**T-SQL:** Язык реляционной базы данных в **SQL Server**

**1.DDL: язык определения данных**

* **CREATE создание**
* **ALTER изменение**
* **DROP удаление**

**2.DML язык манипулирования данными**

* **SELECT** основной оператор для поиска информации в базе данных
* **INSERT** используется для ввода информации в таблицу
* **UPDATE** Для изменения строк таблицы
* **DELETE** Чтобы удалить одну или несколько строк

**3.DCL язык управления данными**

* **GRANT** явно предоставляет одно или несколько разрешений
* **REVOKE** явно запрещает участнику иметь одно или несколько разрешений
* **DENY** удаляет существующие разрешения

**4.TCL язык управления транзакциями**

Транзакция – это несколько DML-операторов, которые либо **все** выполняются, либо все не выполняются

это логическая рабочая единица, которая выполняет одно или несколько действий в базе данных.

Транзакция задает последовательность инструкций языка Transact-SQL, применяемую программистами базы данных **для объединения в один пакет операций чтения и записи для того, чтобы система базы данных могла обеспечить согласованность данных**.

* **BEGIN TRANSACTION** определяет начальную точку явной транзакции
* **COMMIT TRANSACTION** применяет изменения данных к базе данных, и измененные данные становятся постоянными.
* **ROLLBACK TRANSACTION** помогает отменить все изменения данных, примененные транзакцией
* **SAVE TRANSACTION** определить точку сохранения в транзакции

**WHERE,**

**DISTINCT,** позволяет не выводить повторяющиеся строки

**TOP** ограничивает количество результирующих строк

**Файл** Это объекты, в которых хранится определенная информация.

**Файловые группы – поименованный набор файлов БД**

Как минимум, каждая база данных SQL Server содержит два файла операционной системы: файл данных и файл журнала. Файлы данных содержат данные и объекты, такие как таблицы, индексы, хранимые процедуры и представления. Файлы журнала содержат сведения, необходимые для восстановления всех транзакций в базе данных

**Все файлы БД, кроме файлов журнала транзакций, распределены по файловым группам:**

* **Первичная** Содержит сведения, необходимые для запуска базы данных, и ссылки на другие файлы в базе данных
* **Вторичная** Необязательные определяемые пользователем файлы данных.

**Какие файлы могут быть в первичной/вторичной группе**

* Во вторичных файловых группах могут располагаться только вторичные файлы.
* В первичной файловой группе помимо обязательного первичного файла тоже могут быть расположены вторичные файлы.

**Расширения файлов:**

* первичный файл (.**mdf**)
* вторичные файлы (.**ndf**)
* файлы журнала транзакций (.**log**)

**Способы задания первичного/вторичного ключа**

CONSTRAINT PK\_guid PRIMARY KEY (Guid)

CONSTRAINT FK\_idName FOREIGN KEY (idName)

Номер\_товара int foreign key references Товары(Номер\_товара),

Номер\_товара int primary key not null,

**Предикаты**

Предикат BETWEEN позволяет выбирать строки с заданным диапазоном значений.

Ключевое слово LIKE осуществляет сравнение полей с заданным шаблоном, при этом в шаблоне символ **%** обозначает любое количество любых символов

Ключевое слово IN отбирает строки, в которых значение анализируемого поля совпадает с одним из значений, записанных справа от IN в круглых скобках

AND с какого-то значения по какое-то (должны быть выполнены оба условия)

NOT **служит для задания противоположно заданного условия**

OR должно быть выполнено хотя бы одно условие.

Ограничения, накладываемые на столбцы таблиц баз данных, предотвращают появление данных, не соответствующих предварительно заданным свойствам таблиц

|  |  |
| --- | --- |
| Ограничение | Действие ограничения целостности |
| **data type**  тип данных | Предотвращает появление в столбце значений, не соответствующих типу данных |
| **not null**  запрет значений **null** | Предотвращает появление в столбце значений null |
| **default**  знач. по умолчанию | Устанавливает значение в столбце по умолчанию при выполнении операции INSERT |
| **primary key**  первичный ключ | Предотвращает появление в столбце повторяющихся значений и пустого значения |
| **foreign key**  внешний ключ | Устанавливает связь между таблицей со столбцом, имеющим свойство **foreign key** и таблицей, имеющей столбец со свойством **primary key**; |
| **unique**  уникальное значение | Не допускает пустые и повторяющиеся значения, не может быть использовано для связи с полем другой таблицы |
| **check**  проверка значений | Предотвращает появление в столбце значения, не удовлетворяющего логическому условию |