**Список примеров запроса:**

**-- Закомментировать строку**

**Создание таблицы:**

**CREATE TABLE vendors (**

**id SERIAL PRIMARY KEY,**

**name TEXT NOT NULL**

**);**

**Вставка значения в определенный столбец:**

INSERT INTO vendors (name) VALUES ('ООО Ромашка');

Просмотр всей таблицы vendors:

SELECT \* FROM vendors;

**Вывод того, что происходит в базе (время сканирования и т.д.)**

**EXPLAIN** SELECT \* FROM vendors;

**SQL** (structured query language — язык структурированных запросов) — это язык программирования, позволяющий выполнять любые операции с базами данных как таковыми и с любыми данными, которые хранятся в них.

* *SQL* — декларативный язык, то есть вы описываете конечный результат, а не алгоритм получения этого результата, как принято в императивных языках (*Python, Go*, *C*, *Java* и так далее — императивные языки).
* *SQL* появился в 70-х годах и относится к семейству *pascal-*подобных языков, а в наши дни больше распространены *C*-подобные языки.
* Для *SQL* нет сред разработки (*IDE*), сравнимых по функционалу с современными средами вроде *VS Code*, *PyCharm* и так далее.

**Операторы (команды)** языка SQL принято в соответствии со стандартом разделять на несколько групп. Нас будут интересовать только два основных подмножества — **DDL** (data definition language — язык описания структуры данных) и **DML** (data manipulation language — язык манипулирования данными). Первая группа операторов предназначена для работы с базами данных и их объектами как таковыми, а вторая — для работы с самими данными.

**DDL** предназначен для создания, изменения и удаления самих БД и их объектов — таблиц, индексов, последовательностей, представлений, ограничений и так далее.

**DML** предназначен для работы с самими данными, которые хранятся в БД. DML позволяет читать данные из БД, изменять их, удалять и создавать новые записи. DML всегда работает со строками в таблицах БД.

№ Подмножество Оператор Описание

1 DDL CREATE Создаёт БД или объект БД (таблицы, индексы, триггеры и так далее).

2 DDL ALTER Изменяет объект БД.

3 DDL DROP Удаляет БД или объект БД.

4 DML SELECT Чтение набора строк из одной или нескольких таблиц БД.

5 DML INSERT Создание новой записи (строки) в таблице БД.

6 DML UPDATE Обновление строки в таблице БД.

7 DML DELETE Удаление строки из таблицы БД.

**Важно!** Различные СУБД реализуют свой «диалект» языка *SQL*. При этом базовые возможности, указанные в таблице выше, остаются неизменными (хотя синтаксис может немного различаться). А вот дополнительные возможности сильно различаются между разными СУБД. Например, алгоритмы индексирования или названия типов данных в *PostgreSQL* и *Microsoft SQL Server* могут отличаться значительно. В данном курсе мы используем*PostgreSQL*, но сведения из данного модуля по большей части применимы ко всем РСУБД.

**Примечание.** Помимо указанных операторов, в языке *SQL* существует большое количество встроенных функций: для работы со временем, для работы с *JSON*, для вычисления статистических значений, математические функции, логические операторы и так далее. Набор доступных функций сильно зависит от выбранной СУБД.

**Язык *SQL* обладает рядом особенностей синтаксиса, о которых стоит помнить. Самые важные из них:**

* + *SQL* не чувствителен к регистру. То есть оператор SELECT можно записать как select, Select или даже sElECt. Однако принято все операторы и ключевые слова писать большими буквами, а все имена объектов и функций — маленькими. Придерживайтесь этого правила.
  + Каждый оператор должен заканчиваться точкой с запятой. Если вы её не поставите, будет ошибка, и запрос не выполнится.
  + Строки заключаются в одиночные кавычки.
  + Проверка равенства и установка значения осуществляются оператором =.
  + Запросы не чувствительны к переносу строк, пробелам и табуляции. Можно форматировать запросы любым удобным способом.
  + Однострочные комментарии начинаются с --.