



СЫКТЫВКАРСКИЙ
ЛЕСНОЙ
ИНСТИТУТ

БАЗЫ ДАННЫХ

ПРЕПОДАВАТЕЛЬ:

КЛОЧЕВА ЕВГЕНИЯ АЛЕКСАНДРОВНА

Определения

База данных (БД) – это хранилище данных о некоторой предметной области, организованное в виде специальной структуры.

Важно:

- данные о некоторой области (не обо всем)
- упорядоченные

Система управления базой данных (СУБД) – это программное обеспечение для работы с БД.

Функции:

- поиск информации в БД
- выполнение несложных расчетов
- вывод отчетов на печать
- редактирование БД



Информационная система = БД + СУБД!

- **локальные ИС**

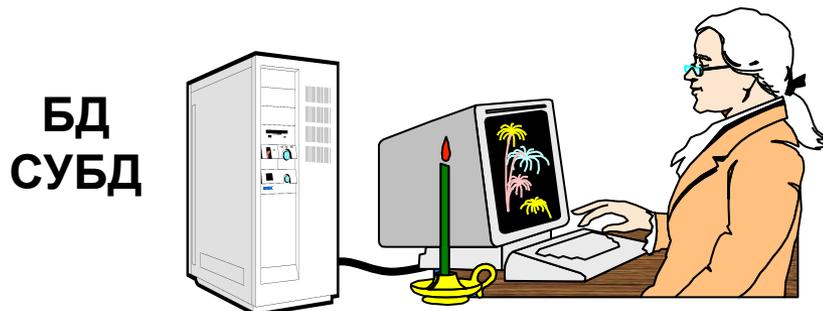
БД и СУБД находятся на одном компьютере.

- **файл-серверные**

БД находится на сервере сети (файловом сервере), а СУБД на компьютере пользователя.

- **клиент-серверные**

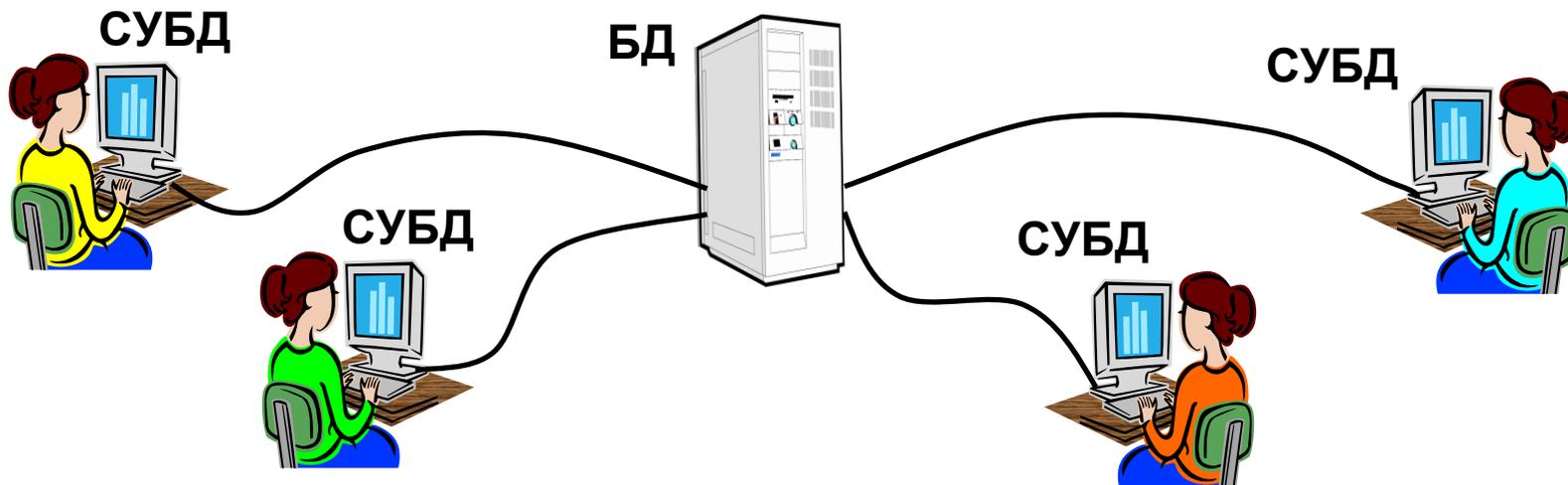
БД и основная СУБД находятся на сервере, СУБД на рабочей станции посылает запрос и выводит на экран результат.



- автономность
(независимость)



- с БД работает только один человек
- сложно обновлять при большом количестве пользователей
- практически невозможно «стыковать» изменения, вносимые несколькими пользователями



Примеры: Microsoft Access, Borland Paradox.

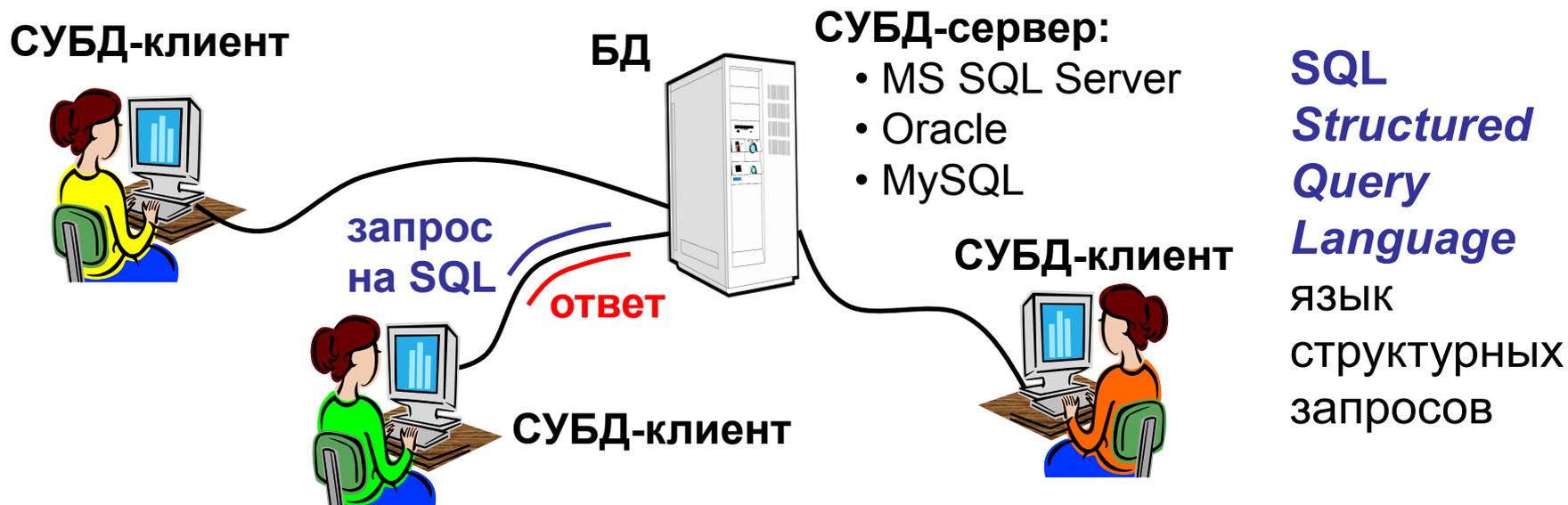


- несколько человек работают с одной базой



- основную работу выполняют рабочие станции (PC), они должны быть мощными
- для поиска строки на PC копируется вся БД – нагрузка на сеть
- слабая защита от взлома (только на PC)
- проблемы при одновременном изменении с разных PC

Клиент-серверные ИС



Примеры: Firebird, Interbase, IBM DB2, MS SQL Server, Oracle, PostgreSQL, MySQL, ЛИНТЕР.



- основную работу выполняет сервер
- проще модернизация (только сервер)
- по сети идут только нужные данные
- защита на сервере (сложнее взломать)
- разделение доступа (очередь заданий)



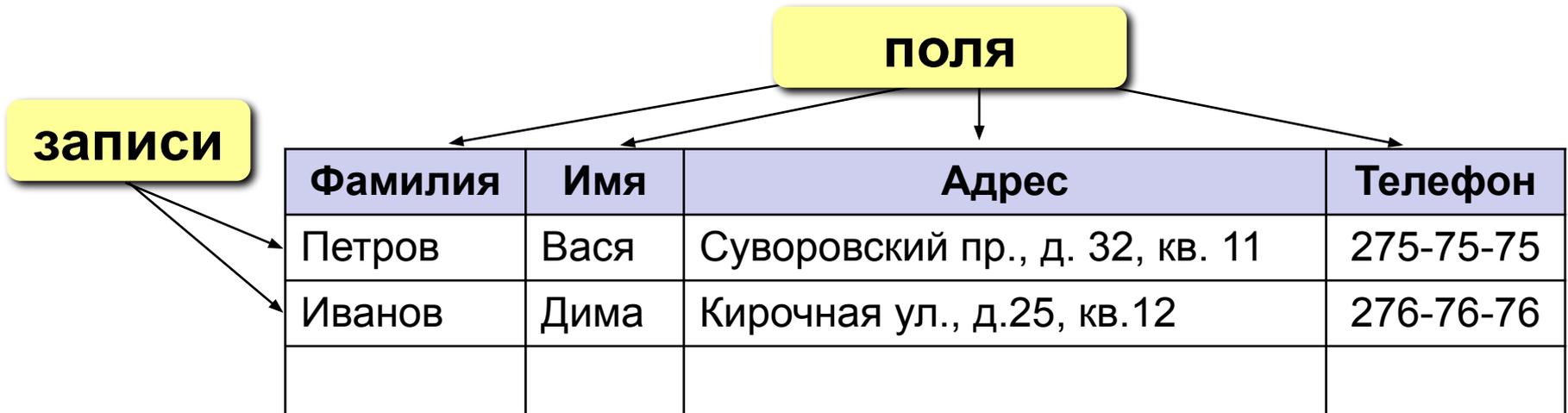
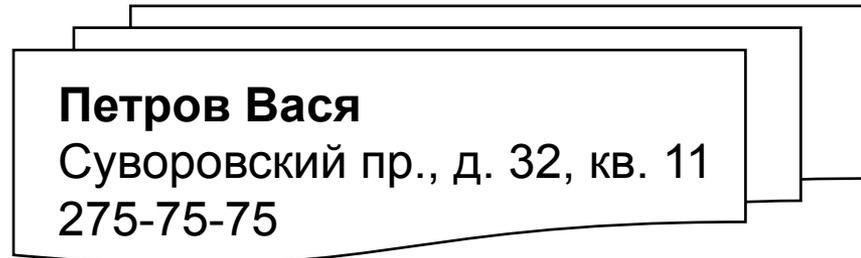
- сложность настройки
- высокая стоимость ПО (тысячи \$)

- **табличные БД**
данные в виде одной таблицы
- **сетевые БД**
набор узлов, в котором каждый может быть связан с каждым.
- **иерархические БД**
в виде многоуровневой структуры
- **реляционные БД (99,9%)**
набор взаимосвязанных таблиц

Модель – картотека

Примеры:

- записная книжка
- каталог в библиотеке



- 1) самая простая структура
- 2) все другие типы БД используют таблицы

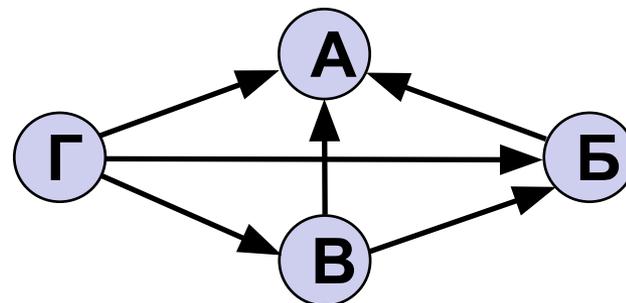
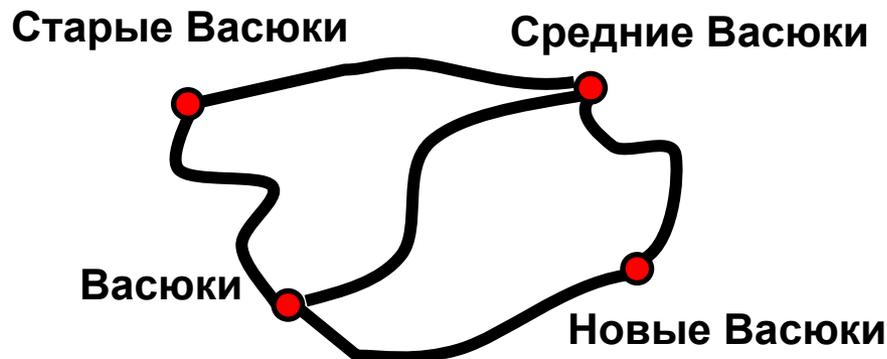


во многих случаях – дублирование данных:

А.С. Пушкин	Сказка о царе Салтане	20 стр.
А.С. Пушкин	Сказка о золотом петушке	12 стр.

- 1. Количество полей определяется разработчиком и не может изменяться пользователем.**
- 2. Любое поле должно иметь уникальное имя.**
- 3. Поля могут иметь различный тип:**
 - строка символов (длиной до 255 символов)
 - вещественное число (с дробной частью)
 - целое число
 - денежная сумма
 - дата, время, дата и время
 - логическое поле (истина или ложь, да или нет)
 - многострочный текст (МЕМО)
 - рисунок, звук или другой объект (объект OLE)
- 4. Поля могут быть обязательными для заполнения или нет.**
- 5. Таблица может содержать сколько угодно записей (это количество ограничено только объемом диска); записи можно добавлять, удалять, редактировать, сортировать, искать.**

Сетевая БД – это набор узлов, в которых каждый может быть связан с каждым (схема дорог).



- лучше всего отражает структуру некоторых задач (сетевое планирование в экономике)



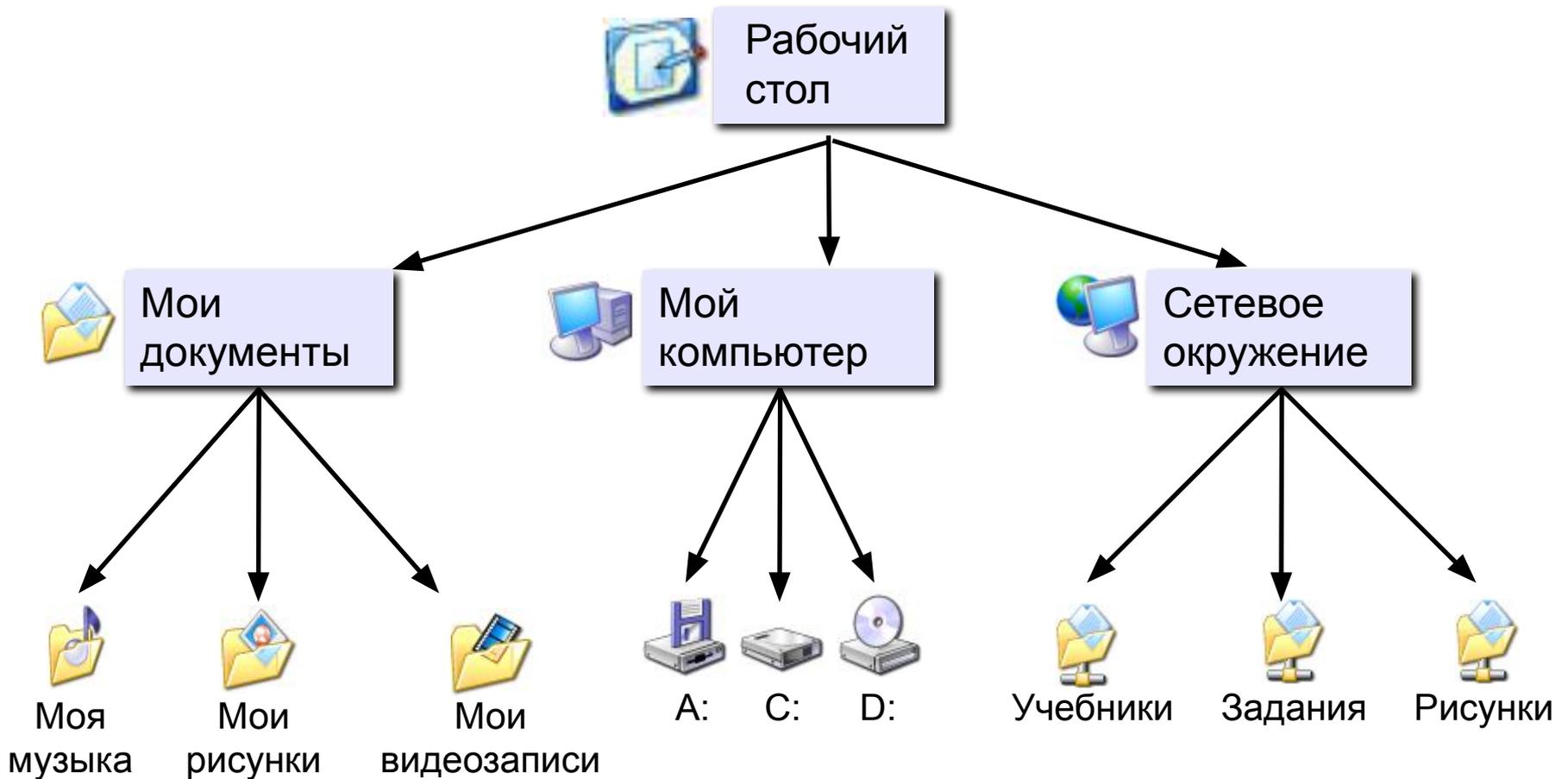
- сложно хранить информацию о всех связях
- запутанность структуры



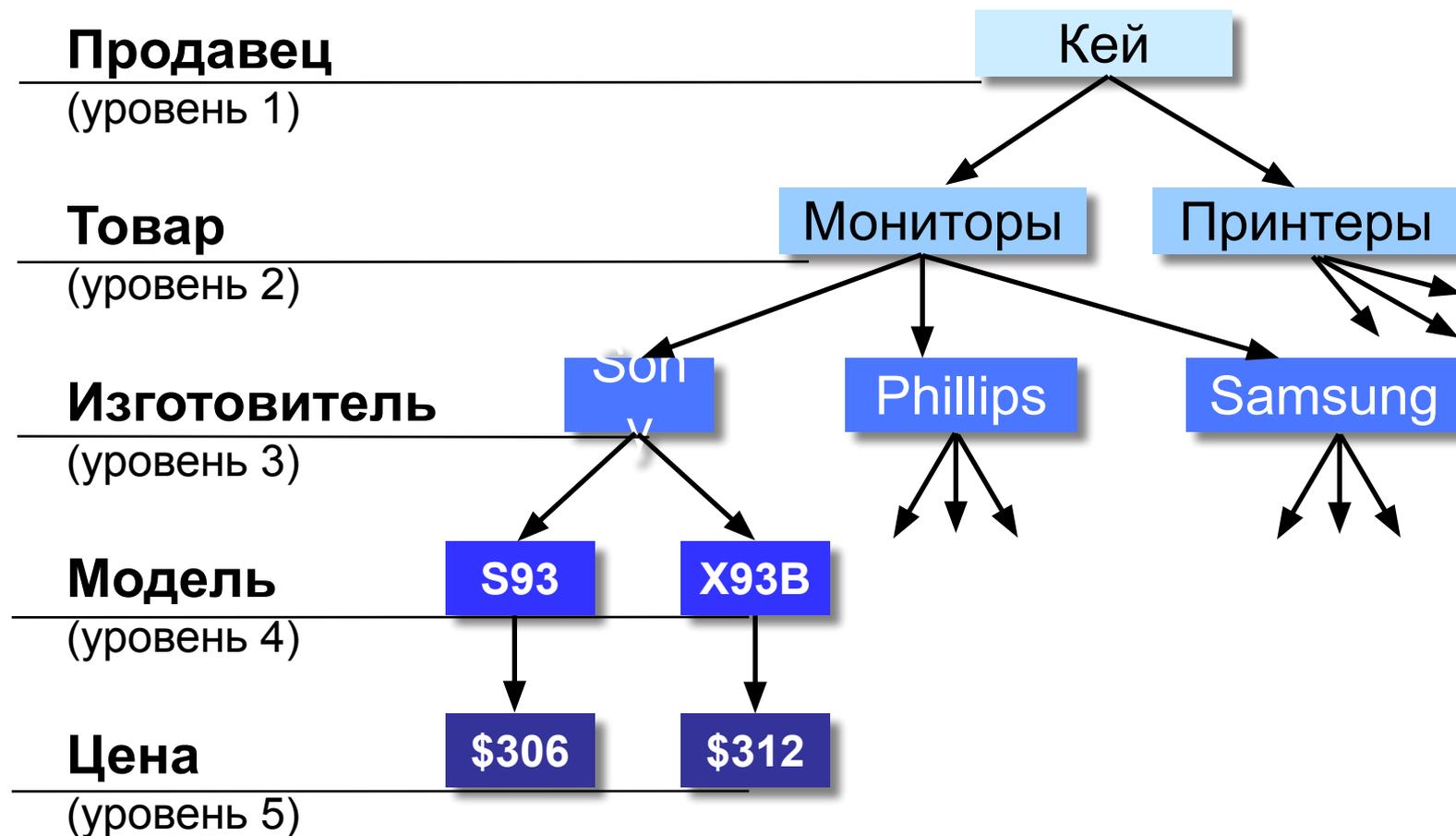
Можно хранить в виде таблицы, но с дублированием данных!

Иерархические БД

Иерархическая БД – это набор данных в виде многоуровневой структуры (дерева).



Прайс-лист:



Иерархические БД

Приведение к табличной форме:

Продавец	Товар	Изготовитель	Модель	Цена
Кей	Монитор	Sony	S93	\$306
Кей	Монитор	Sony	X93B	\$312
Key	Монитор	Phillips	190 B5 CG	\$318
Кей	Монитор	Samsung	SyncMaster 193P	\$452
...				

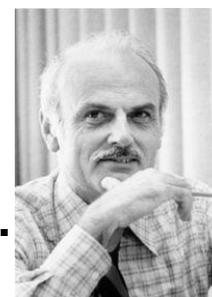
-  дублирование данных
- при изменении адреса фирмы надо менять его во всех строках
- нет защиты от ошибок ввода оператора (*Кей* – *Key*), лучше было бы выбирать из списка



СЫКТЫВКАРСКИЙ
ЛЕСНОЙ
ИНСТИТУТ

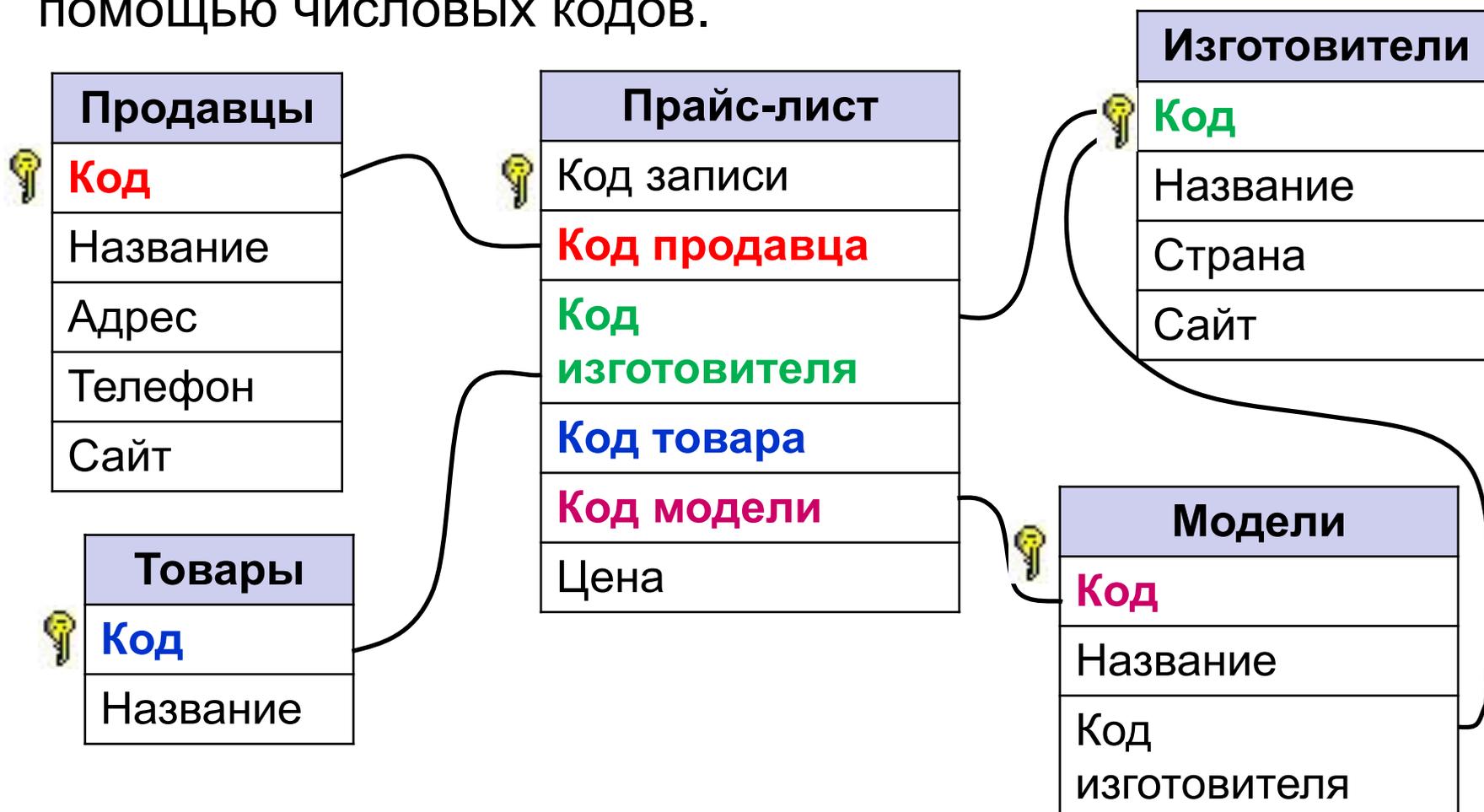
РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ

Реляционные БД



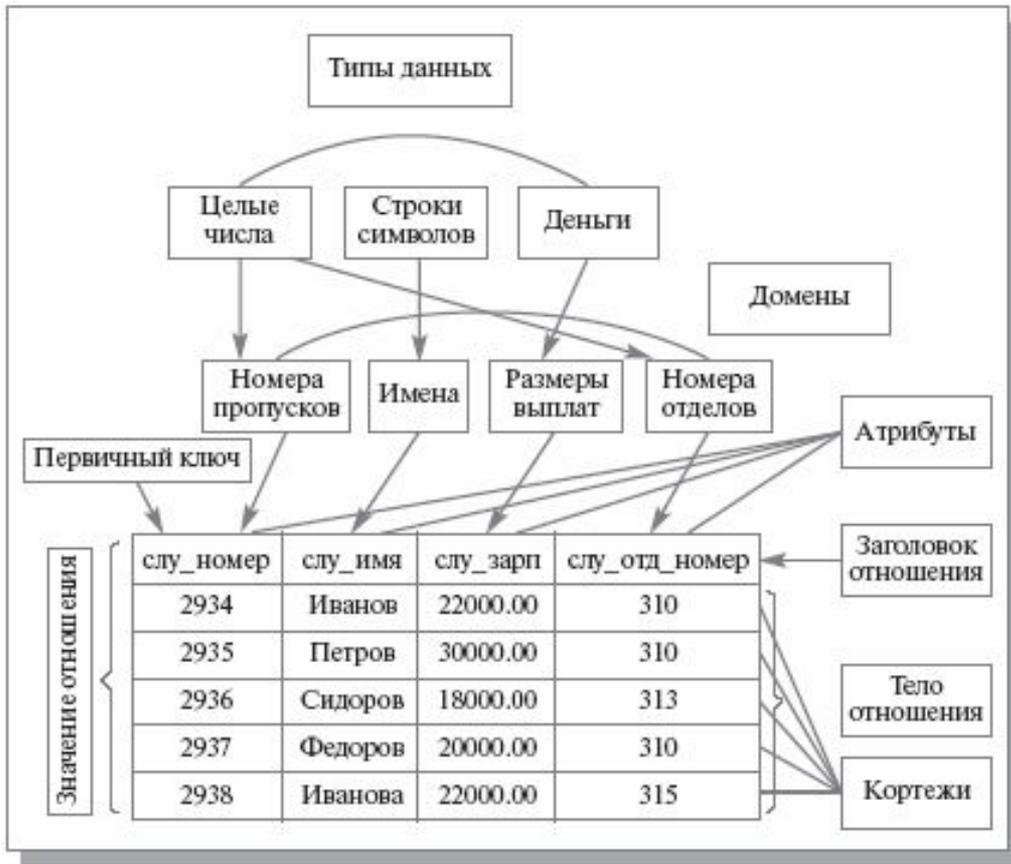
1970-е гг. Эдгар Кодд, англ. *relation* – отношение.

Реляционная база данных – это набор простых таблиц, между которыми установлены связи (отношения) с помощью числовых кодов.



Реляционные БД

-  нет дублирования информации;
 - при изменении адреса фирмы достаточно изменить его только в таблице **Продавцы**;
 - защита от неправильного ввода: можно выбрать только фирму, которая есть в таблице **Продавцы**;
 - механизм **транзакций**: любые изменения вносятся в базу только тогда, когда они полностью завершены.
-
-  сложность структуры (не более 40-50 таблиц);
 - при поиске надо обращаться к нескольким таблицам;
 - нужно поддерживать **целостность**: при удалении фирмы-продавца надо удалять все связанные записи (автоматически, **каскадное удаление**).



Сущность есть объект любой природы, данные о котором хранятся в базе данных.

Тип данных состоит из трех основных компонентов: определение множества значений данного типа; определение набора операций, применимых к значениям типа; определение способа внешнего представления значений типа.

Домен можно рассматривать как подмножество значений некоторого типа данных имеющих определенный смысл.

Атрибуты представляют собой свойства, характеризующие сущность. В структуре таблицы каждый атрибут именуется и ему соответствует заголовок некоторого столбца таблицы

Заголовок представляет собой множество *атрибутов* (именованных вхождений домена в заголовок отношения), а **тело** — множество **кортежей**, соответствующих заголовку.

Ключевое поле (ключ таблицы)

Ключ – минимальный набор атрибутов, по значениям которых можно однозначно найти требуемый экземпляр сущности.

Простой ключ					Составной ключ			
Номер	Автор	Название	Год	Полка	Город	№ школы	Директор	Телефон
001	Беляев А. Р.	Звезда КЭЦ	1990	3	Крюков	1	Иванов А. П.	12 - 35
002	Олеша Ю. К.	Избранное	1987	5	Шадринск	1	Строев С. С.	4 - 33 - 11
003	Беляев А. Р.	Избранное	1994	1	Шадринск	2	Иванов А. П.	4 - 23 - 15

Ключ состоящий из одного атрибута называют простым, а состоящий из нескольких атрибутов- составным (сложным).

Реляционные БД: связи между таблицами

19

Номер л.д.	ФИО	Группа	Специальность
1232	Алексеев А. А.	И-12	ИС
1233	Борисов Б. Б.	И-12	ИС
1234	Васькин В.В.	И-12	ИС
1235	Гришин Г. Г.	У-11	Сист. Управ.

Отношение СТУДЕНТ.

Назв.Предмета	Часы
Общая алгебра	102
ТФКП	85
ТАР	68
Теория Алгоритмов	85

Отношение ПРЕДМЕТ (Назв.Предмета., Часы).

Номер л.д.	ФИО	Назв.Предмета	Оценка
1232	Алексеев А. А.	Общая алгебра	Хор.
1233	Борисов Б. Б.	Общая алгебра	отл.
1234	Васькин В.В.	Общая алгебра	Удовл.
1232	Алексеев А. А.	ТФКП	Хор.

Отношение

СТУДЕНТ_ПРЕДМЕТ(номер личного дела , ФИО, Назв.Предмета , Оценка)

Связи между таблицами

Один к одному («1-1») – одной записи в первой таблице соответствует ровно одна запись во второй.

Применение: выделение часто используемых данных.

Код	Фамилия	Имя
1	Иванов	Кузьма
2	Петров	Василий
...		

Код	Год рожд.	Адрес
1	1992	Суворовский, д.20, кв. 6
2	1993	Кирочная, д. 30, кв 18
...		

Один ко многим («1-∞») – одной записи в первой таблице соответствует сколько угодно записей во второй.

Код	Название
1	Монитор
2	Винчестер
...	

Код	Код товара	Цена
123	1	10 999
345	1	11 999
...		

Связи между таблицами

Многие ко многим (« ∞ - ∞ ») – одной записи в первой таблице соответствует сколько угодно записей во второй, и наоборот.

учителя

Код	Фамилия
1	Иванов
2	Петров
...	

∞ ∞

предметы

Код	Название
1	История
2	География
3	Биология
...	

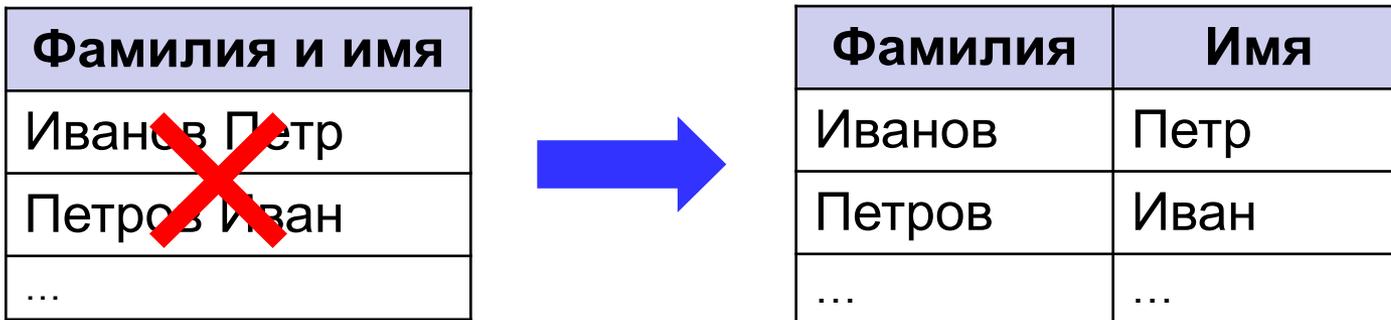
Реализация – через третью таблицу и две связи «1- ∞ ».



Нормализация базы данных

Нормализация – это разработка такой структуры БД, в которой нет избыточных данных и связей.

- Любое поле должно быть **неделимым**.



- Не должно быть полей, которые обозначают различные виды одного и того же, например, товаров.



Нормализация базы данных

- Любое поле должно зависеть только от ключа (**ключ** – это поле или комбинация полей, однозначно определяющая запись).

товары

Код	Название	Цена
1	Монитор	9 000 р.
2	Винчестер	11 000 р.
...		

зависит не только от названия товара!



прайс-лист

- Не должно быть полей, которые могут быть найдены с помощью остальных.

Код	Товар	Цена за тонну	Кол-во, тонн	Стоимость
1	Бананы	1200	10	12 000
2	Киви	1500	20	30 000
...				

Фундаментальные свойства отношений

Свойство 1: Отсутствие кортежей-дубликатов.

Свойство 2: Отсутствие упорядоченности кортежей (сверху вниз).

Свойство 3: Отсутствие упорядоченности атрибутов (слева направо)

Свойство 4: Атомарность значений атрибутов, первая нормальная форма отношения.