**3 лаба**

**Язык SQL** (Structured Query Language, язык структурированных запросов) – специализированный язык, предназначенный для написания запросов к реляционной БД. Язык реляционной базы данных в **SQL Server** называется **Transact-SQL** (**T-SQL**).

**Операторы:**

* DDL - Data Definition Language - язык определения данных
* DML - Data Manipulation Language - язык манипулирования данными
* TCL - Transaction Control Language - язык управления транзакциями
* DCL - Data Control Language - язык управления данными

**СУБД** - Программная реализация **технологии** хранения, извлечения, обновления и обработки данных в базе данны

Язык **DDL** содержит три обобщенных оператора:

* **create** *объект* (создание объекта базы данных),
* **alter** *объект* (изменение характеристик объекта)
* **drop** *объект* (удаление существующего объекта).

Эти опера­торы создают, изменяют и удаляют объекты базы данных, такие как сама база данных, таблицы, столбцы и индексы.

Язык **DML** содержит операторы, которые манипулируют данными: осуществляя выборку информации (**SELECT**), добавление (**INSERT**), удаление (**DELETE**) изменение (**UPDATE**).

При записи операторов можно использовать на клавиатуре любой регистр.

При создании таблиц используются различные ограничения. Ограничения, накладываемые на столбцы таблиц баз данных, предотвращают появление данных, не соответствующих предварительно заданным свойствам таблиц. Эти ограничения называются ограничениями целостности.

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

**4 лаба**

Соединение таблиц **INNER JOIN** (внутреннее соединение) наиболее часто используемый вид соединения реляционных таблиц. Результирующий набор создается следующим образом: выполняется декартово произведение (каждая строка одной таблицы соединяется с каждой строкой другой); из полученного результата выбираются строки, удовлетворяющие указанному условию; из всех столбцов результирующего набора выбираются столбцы, указанные в списке **SELECT**.

Предикат **LIKE** - этом выбрать эл-ты, которые содержат буквосочетание ‘**…**’

Здесь в выражении **CASE** каждое предложение WHEN содержит логическое выражение. Эти выражения проверяются на истинность сверху вниз, и при первом успешном сравнении формируется результирующее значение, указанное за ключевым словом **THEN**. В том случае, если ни одно из логических **WHEN**-выражений не принимает истинного значения, в качестве результата **CASE** формируется значение, указанное в предложении **ELSE**. В последнем столбце выводятся тексты из выражения **CASE**.

Левое внешнее соединение **LEFT OUTER JOIN** включает в набор несоединенные строки таблицы, имя которой записано слева от ключевых слов **LEFT OUTER JOIN,** а правое внешнее соединение **RIGHT OUTER JOIN** – несоединенные строки таблицы, имя которой записано справа.

Встроенная функция **isnull** принимает два параметра и проверяет их значения на **NULL** слева направо. Функция возвращает первое значение, не равное **NULL.**

Операция является ***коммутативной***, если формируемый результирующий набор не зависит от порядка, в котором указаны исходные таблицы.

При использовании соединения **CROSS JOIN** каждая строка одной таблицы соединяется с каждой строкой другой таблицы.