

# Лекция №5 29.09.23

Т.к. Microsoft Access является реляционной СУБД, то её основа это таблица. Таблица состоит из полей и записей. **Поля - это столбцы, а записи - это строки.** Некоторые поля таблицы могут быть помечены как ключевые. Через ключевое поле таблица связывается с полем другой таблицы. Для каждого поля даётся формат данных. Для каждого поля можно дорабатывать дополнительные свойства (например индексы).

Индексы используются для ускорения поиска и сортировки данных. **Индекс** - это упорядоченный список значений и ссылок на те записи, в которых хранятся эти значения. Индексы бывают 2 типов:

- **Простые** - это индексы, созданные по одному столбцу
- **Составные** - это индекс, созданный на нескольких столбцах

Применение индексов приносит не только преимущество, но и недостатки. Главным недостатком заключается в том, что при удалении и добавлении записей требуется обновлять индекс, что может замедлять работу. Поэтому индексы создают только для тех столбцов, по которым наиболее часто выполняется поиск.

**Запрос** - это набор инструкций, который можно использовать для обработки данных. Запрос не только возвращает значение результата, которые можно сортировать, нумеровать и т.д., но и с помощью запроса можно добавлять, сортировать и изменять данные. Существует 3 основных типа sql запросов:

- **Запрос на объединение** - собирает из разных таблиц, не изменяет данные в базовых таблицах. Запросы на объединение добавляют строки во временный набор записей, которые не сохраняются после закрытия запроса
- **Запрос к серверу** - не выполняются самим ядром СУБД Access, вместо этого они перенаправляются на удалённый сервер
- **Управляющий запрос** - при выполнении запроса этого типа, создаются новые, удаляются или изменяются объекты базы данных

**Отчёт** - это форматированное представление данных, которое выводится на экран, печать или файл. Позволяет извлечь из базы данных нужные сведения и представить их в удобном формате.

СУБД Access предоставляет широкие возможности по конструированию практического интерфейса для пользователя. СУБД Access под программированием подразумевается добавление функциональных возможностей с использованием макросов или программ на языке **VBA**.

**Макрос СУБД Access** - это именованная последовательность макрокоманд.

**Макрокоманда Access** - это поднабор команд, доступных в **VBA**.

## SQL

### Структура языка SQL

В современной СУБД, которая поддерживает реляционную модель для обработки данных используется стандартизированный язык SQL. Прототип этого языка был создан в IBM.

Первая его версия появилась в 1986г. и называлась **SQL-86** или **SQL 1**. Первая версия стандарта включала в себя команды для чтения, регулирования данными и некоторые средства для ограничения целостности.

В 1989г. взяли за основу стандарт SQL-86, доработали и получили **SQL-89** или **SQL 1.1**. Особых изменений он не получил.

Сильные изменения произошли в 1992г. с выходом **SQL-92** или **SQL 2**. Была добавлена возможность изменения таблиц, описаны служебные таблицы для хранения метаданных (название таблиц, название столбцов, имена пользователей и т.д.).

В 1999г. вышел **SQL 1999** или **SQL 3**. В нём добавили описание расширений языка, которые выходят за пределы реляционной модели (процедурный язык в виде хранимых процедур функций, пользовательские типы данных, рекурсивные запросы, регулярные выражения, откорректированные расширения).

В 2003 г. выходит **SQL 2003**. В него были добавлены новые команды, средства для работы с xml данными, средства просмотра таблиц, которое относится к

различным периодам времени.

В 2006 г. вышел **SQL 2006**, поддерживающий XQuery.

В 2008 г. был выпущен **SQL 2008**. В нём появились команда TRUNCATE TABLE и триггеры баз данных, по типу INSTEAD OF, добавление новых типов данных.

В 2011 г. представили **SQL 2011**. Обновление содержит в себе средства организации темпоральных баз данных, то есть баз данных, которые поддерживают хронологию.

В 2016 г. выходит **SQL 2016**, в котором появляются новые типы данных и функции для создания json документов.

В своей основе SQL является декоративным языком, то есть позволяет указать, что вы хотите сделать, то есть какие данные из каких таблиц в каком виде вы хотите получить. Что сделать. как сделать и как обработать для вывода желаемого результата.

Процедурные и откорректированные объекты SQL оформляются как отдельный язык со своим названием у каждого отдельного производителя.

## Подъязыки SQL

Основные подъязыки SQL:

1. **DQL** - язык запросов к данным, который включает в себя единственную команду **SELECT**. Эта команда позволяет считывать из таблиц.
2. **DML** - язык манипулирования данными. В нём присутствуют команды для изменения таблицы путём вставки **INSERT**, обновления **UPDATE** и удаления **DELETE**. Также в него очень часто включают **SELECT**.
3. **DDL** - язык определения данных. Позволяет создавать **CREATE**, изменять **ALTER** и удалять объекты базы данных **DROP**. Так же к нему относят команду **TRUNCATE TABLE** - позволяет быстро отчистить таблицу, сохранив её структуру.
4. **DCL** - язык управления данными. Сюда входят команды для управления определённых прав управления пользователями. **GRANT** - выдача прав, **REVOKE** - отзыв прав.
5. **TCL** - язык управления транзакциями. Включает в себя команды для подтверждения **COMMIT** и отката **ROLLBACK** транзакций.

Имена объектов базы данных называют **идентификаторами**. Правила формирования каждой СУБД свои. В общем случае идентификаторы могут включать только символы из определённого набора, должны начинаться с буквами, максимальная длина ограничена, не должны совпадать с зарезервированными словами SQL или конкретной СУБД. Идентификаторы регистронезависимы. Во многих СУБД большую часть правил можно обойти, заключив идентификатор можно обойти заключав идентификатор в двойные кавычки (то есть " ").

**Литеры** - это любые const, использующиеся при формировании команд и не являющиеся идентификаторами. Кроме того синтаксическим элементом языка SQL относят переменные, операторы, функции и выражения.

Как и любой язык SQL допускает комментарии. `--` - однострочный комментарий, `/* */` многострочный комментарий.

## Типы данных

1. **Строковые** - `CHAR`, `NVARCHAR` (для хранения символьных строк)
2. **Числовые** - `INTEGER`, `DIGINT`, `SMAUDIT`, `NUMERIC`, `FLOAT`
3. **Логические** - `BOOLEAN` (true, false, null)
4. **Дата/время** - `DATE` / `TIME`

## Создание домена

Приведённые типы приведённые с помощью CHECK используются для создания неявного атрибута. Согласно стандарту SQL пользователь может явно создать домен на основе одного из базовых типов данных с помощью команды CREATE DOMAIN.

## Функции SQL

Все функции можно разделить на 2 группы:

- **Однострочные** (значение 1 → обработка → результат 1, значение 2 → обработка → результат 2 и т.д.)

- **Групповые** (агрегатные) (значение 1 + значение 2 → обработка → результат)