

# SQL — декларативный язык

- Мы говорим, **что** надо получить в результате запроса, но не говорим, **как**.
- Аналогично с таблицами — мы определяем атрибуты и связи между объектами, но не управляем физическим хранением.

# Что влияет на выполнение запроса?

- Производительность сервера.
- Конфигурация памяти.
- Особенности операционной системы.
- Внутренний оптимизатор СУБД.
- Использование «правильных» запросов.

# Выполнение запросов

- Многие команды SQL могут быть выполнены разными способами.
- Критерий оценки стоимости выполнения запроса — число обменов с устройствами внешней памяти.

# Стоимость выполнения

Стоимость (затраты) — это оценка ожидаемого времени выполнения запроса с использованием конкретного плана выполнения.

# Как выполнить запрос?

- Задача: построить план выполнения запроса.
- Уточнение задачи:
  - построить все возможные программы, результаты которых соответствуют указанным свойствам;
  - выбрать из множества этих программ наиболее эффективную.

# Шаги при выборе оптимального плана

- Обнаружить все корректные планы выполнения запроса.
- Сократить пространство планов, оставив наиболее эффективные.
- Выбрать наилучший план выполнения запроса.

# Пример: студенты 341 группы не старше 19 лет

```
SELECT * FROM Student  
WHERE GroupNumber=341 AND  
BirthDate >'1998/01/01'
```

# Примеры возможных вариантов исполнения запроса

- 1 Последовательно считать все записи.
- 2 Использовать INDEX(GroupNumber).
- 3 Использовать INDEX(BirthDate).



# Выбор оптимального плана выполнения запроса

- Цель СУБД — выполнить запрос наиболее эффективным способом.
- Оптимизатор СУБД — специальная компонента, которая строит планы запросов.

# Стадии оптимизации запросов:

1. Лексический и синтаксический анализ.
2. Привязка объектов базы.
3. Логическая оптимизация.
4. Построение возможных планов выполнения запроса.
5. Выполнение запроса.

# 1. Лексический и синтаксический анализ

- Лексический анализатор разбивает запрос на лексические единицы — лексемы.
- Синтаксический анализатор проверяет синтаксическую правильность запроса.

## 2. Привязка объектов базы

- Создаются внутренние представления запроса (таблицы, поля, константы).
- Информация об объектах базы данных выбирается из словаря-справочника данных.

# 3. Логическая оптимизация

Проводятся различные преобразования,  
«улучшающие» начальное представление запроса.

## 4. Построение возможных планов выполнения запроса

- Основа — информация о существующих путях доступа к данным:
  - прямой просмотр;
  - использование индексов.
- Оценка предполагаемой стоимости выполнения запроса.
- Статистика — основа для оценки стоимости запроса.

# Статистика

- Статистика — сведения о распределении значений данных.
- Статистика хранится в словаре-справочнике данных.

# Статистика по таблице

- Количество блоков данных в таблице.
- Количество пустых блоков данных.
- Количество записей в таблице.
- Средняя длина записи в таблице.
- Количество записей в блоке.
- Наличие индексов для таблицы.



# Статистика по индексам

- Общее количество проиндексированных записей.
- Минимальное и максимальное индексированные значения.
- Количество различных индексированных значений.
- Распределение значений в столбце.

# Использование статистики

Статистика влияет на метод выборки данных:

- сканирование таблицы;
- поиск по индексу.

Перед выполнением запроса оптимизатор пытается построить статистику, если ее до этого не было.

## 5. Выполнение запроса

- Запрос выполняется по оптимальному плану.
- План выполнения запроса сохраняется в разделяемой памяти и может быть использован повторно.
- Подготовленные к исполнению SQL-операторы помещаются в разделяемую SQL-область.