

---

# Реляционная модель

---

---

# Рассматриваемые вопросы:

## 1. Реляционная модель

- Краткий обзор истории реляционной модели
- Используемая терминология
- Альтернативная терминология
- Математические отношения
- Отношения и их свойства в базе данных
- Реляционные ключи
- Представление схем в реляционной базе данных
- Реляционная целостность

## 2. Реляционные языки

## 3. Реляционная алгебра

- Унарные операции реляционной алгебры
- Операции с множествами
- Операции соединения
- Деление

## 4. Реляционное исчисление

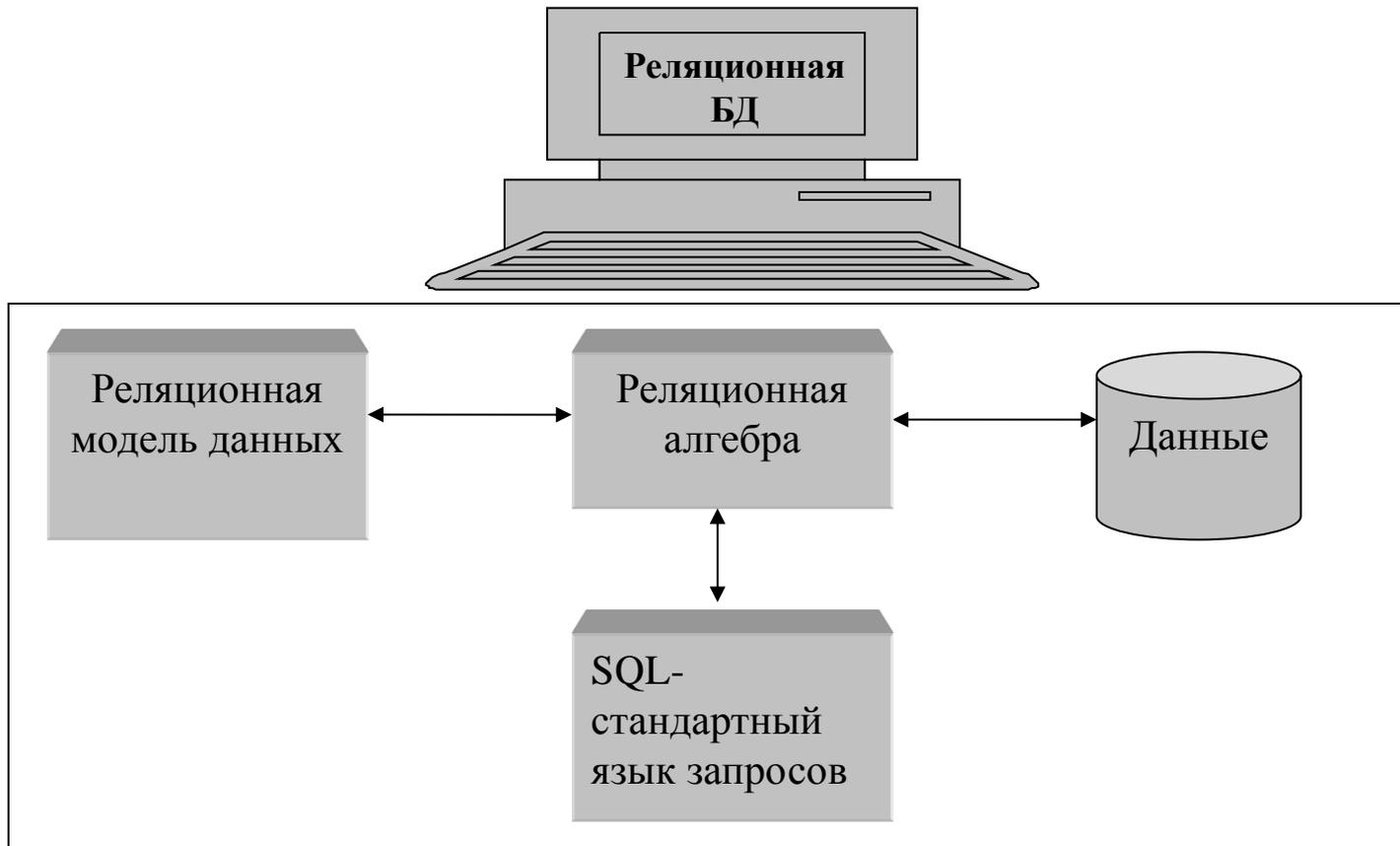
- Реляционное исчисление кортежей
- Реляционное исчисление доменов

## 5. Другие языки

---

# Реляционная модель

## Структура обработки информации в реляционной БД



---

# Реляционная модель

## Цели создания реляционной модели:

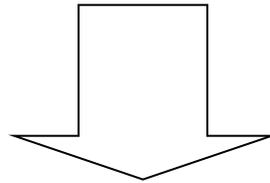
- 1) Обеспечение высокой степени независимости от данных.
  - 2) Нормализация отношений, т.е. создание отношений без повторяющихся групп.
  - 3) Расширение языков управления данными за счет включения операций над множествами.
-

---

# Реляционная модель

## Используемая терминология

По сколько Кодд, будучи опытным математиком, широко использовал математическую терминологию теории множеств и логики предикатов.



*Реляционная модель основана на математическом понятии отношения, физическим представлением которого является таблица.*

---

# Используемая терминология

## Структура реляционных данных

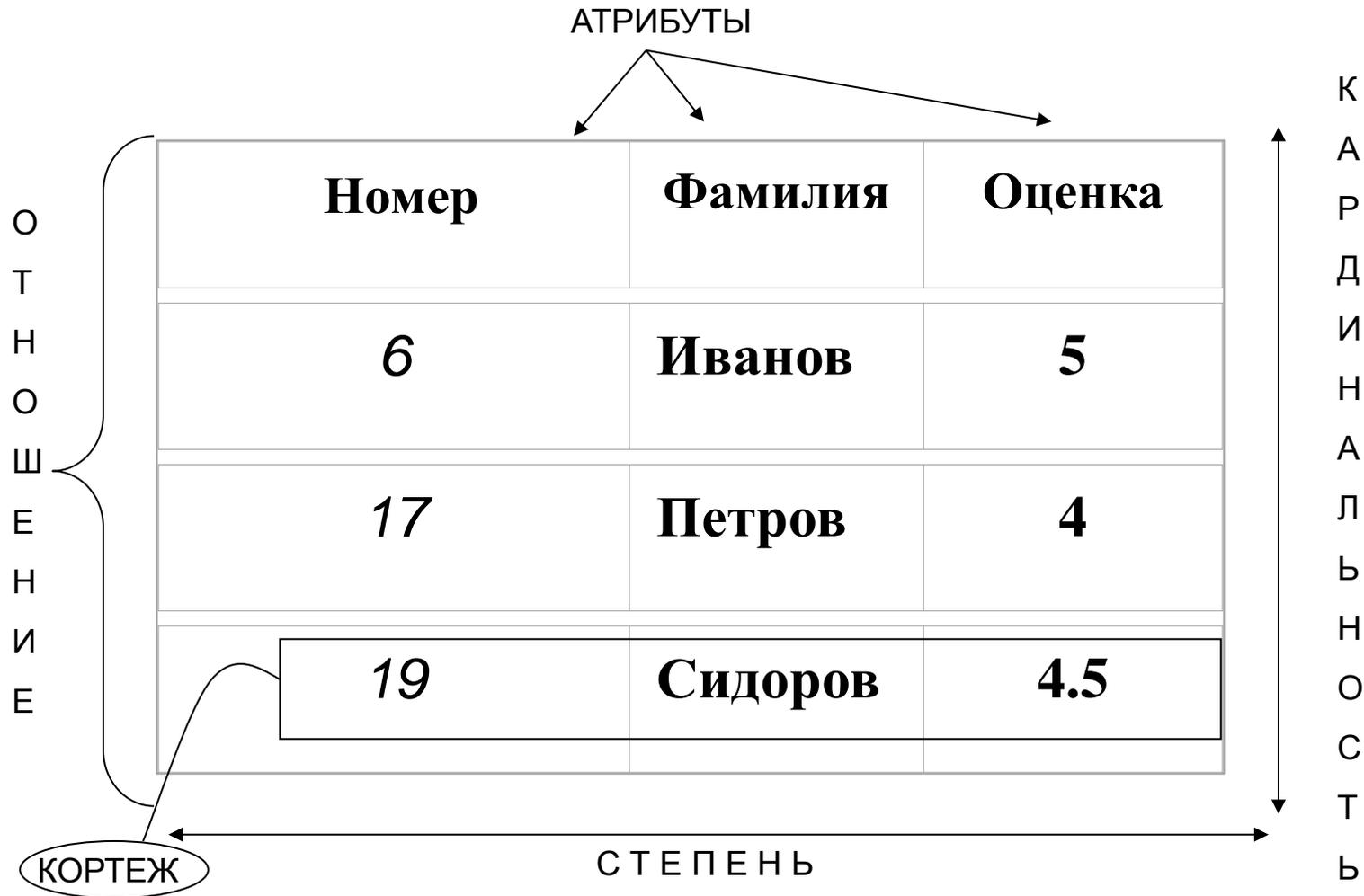
- Отношение
  - Атрибут
  - Домен
  - Кортеж
- ★ Кардинальность
  - ★ Степень отношения

Таблицы данных

Реляционная база данных

# Используемая терминология

## Структура реляционных данных



# Используемая терминология

**Отношение** - это плоская таблица (двумерная), состоящая из столбцов и строк.

**Атрибут** - это поименованный столбец отношения.

**Домен** - это набор допустимых значений для одного или нескольких атрибутов, благодаря ему пользователь может централизованно определять смысл и источник значений, которые могут получать атрибуты.

**Кортеж** - это строка отношения.

Кортежи называются **расширением, состоянием или телом отношения**, которое постоянно меняется.

Описание структуры отношения вместе со спецификацией доменов и любыми другими ограничениями возможных значений атрибутов иногда называют его **заголовком** (или **содержанием** (intension)).

---

# Используемая терминология

**Степень отношения** *определяется количеством атрибутов, которое оно содержит.*

**Кардинальность** - *это количество кортежей, которое содержит отношение.*

Кардинальность - свойство тела отношения (меняется при каждом добавлении или удалении кортежей ).

**Реляционная база данных** - *это набор нормализованных отношений.*

Реляционная база данных состоит из отношений, структура которых определяется с помощью особых методов, называемых *нормализацией*.

---

# Альтернативная терминология

<b>Официальные термины</b>	<b>Альтернативный вариант 1</b>	<b>Альтернативный вариант 2</b>
<b>Отношение</b>	<b>Таблица</b>	<b>Файл</b>
<b>Кортеж</b>	<b>Строка</b>	<b>Запись</b>
<b>Атрибут</b>	<b>Столбец</b>	<b>Поле</b>

# Отношения и их свойства в базе данных

**Реляционная схема** – это имя отношения, за которым следует множество пар атрибутов и доменов.

атрибуты  $A_1, A_2, \dots, A_n$   
домены  $D_1, D_2, \dots, D_n$   $\Rightarrow$  реляционная схема:  
 $\{A_1:D_1 \dots A_n:D_n\}$

В реляционной модели отношение можно представить как произвольное подмножество декартового произведения, а таблица – это физическое представление такого отношения.

# Отношения и их свойства в базе данных

## Свойства отношений:

- Отношение имеет неповторимое *имя*.
- Каждая ячейка отношения содержит только *атомарное (неделимое) значение*.
- Каждый атрибут имеет *уникальное имя*.
- Значения атрибута берутся из одного и того же домена.
- Порядок следования атрибутов не имеет никакого значения.
- Каждый кортеж является *уникальным*, т.е. дубликатов кортежей быть не может.
- Порядок следования кортежей в отношении не имеет никакого значения.

# Реляционные ключи

**Суперключ** (*superkey*) – атрибут или множество атрибутов, которое единственным образом идентифицирует кортеж данного отношения.

**Потенциальный ключ** – это суперключ, который не содержит подмножества, также являющегося суперключом данного отношения.

Потенциальный ключ  $K$  для данного отношения  $R$  обладает двумя свойствами:

- **Уникальность.** В каждом кортеже отношения  $R$  значение ключа  $K$  единственным образом идентифицируют этот кортеж.
- **Неприводимость.** Никакое допустимое подмножество ключа  $K$  не обладает свойством уникальности.

---

# Реляционные ключи

Наличие значений-дубликатов в конкретном существующем наборе кортежей доказывает то, что некоторая комбинация атрибутов не может быть потенциальным ключом.

*Если ключ состоит из нескольких атрибутов, то он называется **составным ключом**.*

**Первичный ключ** – это потенциальный ключ, который выбран для уникальной идентификации кортежей внутри отношения.

Поскольку отношение не содержит кортежей-дубликатов, всегда можно уникальным образом идентифицировать каждую его строку.

*Это значит, что отношение всегда имеет первичный ключ.*

---

---

# Реляционные ключи

*Потенциальные ключи, которые не выбраны в качестве первичного ключа, называются*  
**альтернативными ключами.**

**Внешний ключ** – это атрибут или множество атрибутов внутри отношения, которое соответствует потенциальному ключу некоторого (может быть, того же самого) отношения.

---

---

# Представление схем в реляционной базе данных

Реляционная база данных может состоять из произвольного количества отношений.

**Концептуальной моделью, или концептуальной схемой, называется множество всех реляционных баз данных.**

---

---

# Реляционная целостность

**Модель данных имеет две части:**

- *управляющую часть*, которая определяет типы допустимых операций с данными,
- *набор ограничений целостности*, которые гарантируют корректность данных.

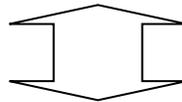
**Определитель NULL** вводится в связи с поддержанием правил целостности и указывает, что значение атрибута в настоящий момент неизвестно или неприемлемо для этого кортежа.

*Нули и пробелы* представляют собой некоторые значения, тогда как ключевое слово *NULL* призвано обозначать отсутствие какого-либо значения.

---

# Реляционная целостность

**Целостность сущностей** означает, что в отношении ни один атрибут первичного ключа не может содержать отсутствующих значений, обозначаемых определителем *NULL*.



Если будет определителя *NULL* в любой части первичного ключа, это утверждает, что не все его атрибуты необходимы для уникальной идентификации кортежей. Это противоречит определению первичного ключа.

---

# Реляционная целостность

## Ссылочная целостность.

*Если в отношении существует внешний ключ, то значение внешнего ключа должно либо соответствовать значению потенциального ключа некоторого кортежа в его базовом отношении, либо задаваться определителем NULL.*

## **Корпоративные ограничения целостности данных**

*это дополнительные правила поддержки целостности, определяемые пользователями или администраторами базы данных.*

---