

# Лекция №2

**База данных (БД)** – динамическая информационная модель своей предметной области, адекватно отражающая ее состояние в любой момент времени.

Динамическое изменение информации на примере студентов:



## *Студенты*

<b>Фамилия</b>	<b>Имя</b>	<b>Отчество</b>	<b>Группа</b>
Петров	Иван	Васильевич	529-1
Сидорова	Мария	Анатольевна	529-2
Иванов	Иван	Иванович	579

## *Студенты*

<b>Фамилия</b>	<b>Имя</b>	<b>Отчество</b>	<b>Группа</b>
Петров	Иван	Васильевич	529-1
Петрова			
<del>Сидорова</del>	Мария	Анатольевна	529-2
Иванов	Иван	Иванович	579

## Свойства БД:

1. *Доступность* - возможность работы со всеми данными из БД.
2. *Целостность* – уменьшение дублирования данных влечет их согласованность между собой.
3. *Гибкость* – изменение, редактирование как самих данных так и структуры БД при правильной организации работы с БД сравнительно простой процесс.
4. *Скорость* – возможность оперативного доступа ко всем хранимым данным в БД.

# Система управления базами данных (СУБД)

**СУБД** – представляет собой обобщенный инструмент манипулирования данными.

**СУБД** – система программных и языковых средств (информационная система) предназначенный для:

- создания БД;
- обеспечение эффективного доступа к данным БД;
- управление данными в БД;
- поддержание БД в актуальном состоянии для предметной области.



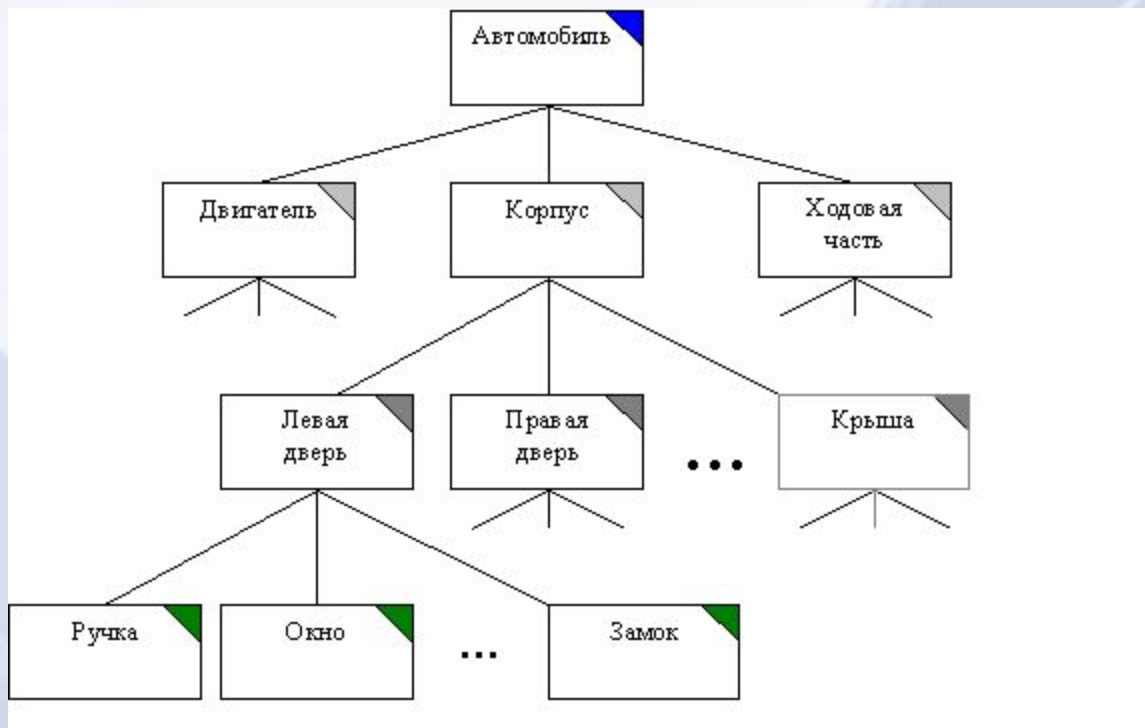
## Классификация СУБД

1. По типу управляемой базы данных СУБД.
2. По архитектуре СУБД и организации хранения данных.
3. По способу доступа СУБД к базе данных.



# По типу управляемой базы данных СУБД

**Иерархические СУБД - поддерживают  
древовидную организацию информации.**



## Особенности:

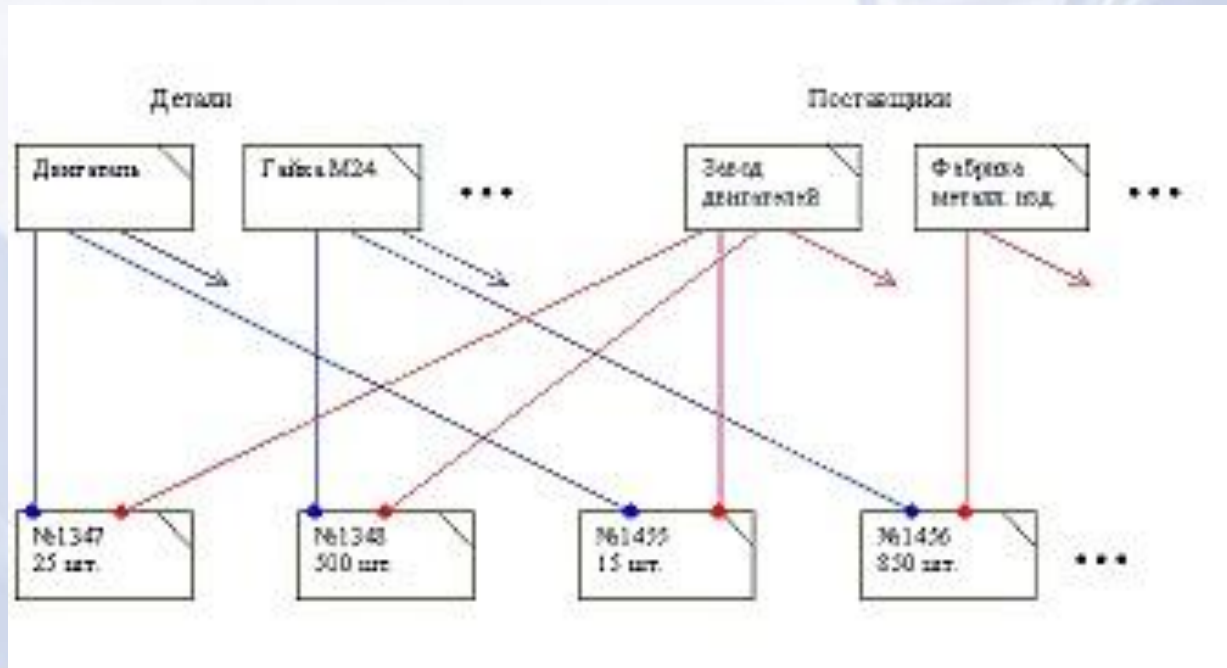
1. Связи между записями выражаются в виде **отношений предок/потомок**, а у каждой записи есть ровно одна родительская запись. Это помогает поддерживать ссылочную целостность.

2. Иерархические базы данных имеют **централизованную структуру**, т.е. безопасность данных легко контролировать.

## Недостатки:

1. Необходимы знания о **физическом порядке хранения записей**, так как отношения предок/потомок реализуются в виде физических указателей из одной записи на другую. Это означает, что поиск записи осуществляется **методом прямого обхода дерева**. Записи, расположенные в одной половине дерева, ищутся быстрее, чем в другой.

**Сетевые СУБД** - сетевая модель расширяет иерархическую модель СУБД, позволяя группировать связи между записями в множества.



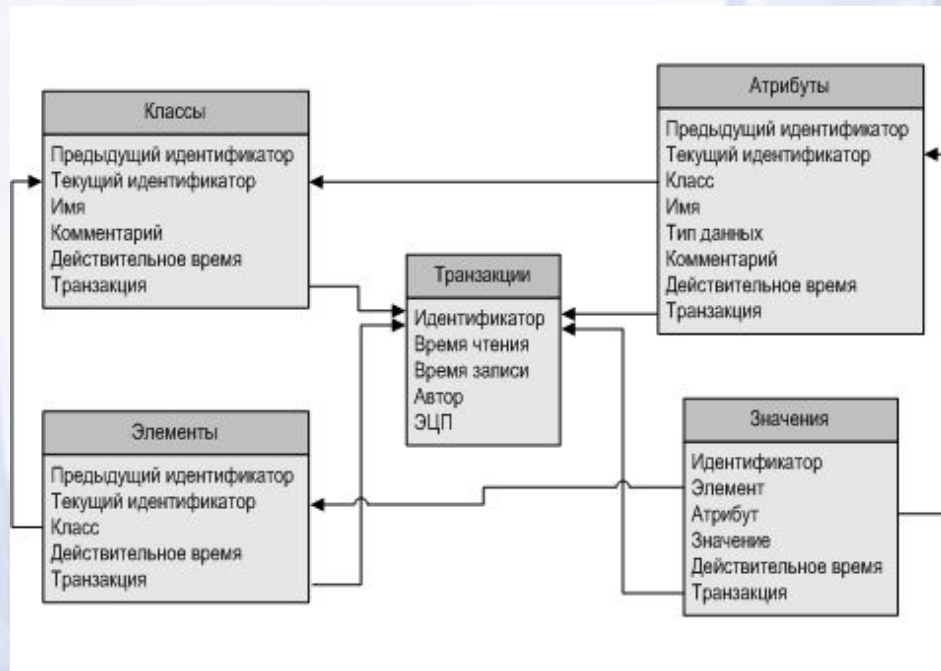
## Особенности:

- Как и в иерархической модели, связи ведут от родительской записи к дочерней, но на этот раз поддерживается множественное наследование.
- Сетевая модель поддерживает DDL (Data Definition Language — язык определения данных) и DML (Data Manipulation Language — язык обработки данных). Это специальные языки, предназначенные для определения структуры базы данных и составления запросов.

## Недостатки:

- Существует необходимость знания структуры БД.
- Готовую структуру трудно менять. Если вид таблицы претерпевает изменения, все отношения с другими таблицами должны быть установлены заново, чтобы не нарушилась целостность данных.

**Реляционные СУБД** - представляет собой централизованное хранилище таблиц, обеспечивающее безопасный одновременный доступ к информации со стороны многих пользователей.





## Особенности:

- применяется язык SQL, позволяющий формулировать произвольные, нерегламентированные запросы.

**Объектно-ориентированные СУБД** - позволяет, интерпретировать все свои информационные сущности как объекты, хранящиеся в оперативной памяти. Дополнительный интерфейсный уровень абстракции обеспечивает перехват запросов, обращающихся к тем частям базы данных которые находятся в постоянном хранилище на диске. Изменения, вносимые в объекты, оптимальным образом переносятся из памяти на диск.



## Особенности:

- Приложения получают возможность интерпретировать данные в контексте того языка программирования, на котором они написаны. Методы манипулирования данными всегда остаются одинаковыми независимо от того, находятся данные на диске или в памяти.

## Недостаток:

- тесная связь с применяемым языком программирования.

**Объектно-реляционные СУБД** - объединяют в себе черты реляционной и объектной моделей.

**Особенность:**

- база данных сохраняет свою табличную структуру, но способ обработки некоторых полей таблиц определяется извне

# По архитектуре СУБД и организации хранения данных

1. **Локальные СУБД** (все части локальной СУБД размещаются на одном компьютере);
2. **Распределенные СУБД** (части СУБД могут размещаться на двух и более компьютерах).

# По способу доступа СУБД к базе данных



## **Файл-серверные СУБД.**

В файл-серверных СУБД файлы данных располагаются централизованно на файл-сервере СУБД.

Ядро СУБД располагается на каждом клиентском компьютере.

Доступ к данным осуществляется через локальную сеть. Синхронизация чтений и обновлений осуществляется посредством файловых блокировок.

**Преимуществом** этой архитектуры является низкая нагрузка на ЦП сервера.

**Недостатком** — высокая загрузка локальной сети.

## **Клиент-серверные СУБД.**

Такие СУБД состоят из клиентской части (которая входит в состав прикладной программы) и сервера СУБД.

Клиент-серверные СУБД, в отличие от файл-серверных, обеспечивают разграничение доступа между пользователями и мало загружают сеть и клиентские машины.

Сервер является внешней по отношению к клиенту программой, и по надобности его можно заменить другим.

**Недостаток** клиент-серверных СУБД в самом факте существования сервера СУБД и больших вычислительных ресурсах, потребляемых сервером.

## **Встраиваемые СУБД.**

Встраиваемая СУБД — библиотека, которая позволяет унифицированным образом хранить большие объёмы данных на локальной машине.

Доступ к данным может происходить через SQL либо через особые функции СУБД.

Встраиваемые СУБД быстрее обычных клиент-серверных и не требуют установки сервера, поэтому востребованы в локальном ПО, которое имеет дело с большими объёмами данных.

# Жизненный цикл БД

Жизненный цикл можно разделить на 2 фазы:

1. Анализа и проектирования.
2. Реализации и функционирования.



# 1. Анализ и проектирование:

1. формулирование и анализ требований;
2. концептуальное проектирование;
3. проектирование реализации;
4. физическое проектирование.

## 2. Реализация и функционирование:

1. реализация БД;
2. анализ функционирования и поддержка;
3. модификация и адаптация.

**Спасибо за внимание!!!**