



# БД И СУБД

Обобщение



**БАЗА ДАННЫХ**



**База данных – это организованная структура, предназначенная для хранения информации**

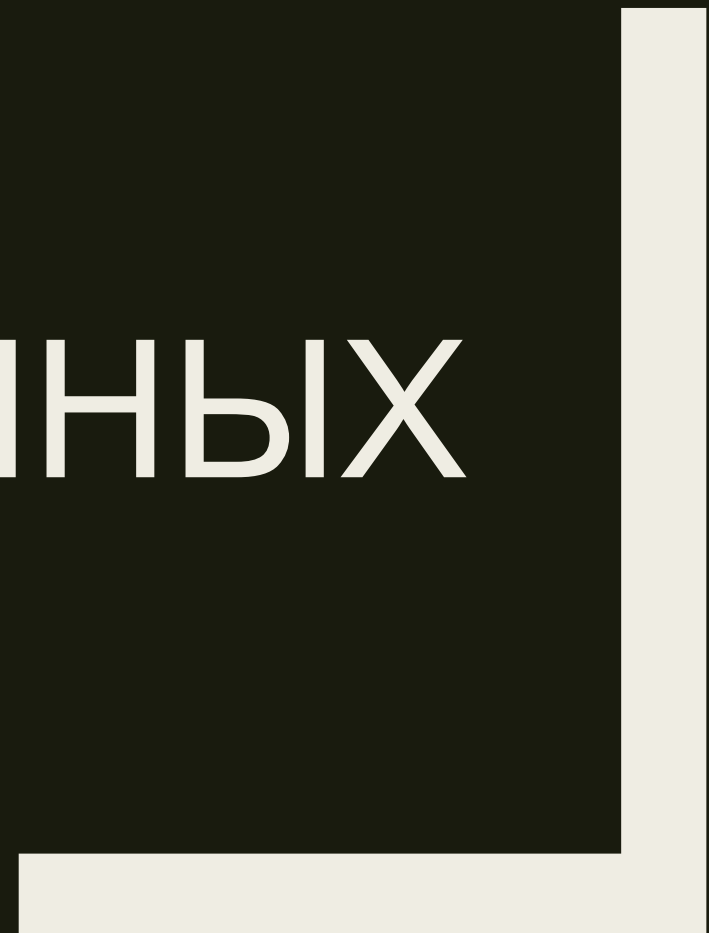


# Классификация БД

- По технологии обработки данных базы данных делятся на **централизованные** и **распределенные**.
- По способу доступа к данным базы данных делятся на **базы данных с локальным доступом** и **базы данных с удаленным (сетевым доступом)**.



# МОДЕЛИ ДАННЫХ



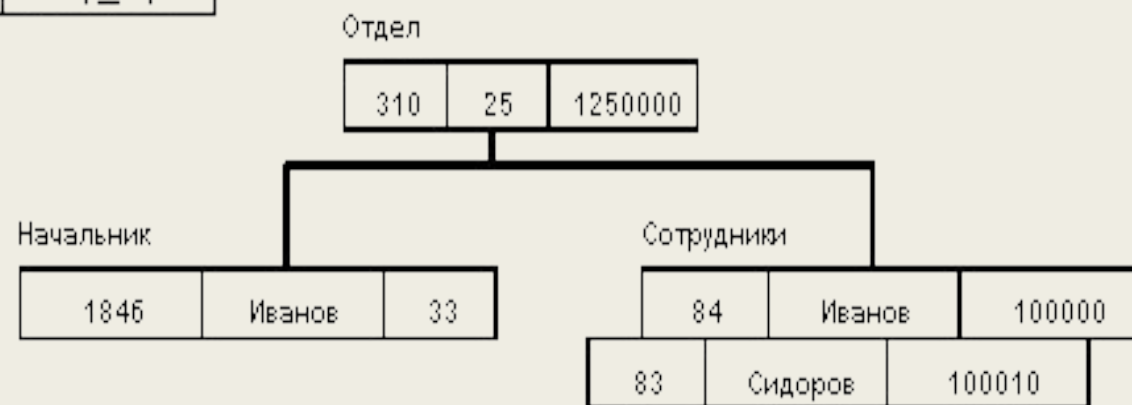
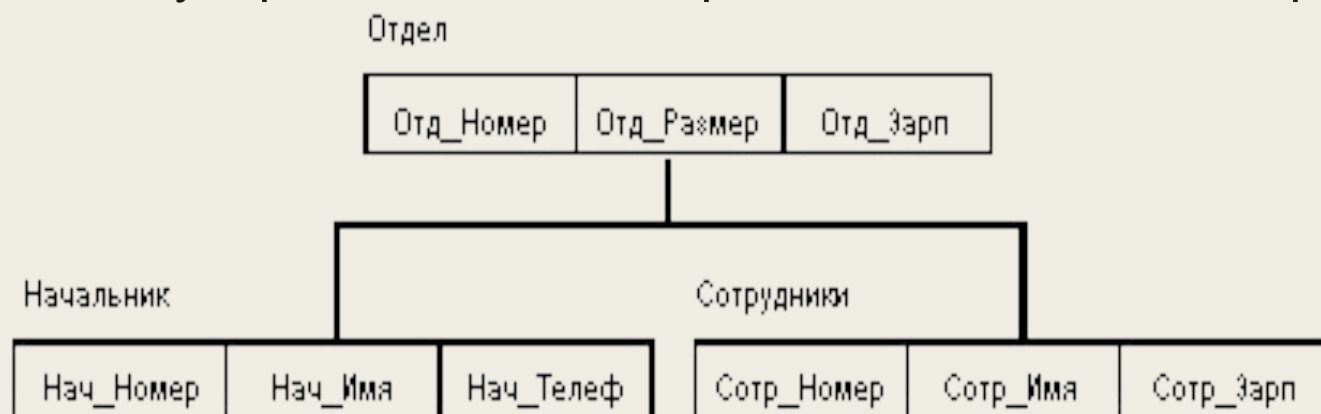
# Модель данных

математическое средство абстракции, позволяющее отделить факты от их интерпретации и вместе с тем обеспечить развитые возможности представления соотношения данных.



# Иерархические структуры данных

Иерархическая БД состоит из упорядоченного набора деревьев; из упорядоченного набора нескольких экземпляров одного типа дерева.



# Сетевые структуры данных

Сетевой подход к организации данных является расширением иерархического.

- *В иерархических структурах запись-потомок должна иметь в точности одного предка;*
- *в сетевой структуре данных потомок может иметь любое число предков.*



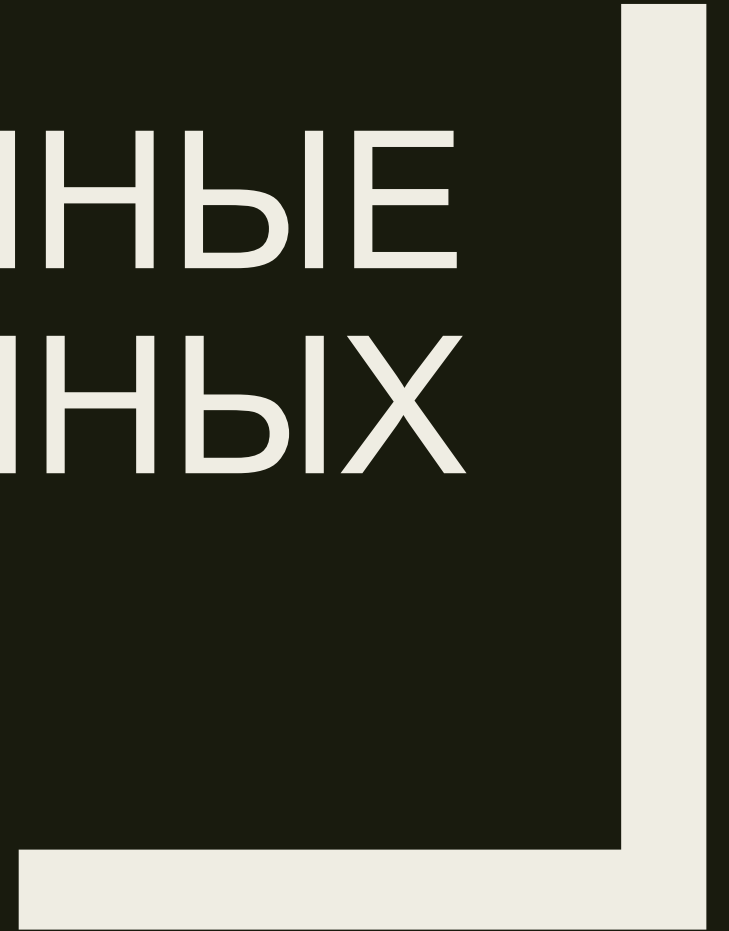


# Реляционная модель данных

является одним из самых простых видов представления информации и самой легкой концепцией для понимания. *Эта модель представляет собой данные упорядоченные в таблицы, чаще всего двумерные.* Таблицы такого вида называются отношениями (реляциями) и каждая таблица представляет собой отдельный файл.

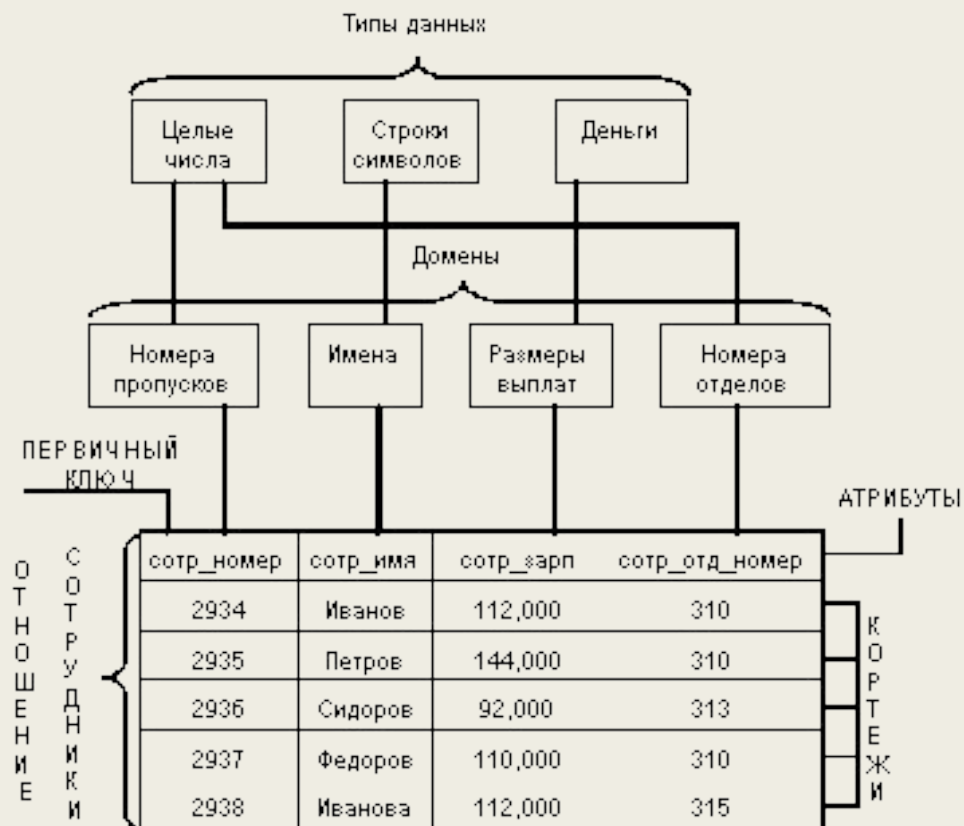
НОМЕР – КЛИЕНТА	ФАМИЛИЯ – КЛИЕНТА	ИМЯ - КЛИЕНТА	ОТЧЕСТВО - КЛИЕНТА

# РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ



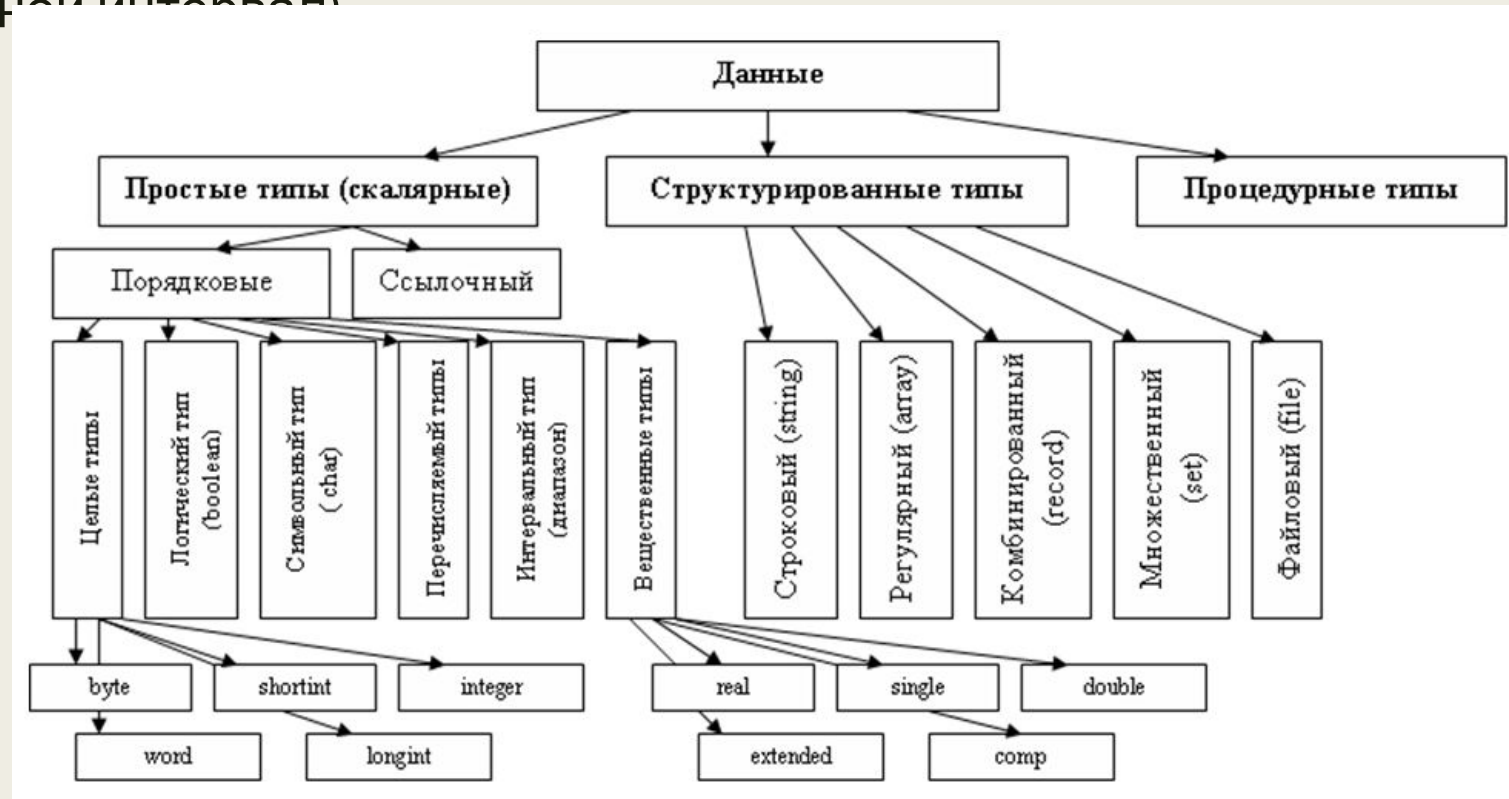
# Базовые понятия реляционных баз данных

Основными понятиями реляционных баз данных являются тип данных, домен, атрибут, кортеж, первичный ключ и отношение.



# Типы данных

Обычно в современных реляционных БД допускается хранение символьных, числовых данных, битовых строк, специализированных числовых данных (таких как "деньги"), а также специальных "темпоральных" данных (дата, время, временной интервал)



# Домен

- *В самом общем виде домен определяется заданием некоторого базового типа данных, к которому относятся элементы домена, и произвольного логического выражения, применяемого к элементу типа данных.*
- Допустимое потенциальное множество значений данного типа.

Например, домен "Имена" в нашем примере определен на базовом типе строк символов, но в число его значений могут входить только те строки, которые могут изображать имя (в частности, такие строки не могут начинаться с мягкого знака).



# Схема отношения.

Отношение СОТРУДНИК (таблица)      Атрибут Отдел (заголовок столбца)      Схема отношения (строка заголовков)

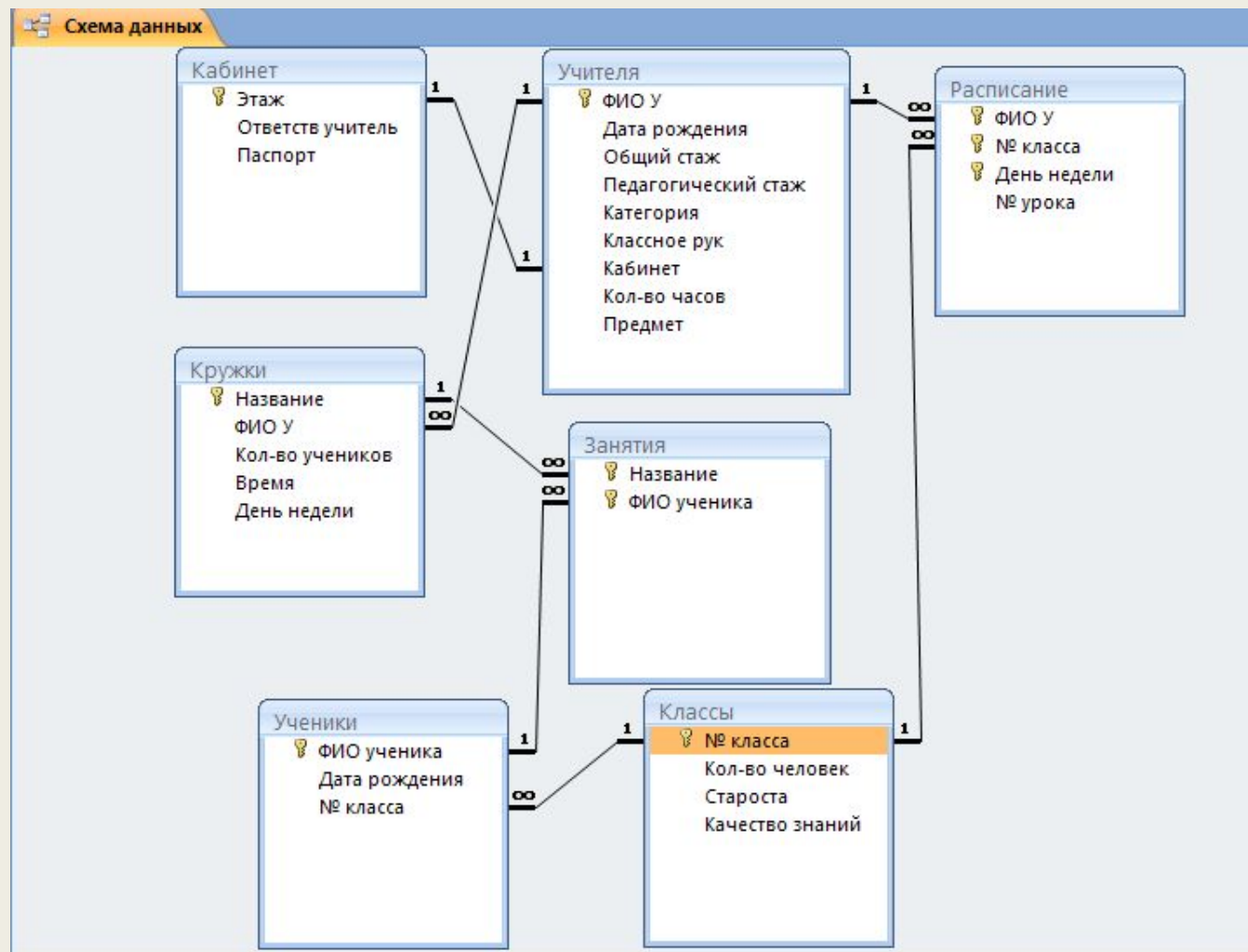
ФИО	Отдел	Должность	Д_рождения
Иванов И.И.	002	Начальник	27.09.51
Петров П.П.	001	Заместитель	15.04.55
Сидоров И.П.	002	Инженер	13.01.70

Кортеж (строка)

Значение атрибута (значение поля в записи)

The diagram illustrates the components of a relation schema. It features a table with four columns: 'ФИО', 'Отдел', 'Должность', and 'Д\_рождения'. The first row is the header, with labels 'Атрибут Отдел (заголовок столбца)' pointing to the 'Отдел' header and 'Схема отношения (строка заголовков)' pointing to the entire header row. The subsequent rows represent data records. A label 'Кортеж (строка)' points to the second row (Ivanov). A label 'Значение атрибута (значение поля в записи)' points to the 'Инженер' value in the third row. A dashed box encloses the second and third rows, representing a subset of the relation.

# Схема базы данных



# Кортеж

- **Кортеж**, соответствующий данной схеме отношения, - это множество пар {имя атрибута, значение}, которое содержит одно вхождение каждого имени атрибута, принадлежащего схеме отношения.
- **набор именованных значений заданного типа.**

сотр_номер	сотр_имя	сотр_зарп	сотр_отд_номер
2934	Иванов	112,000	310
2935	Петров	144,000	310
2936	Сидоров	92,000	313
2937	Федоров	110,000	310
2938	Иванова	112,000	315



# Отношение

- множество кортежей, соответствующих одной схеме отношения.

**схему отношения называют заголовком отношения, а отношение как набор кортежей - телом отношения.**

сотр_номер	сотр_имя	сотр_зарп	сотр_отд_номер
2934	Иванов	112,000	310
2935	Петров	144,000	310
2936	Сидоров	92,000	313
2937	Федоров	110,000	310
2938	Иванова	112,000	315

# Атрибут

сотр_номер	сотр_имя	сотр_зарп	сотр_отд_номер
2934	Иванов	112,000	310
2935	Петров	144,000	310
2936	Сидоров	92,000	313
2937	Федоров	110,000	310
2938	Иванова	112,000	315

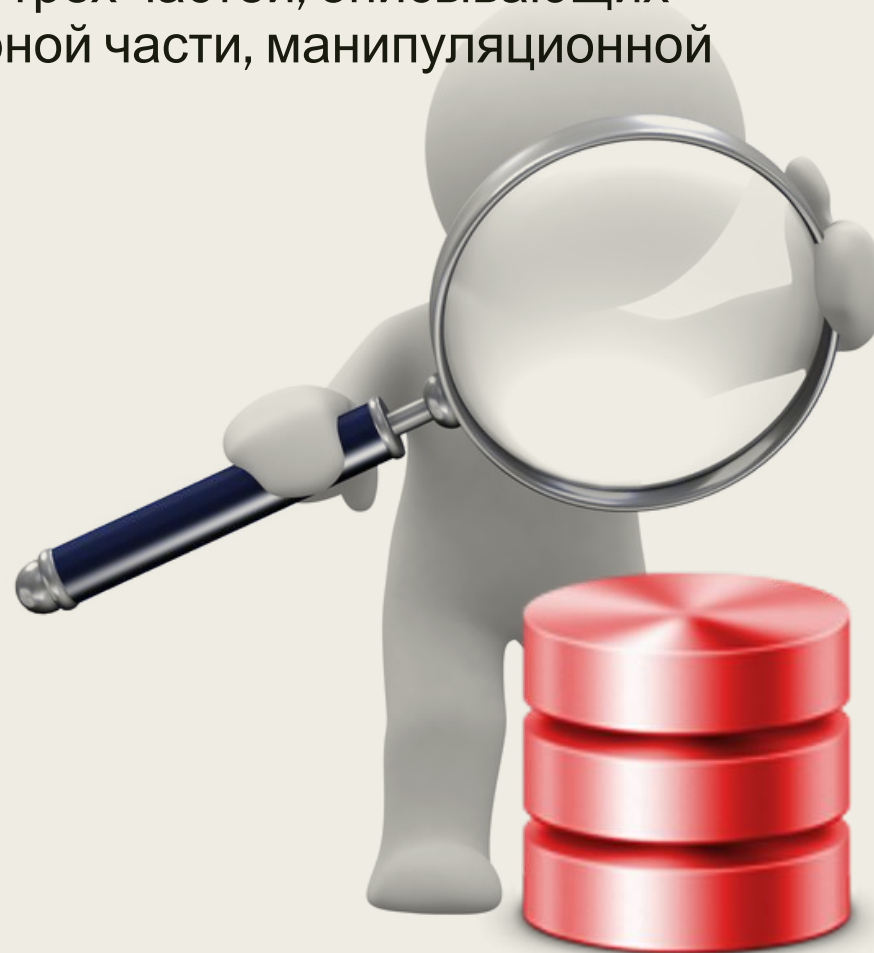
**Отношения является таблица,  
заголовком которой является схема отношения, а  
строками - кортежи отношения-экземпляра;**

**в этом случае имена атрибутов именуют  
столбцы этой таблицы.**

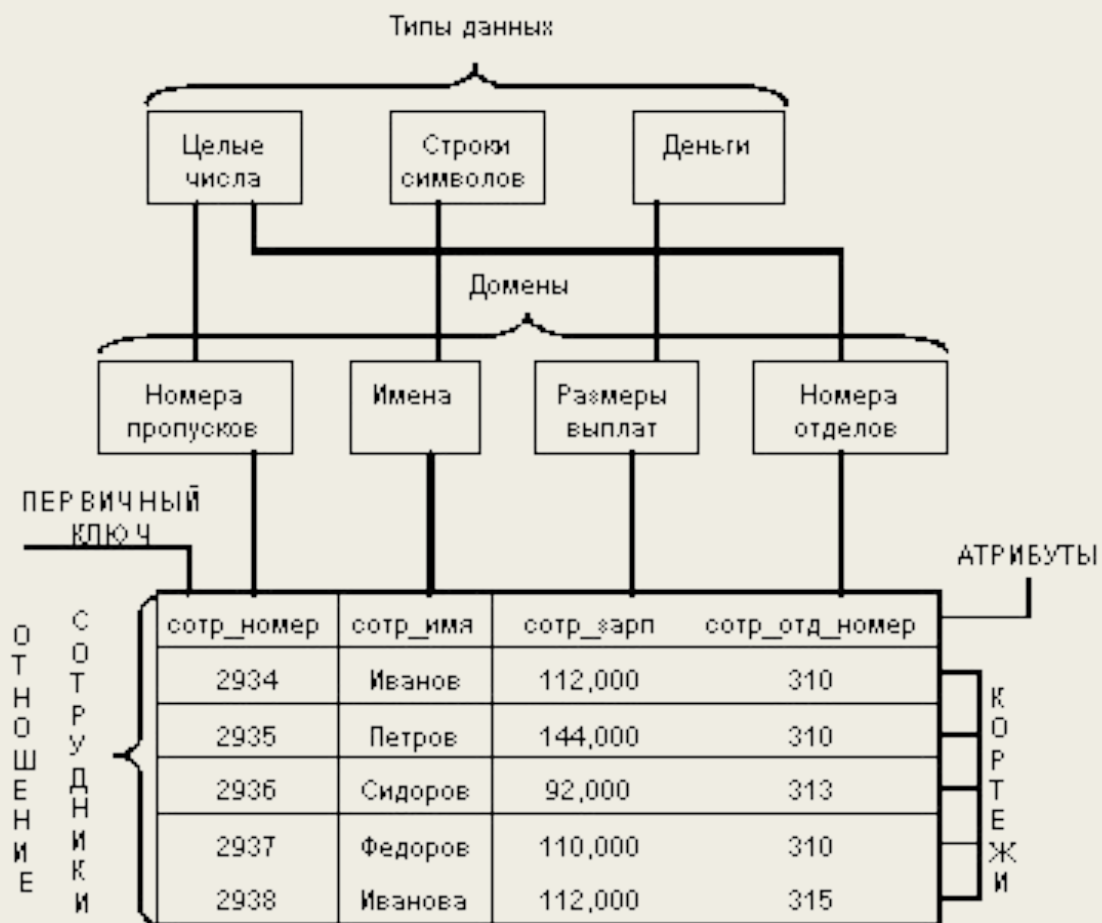
Поэтому иногда говорят "столбец таблицы", имея в виду "атрибут отношения".

# Общая характеристика

Согласно Дейту реляционная модель состоит из трех частей, описывающих разные аспекты реляционного подхода: структурной части, манипуляционной части и целостной части.



# Структурная часть



# Манипуляционная часть

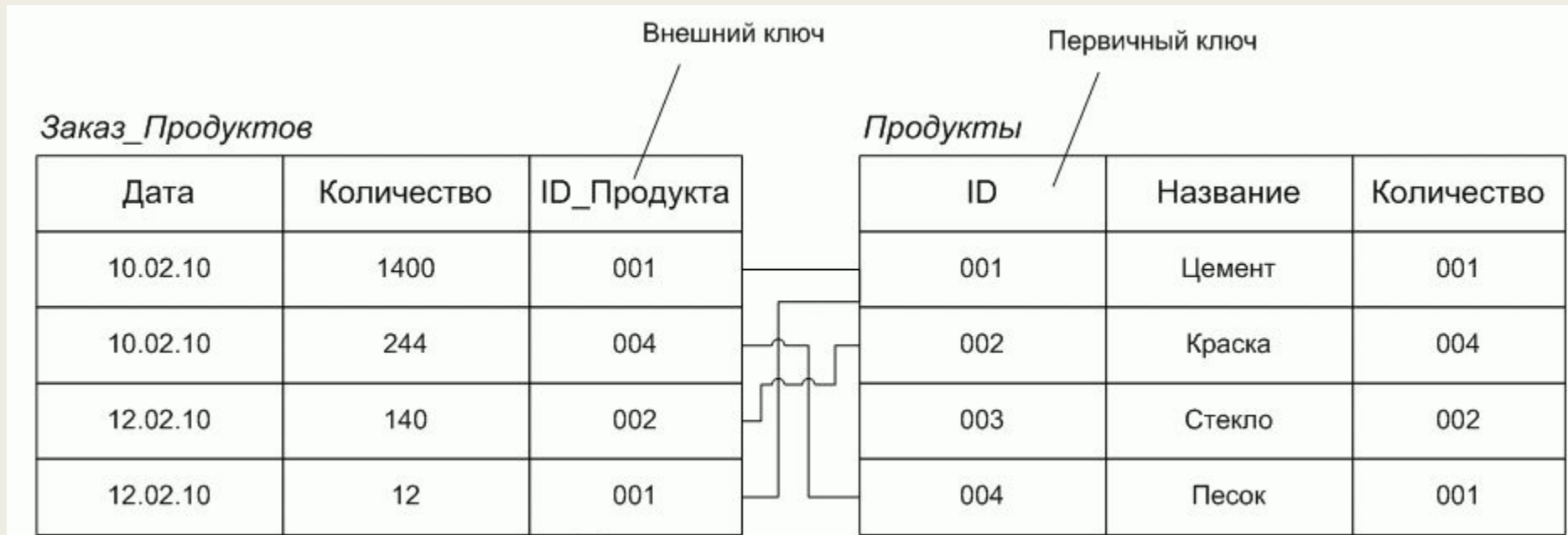
Здесь утверждаются два фундаментальных механизма манипулирования реляционными БД - **реляционная алгебра** и **реляционное исчисление**.

Первый механизм базируется в основном на классической теории множеств (с некоторыми уточнениями), а второй - на классическом логическом аппарате исчисления предикатов первого порядка.

# Целостная часть

Здесь фиксируются два базовых требования целостности :

- требованием целостности сущностей.
- требованием целостности по ссылкам .





# СУБД

Система управления базами данных



**Система управления базами данных (СУБД)** – это программное обеспечение, с помощью которого пользователи могут определять, создавать и поддерживать базу данных, а также осуществлять к ней контролируемый доступ

ORACLE

MySQL®

phpMyAdmin



PostgreSQL

Microsoft®  
SQL Server®



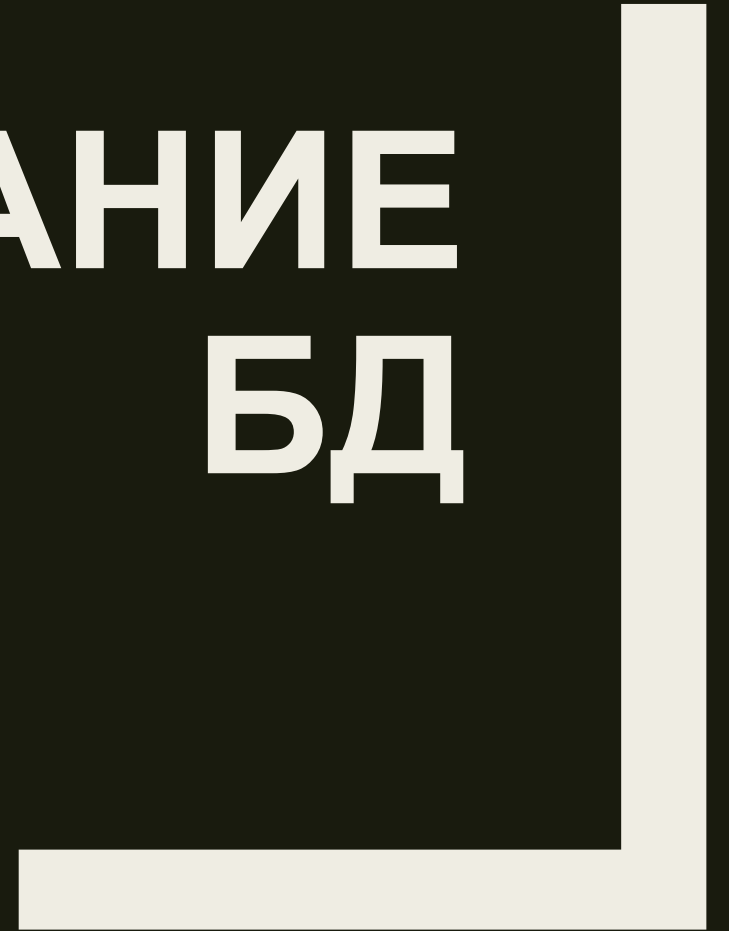
# Современные СУБД имеют следующие возможности:

- включают язык определения данных, с помощью которого можно определить базу данных, ее структуру, типы данных, а также средства задания ограничения для хранимой информации;
- позволяют вставлять, удалять, обновлять и извлекать информацию из базы данных посредством языка запросов (SQL);
- большинство СУБД могут работать на компьютерах с разной архитектурой и под разными операционными системами;
- многопользовательские СУБД имеют развитые средства администрирования баз данных.

# Для работы с базой данных СУБД должна обеспечивать:

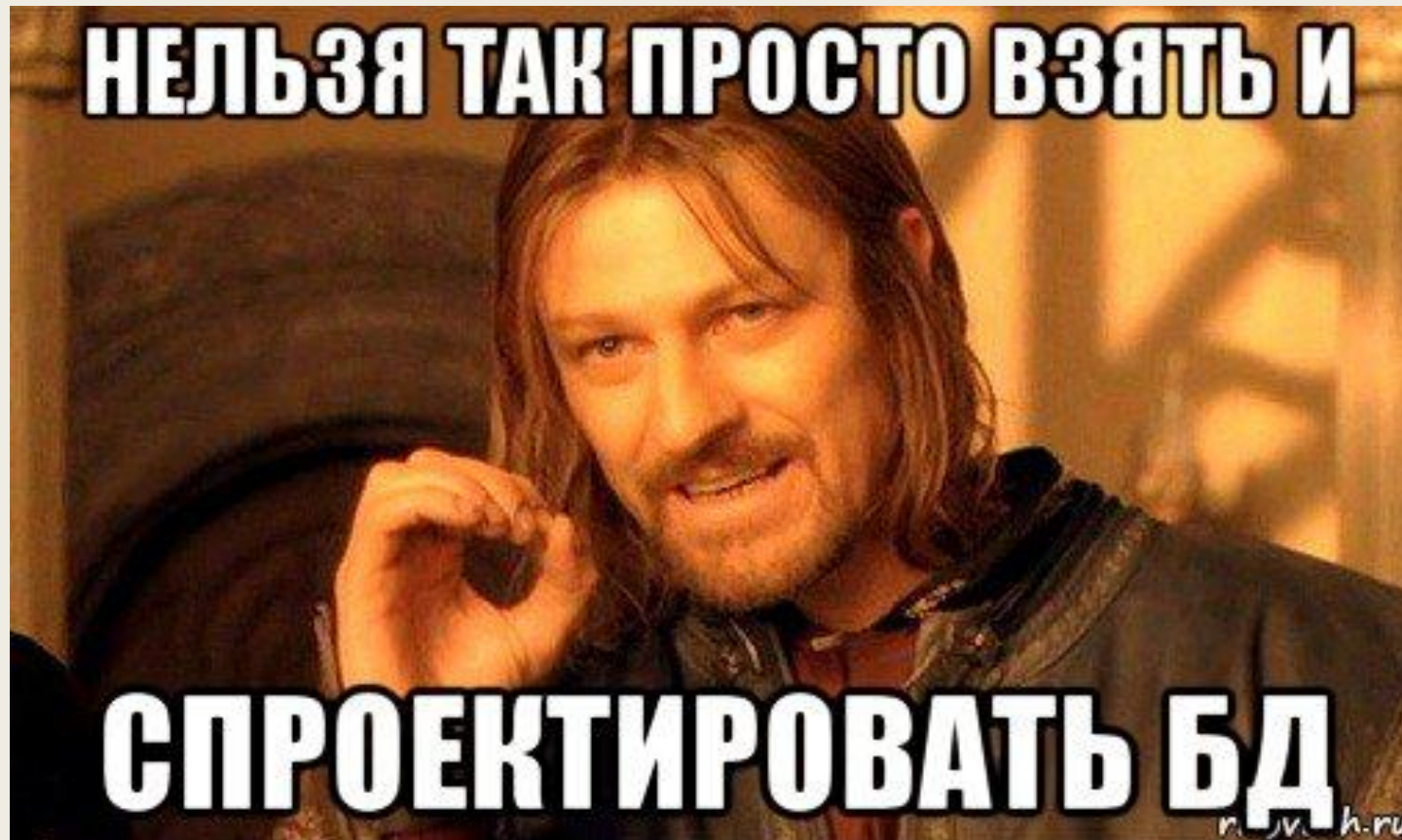
- возможность внесения и чтения информации;
- работу с большим объемом данных;
- быстроту поиска данных;
- целостность данных (их непротиворечивость);
- защиту от разрушения, уничтожения (не только при случайных ошибках пользователя), от несанкционированного доступа;
- систему дружественных подсказок (в расчете на пользователя без специальной подготовки).

# ПРОЕКТИРОВАНИЕ БД



# Этапы проектирования БД

Концептуальный, логический,  
физический



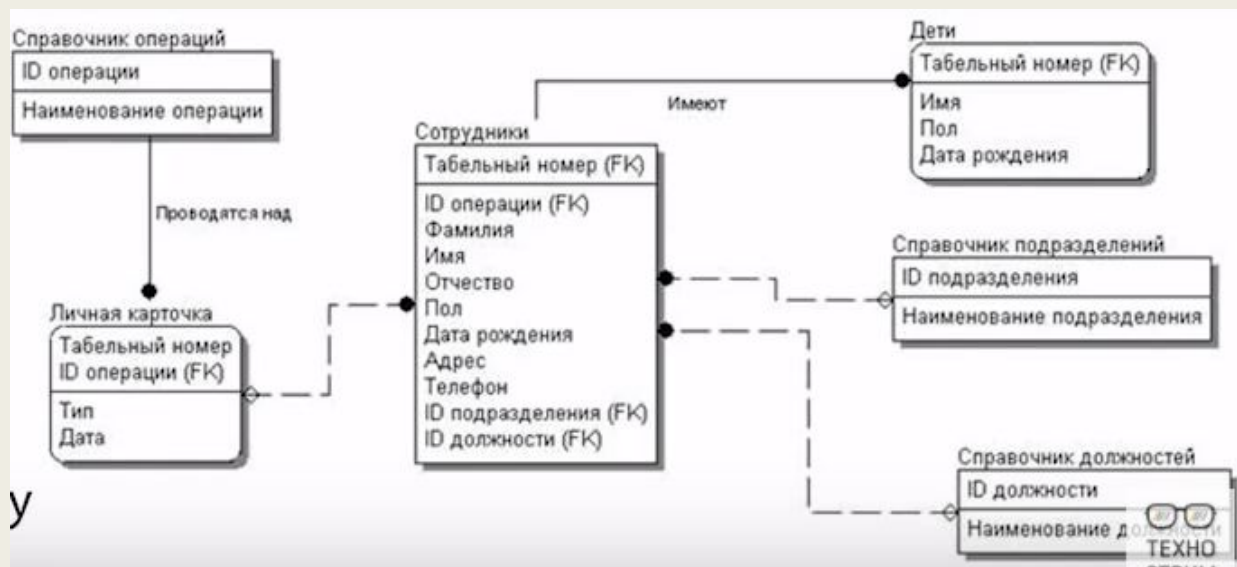
# Концептуальный этап

Системный анализ и словесное описание информационных объектов ПО



# Логический этап (инфологический)

Задача инфологического этапа проектирования: получение семантических (смысловых) моделей данных (например, в терминах ER-моделей), отображающих информационное содержание конкретной ПО. Вначале выполняется выделение из воспринимаемой реальности требуемой части ПО, определяются ее границы, происходит абстрагирование от несущественных частей для конкретного применения БД. В результате определяются объекты, их свойства и связи, которые будут существенны для будущих пользователей системы.



# Физический этап

Задачей физического этапа проектирования является выбор рациональной структуры хранения данных. и методов доступа к ним, исходя из того арсенала средств и методов, который предоставляет разработчику конкретная СУБД.



# Нормализация

Процесс проектирования представляет собой **процесс нормализации** схем отношений, причем каждая следующая нормальная форма обладает свойствами лучшими, чем предыдущая.

В теории реляционных баз данных обычно выделяется следующая последовательность нормальных форм:

- первая нормальная форма (1NF);
- вторая нормальная форма (2NF);
- третья нормальная форма (3NF);
- нормальная форма Бойса-Кодда (BCNF);
- четвертая нормальная форма (4NF);
- пятая нормальная форма (5NF или PJ/NF).

# Основные свойства нормальных форм:

- каждая следующая нормальная форма в некотором смысле лучше предыдущей;
- при переходе к следующей нормальной форме свойства предыдущих нормальных свойств сохраняются.

ТРАНЗАКЦИИ



# Транзакция - это неделимая, с точки зрения воздействия на СУБД, последовательность операций манипулирования данными.

Для пользователя транзакция выполняется по принципу "***все или ничего***", т.е. либо транзакция выполняется целиком и переводит базу данных из одного ***целостного состояния*** в другое ***целостное состояние***, либо, если по каким-либо причинам, одно из действий транзакции невыполнимо, или произошло какое-либо нарушение работы системы, база данных возвращается в исходное состояние, которое было до начала транзакции (происходит откат транзакции).

**В однопользовательских системах транзакции** - это логические единицы работы, после выполнения которых база данных остается *в целостном состоянии*

# Свойства транзакций

- **(А) Атомарность.** Транзакция выполняется как атомарная операция - либо выполняется вся транзакция целиком, либо она целиком не выполняется.
- **(С) Согласованность.** Транзакция переводит базу данных из одного согласованного (целостного) состояния в другое согласованное (целостное) состояние. Внутри транзакции согласованность базы данных может нарушаться.
- **(И) Изоляция.** Транзакции разных пользователей не должны мешать друг другу (например, как если бы они выполнялись строго по очереди).
- **(Д) Долговечность.** Если транзакция выполнена, то результаты ее работы должны сохраниться в базе данных, даже если в следующий момент произойдет сбой системы.