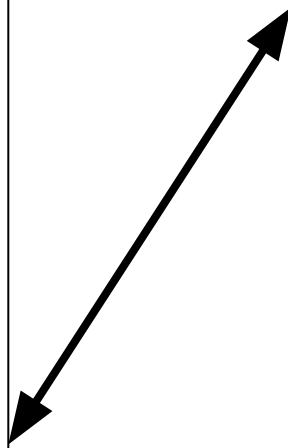
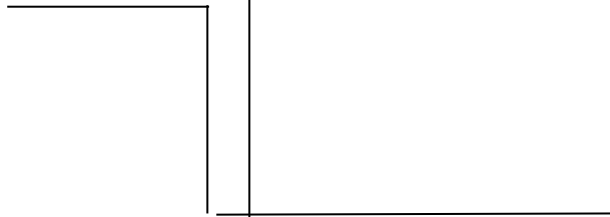
Внешние ключи


Подчиненное отношение

Главное отношение



Внешний
ключ





Внешний ключ – это атрибут (поле),
ссылающийся на первичный ключ
другой таблицы.

Отношение, на которое ссылается
внешний ключ, называется **главным**
(**ССЫЛОЧНЫМ**).

Отношение, содержащее внешний ключ,
называется **подчиненным**
(**ССЫЛАЮЩИМСЯ**).

Правило ссылочной целостности

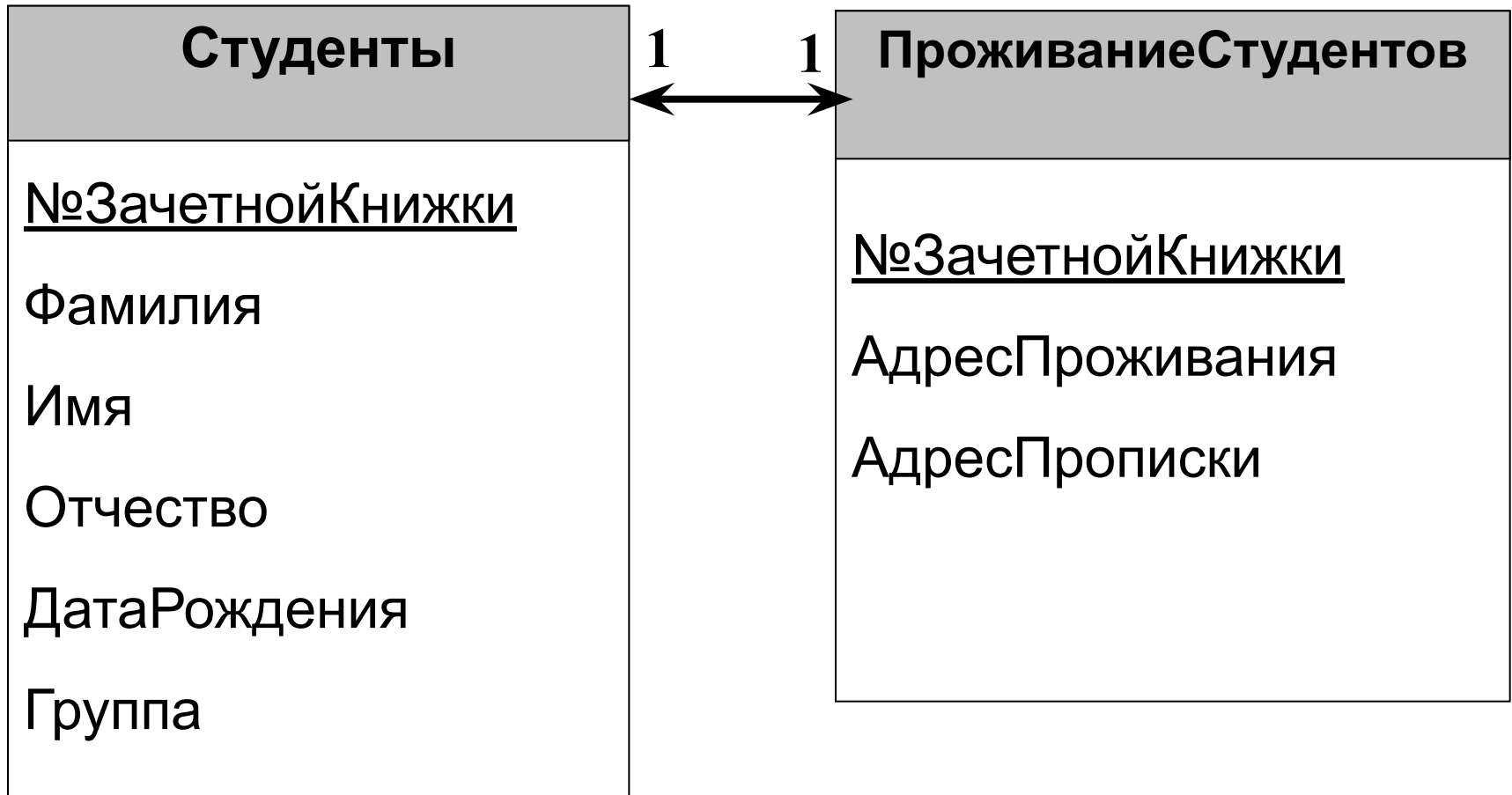
- БД не должна содержать несогласованных значений внешнего ключа.

Типы связей между отношениями в реляционной БД

1. Один-к-одному:

каждому кортежу (записи) первого отношения соответствует не более одного кортежа (записи) второго отношения и наоборот.

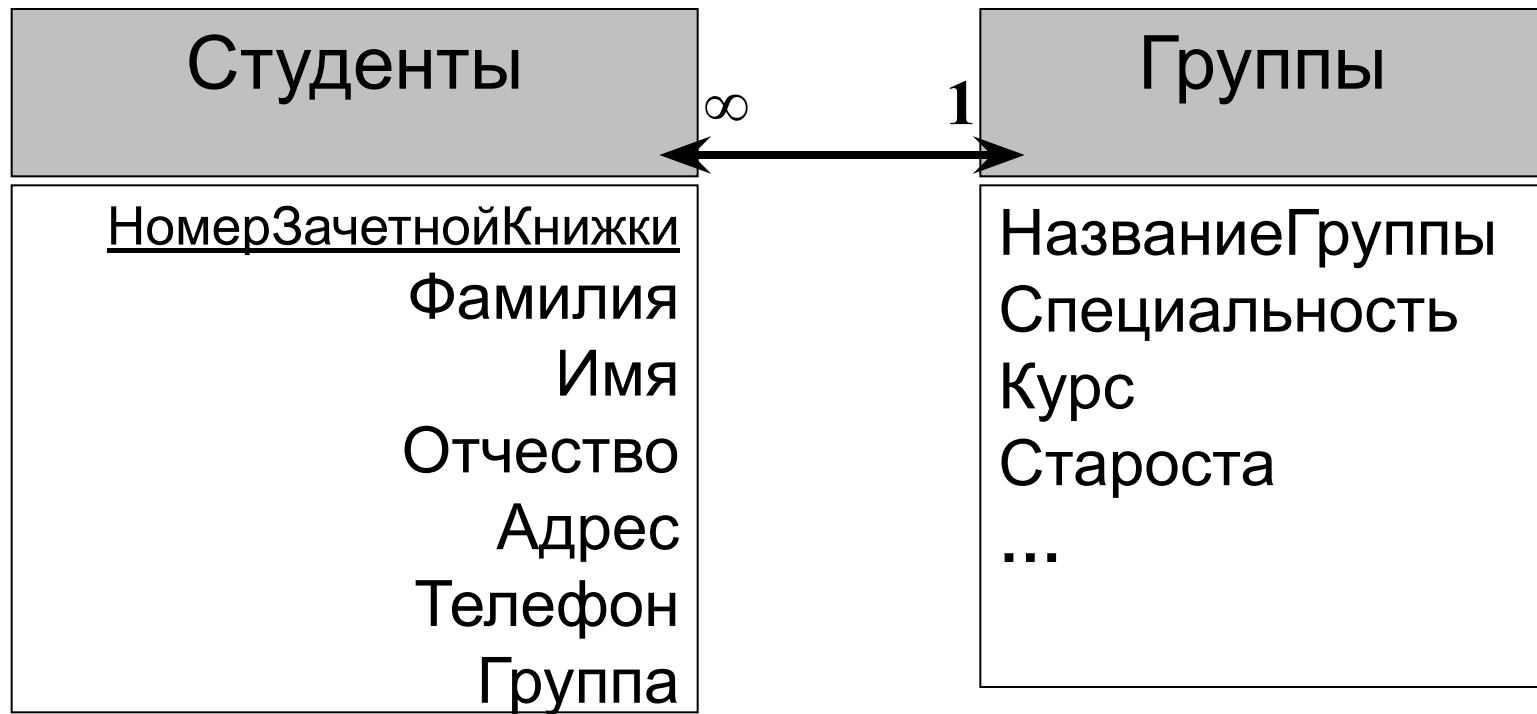
ПРИМЕР:



2. **Один-ко-многим:**

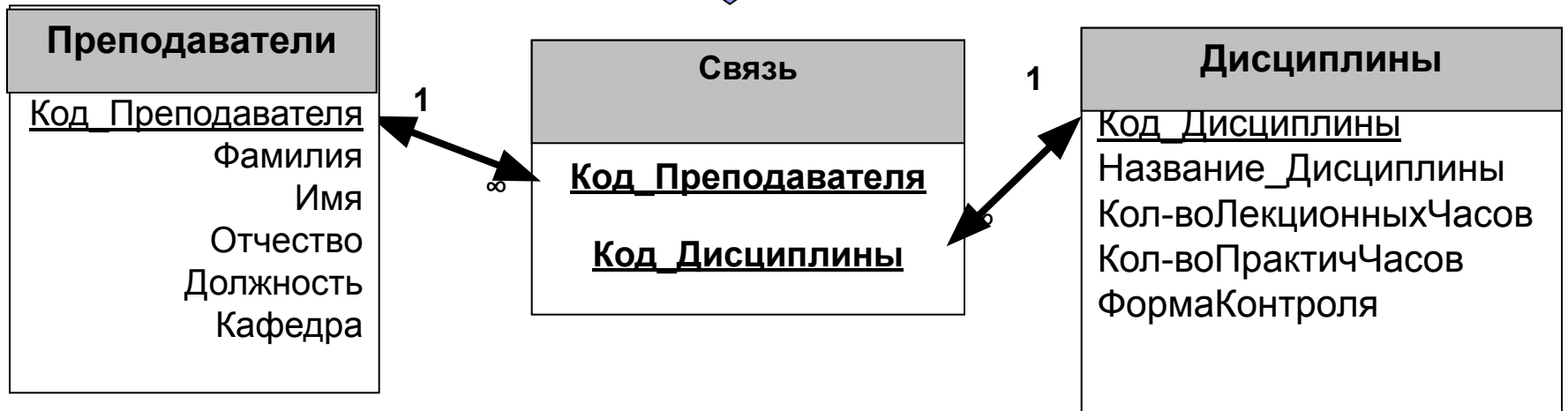
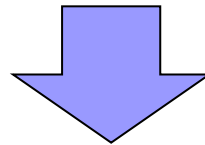
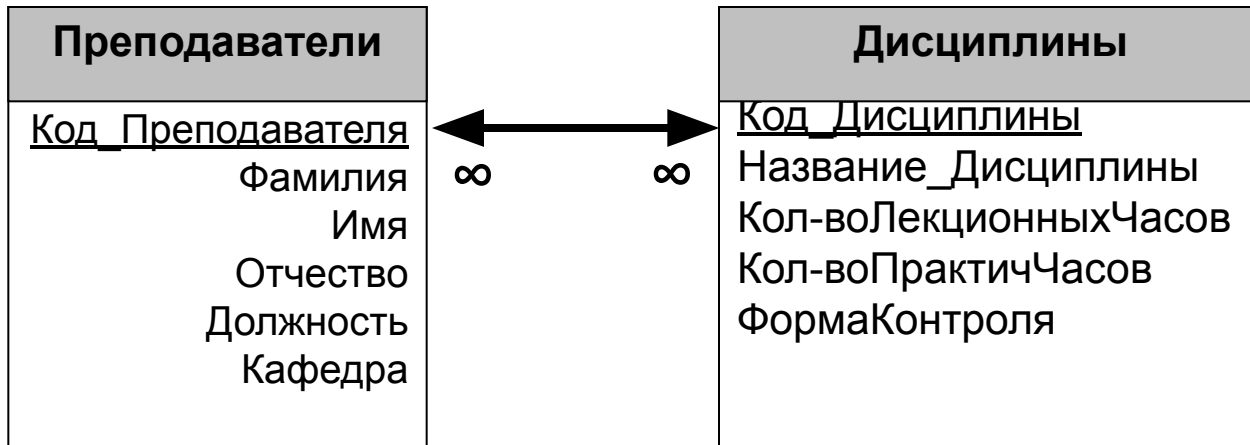
кортежу (записи) главного отношения соответствует ноль, один или несколько кортежей (записей) подчиненного отношения, а каждому кортежу (записи) подчиненного отношения соответствует только один кортеж (запись) главного отношения.

ПРИМЕР:



3. **Многие-ко-многим:**

каждому кортежу (записи) первого отношения соответствует ноль, один или несколько кортежей (записей) второго отношения, и наоборот.



Правила внешних ключей

Цель: сохранить целостность БД во время ее изменения/обновления

КОМПЕНСИРУЮЩИЕ ОПЕРАЦИИ



ограничить

каскадировать

Null-значения. Правило целостности объектов

- **Null-значение** – это метка для обозначения отсутствия любого значения.
- Ни один элемент **первичного ключа** базового отношения **не может быть** Null-значением.
- Внешний ключ может быть Null-значением, если:
 - временно отсутствуют данные,
 - осуществляется *каскадное удаление данных*.