**Модуль 6: Язык SQL (додатково)**

**Заняття 1:**

**Почему SQL?**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc2#%D0%BF%D0%BE%D1%87%D0%B5%D0%BC%D1%83-sql)

SQL – Structured Query Language (язык структурированных запросов).

Давным-давно до 86-х годов языка SQL не было, но реляционные базы данных были, реляционными они называются, потому что пошли от английского слова relation - связь. И почти у каждой компании была своя самописная база данных у которой был свой язык взаимодействия и в один прекрасный момент разработчики собрались и сказали, довольно! И сделали стандарт для взаимодействия с реляционными базами данных который и назвали SQL в 1983 году Международная организация по стандартизации (ISO) и Американский национальный институт стандартов (ANSI) начали разработку этого языка и закончили в 1986.

И теперь практически не важно какая именно база будет в компании, PostgreSQL, MySql, Oracle, SQLite основная суть остается неизменной, сделать запросы и записать данные в базу вы точно сможете, так как это язык.

**Структура языка**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc2#%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0-%D1%8F%D0%B7%D1%8B%D0%BA%D0%B0)

Все операторы SQL делятся на 3 группы.

* DDL (Data Definition Language) – Операторы определения данных. Операторы, которые объясняют как создать или изменить структуру данных
* DML (Data Manipulation Language) – Операторы манипуляции данными. Взаимодействия с базой данных, что мы хотим взять, положить и прочее
* DCL (Data Control Language) – Операторы определения доступа к данным. Управление правами доступа.

Язык SQL совершенно регистра независимый. Но принято писать команды заглавными буквами (верхний регистр).

Мы разберем их по отдельности по курсу, пока я опишу, что входит в эти группы.

**Операторы определения данных (DDL)**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc2#%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80%D1%8B-%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F-%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85-ddl)

Нужны для:

* CREATE DATABASE - Создание базы данных
* DROP DATABASE - Удаления базы данных
* CREATE TABLE - Создание таблицы
* DROP TABLE - Удаление таблицы
* ALTER DATABASE Модификация базы
* ALTER TABLE Модификация таблицы

Как мы видим основные операторы это CREATE, DROP и ALTER

**Операторы манипуляции данными (DML)**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc2#%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80%D1%8B-%D0%BC%D0%B0%D0%BD%D0%B8%D0%BF%D1%83%D0%BB%D1%8F%D1%86%D0%B8%D0%B8-%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%BC%D0%B8-dml)

Это вездесущие операторы с которыми вы будете чаще всего работать:

* SELECT - Получить данные из таблицы
* UPDATE - Обновить данные в таблице
* INSERT - Добавить данные в таблицу
* DELETE - Удалить данные из таблицы

**Операторы определения доступа к данным (DCL)**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc2#%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80%D1%8B-%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F-%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%B0-%D0%BA-%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%BC-dcl)

Управление правами доступа, как к пользователю так и к таблице, так же транзакции тоже считаются контролем данных.

* GRANT - права пользователей
* CONNECT - соединение с сервером
* LOCK TABLE - блокировка таблицы
* UNLOCK TABLE - разблокировать таблицу
* COMMIT - так же как и в git добавить изменения
* ROLLBACK - отменить изменения в транзакции (если не было COMMIT)

**Синтаксис**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc2#%D1%81%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81)

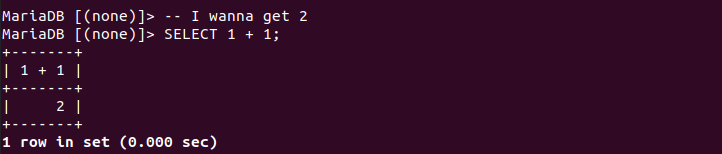
Как я писал выше все команды заглавными буквами.

Разделитель для запроса это точка с запятой ";"

Вот какие бывают комментарии:

* "--" два минуса - это однострочный комментарий, стандартный, который железно работает во всех базах
* "#" - тоже комментарий
* "/\*\*/" - многострочный комментарий.

Пример запроса:



Что я сделал, написал комментарий "-- I wanna get 2", он никак не отработал, это понятно из того, что ошибок нет, а потом сделал запрос "SELECT 1 + 1;". Результат выполнения SQL всегда таблица, колонка называется 1 + 1, а ее значение 2.

**Установка и настройка**

В зависимости от вашей системы у вас будут разные пути установки.

Вы можете установить классический MySQL или ее форк MariaDB, я рекомендую MariaDB для этого есть аргументы, MySQL выкупил ORACLE с одной целью, чтобы убрать конкурента на рынке и перестал ее поддерживать, но разработчик MySQL сделал форк и назвал ее MariaDB тем самым продолжил разработку базы, и деньги получил, и продукт сохранил.

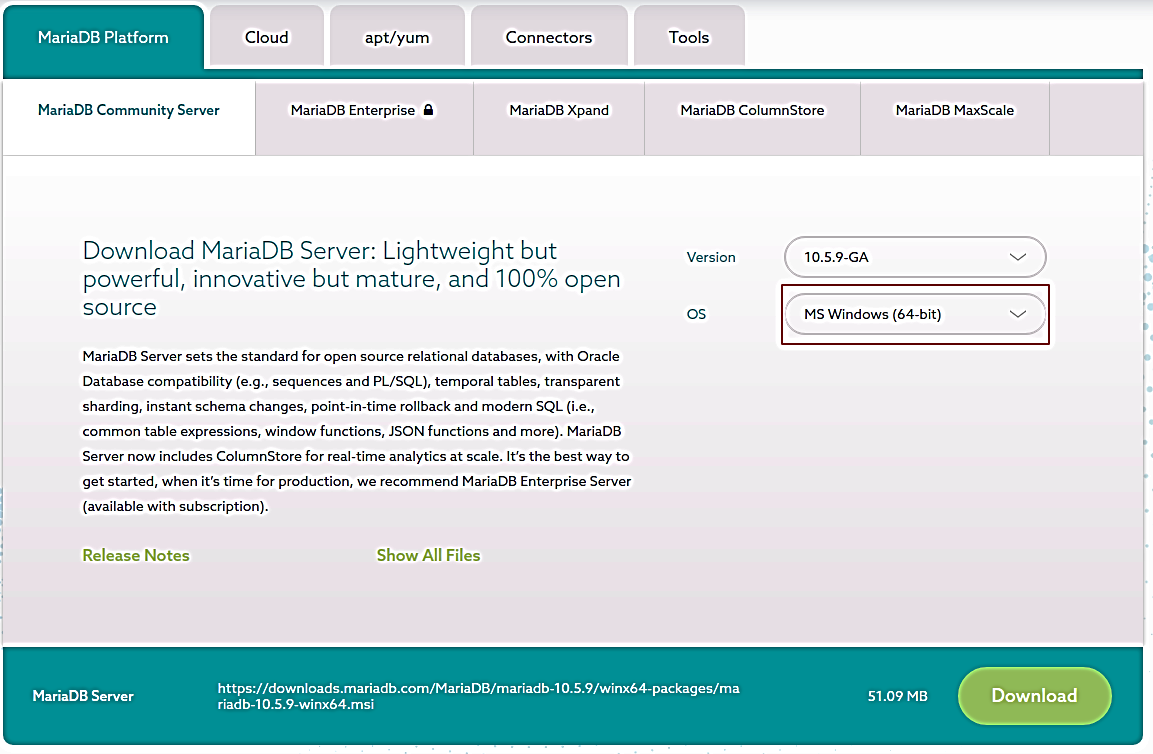
**Установка сервера**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc3#%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80%D0%B0)

**Установка на WINDOWS**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc3#%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-%D0%BD%D0%B0-windows)

Как и все в windows вы просто ее устанавливаете.

Переходим на сайт производителя <https://mariadb.com/downloads/>

Выбираем версию постарше и выбираем, что это windows и ваш разряд системы.



И дальше следуем инструкции по установке ниже.

<https://mariadb.com/kb/en/installing-mariadb-msi-packages-on-windows/>

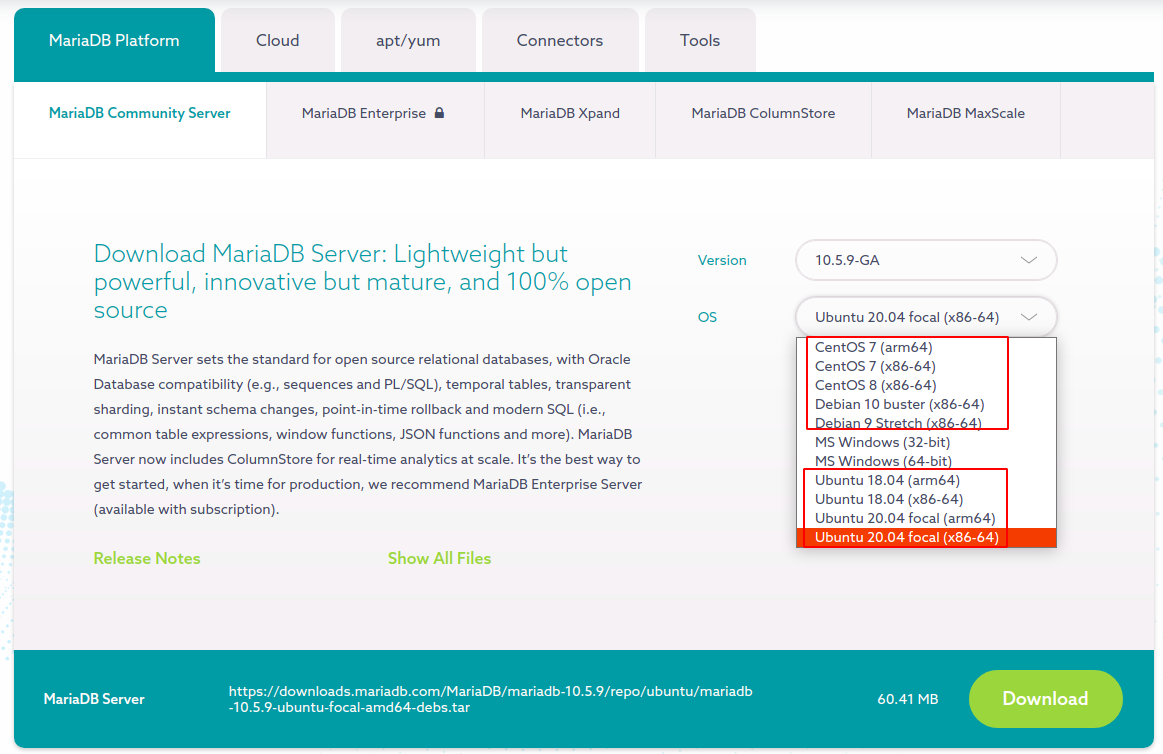
Я не рекомендую на локальную версию ставить пароль, это просто неудобно.

**Установка на LINUX**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc3#%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-%D0%BD%D0%B0-linux)

Я пользователь Ubuntu, но не думаю, что на других сборках все как-то иначе.

Вы можете так же установить из архива с офф сайта <https://mariadb.com/downloads/>

Выбираем версию постарше и какой у вас дистрибутив.



Я лично предпочитаю более простой путь.

sudo apt update

sudo apt install mariadb-server

Copy

Если хотите добавить пароль, можете выполнить скрипт, но не рекомендую для локальной версии, это неудобно.

sudo mysql\_secure\_installation

Copy

Статья на эту тему <https://www.digitalocean.com/community/tutorials/how-to-install-mariadb-on-ubuntu-20-04-ru>

Так же у некоторых были проблемы с тем, что порт был неправильно указан, это проблема решается следующим образом.

Заходим в базу

mysql -u root

Copy

Делаем запрос

*SHOW* *GLOBAL* VARIABLES LIKE 'PORT';

Copy

Как видим у меня правильный порт, но если по каким-то причинам там не 3306, а 0 то нужно поправить конфиг.

Вот хорошая статья на эту тему <https://www.tecmint.com/change-default-mysql-mariadb-port-in-linux/>

**Установка на MAC**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc3#%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-%D0%BD%D0%B0-mac)

Устанавливаем по инструкции от производителей

<https://mariadb.com/kb/en/installing-mariadb-on-macos-using-homebrew/>

Я не рекомендую на локальную версию ставить пароль, это просто неудобно.

**Работа в консоли**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc3#%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0-%D0%B2-%D0%BA%D0%BE%D0%BD%D1%81%D0%BE%D0%BB%D0%B8)

Так как MariaDB это форк MySQL сама команда запуска будут одинаковые.

Открываем командную строку и заходим в базу.

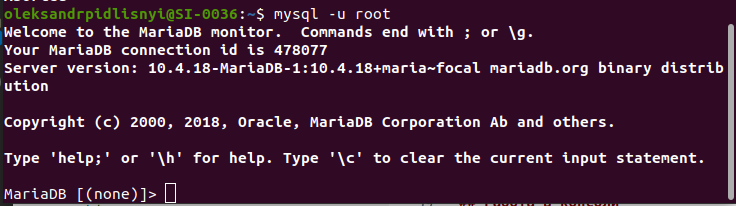
mysql -u root

Copy

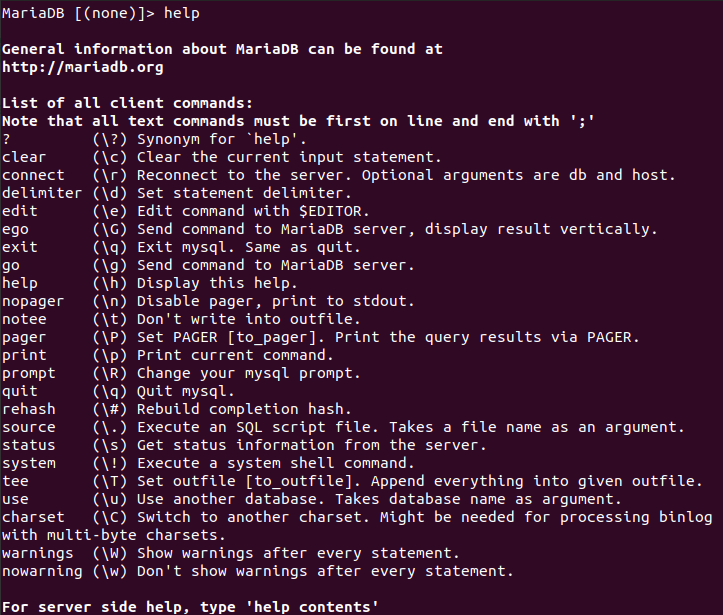
Если вы установили пароль, то можете вписать его используя атрибут **-p**

mysql -u root -p qwerty [последним параметром может идти имя базы данных]

Copy



После этого мы попадаем во внутрь консоли, все допустимые команды консоли можно посмотреть если написать команду **help**.



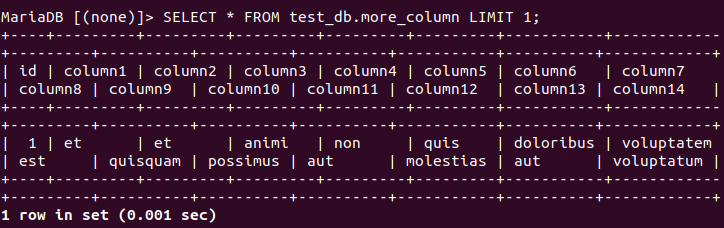
Консоль SQL не понимает команды enter и поэтому нужно ставить так называемое окончание.

Стандартным окончанием является точка с запятой "**;**".

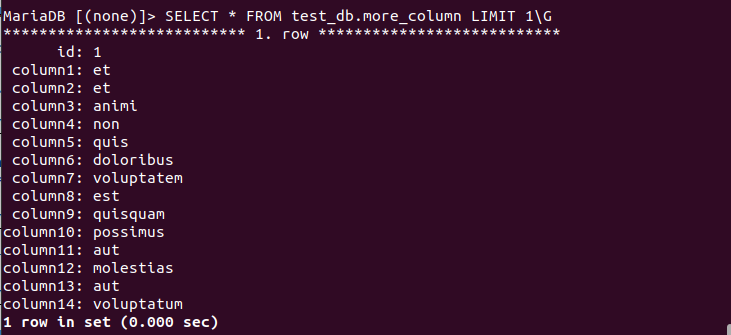
Синонимом "**;**" является **\g** – go

При большом количестве информации могут быть проблемы c отображением, что бы отобразить вертикальный список используется команда **\G**

Приведу пример когда это действительно нужно. Я сделал таблицу в которой много колонок, попробуем стандартным методом отобразить.



Как мы видим визуально не видно где содержимое колонки, а где сама колонка, попробуем через **\G**



Теперь мы явно видим, где колонка, а где ее значение.

Если мы не хотим выполнять запрос, например, написали с ошибкой, то нужно использовать **\c**

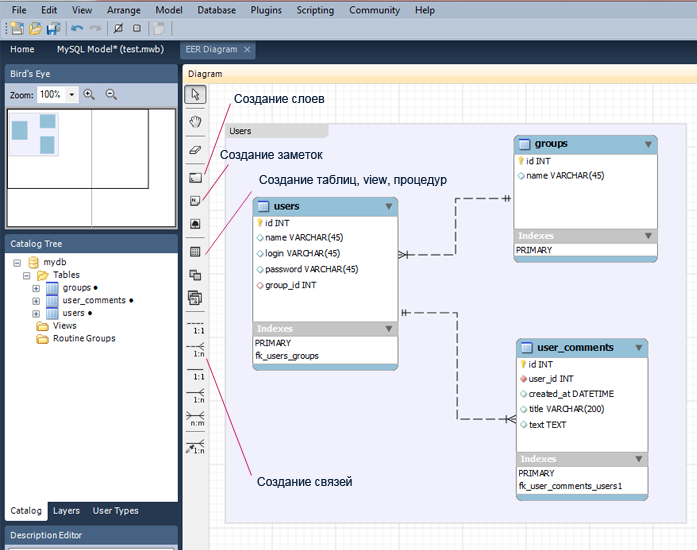
Рассмотрим еще полезные команды:

* \r – connect соединение с сервером, если соединение было, например, утрачено, мы можем так переконектится
* \q – exit альтернатива Ctrl + C
* \T – tee – включает функции записи непосредственно в файл, как пример tee /save\_to\_file.txt
* \t – notee – отключает tee
* \W – вывод ошибок
* \w – отключить вывод ошибок.

**Работа в GUI**[**#**](https://goit.global/textbooks/mysql-qhg4na/v1/docs/module1/docs/doc3#%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0-%D0%B2-gui)

Программ для работы с базой достаточно много, MySQL Workbench, HeidiSQL, DBeaver и другие на любой цвет и вкус.

**MySQL Workbench** - мне нравится в нем создание схем для проектирования.

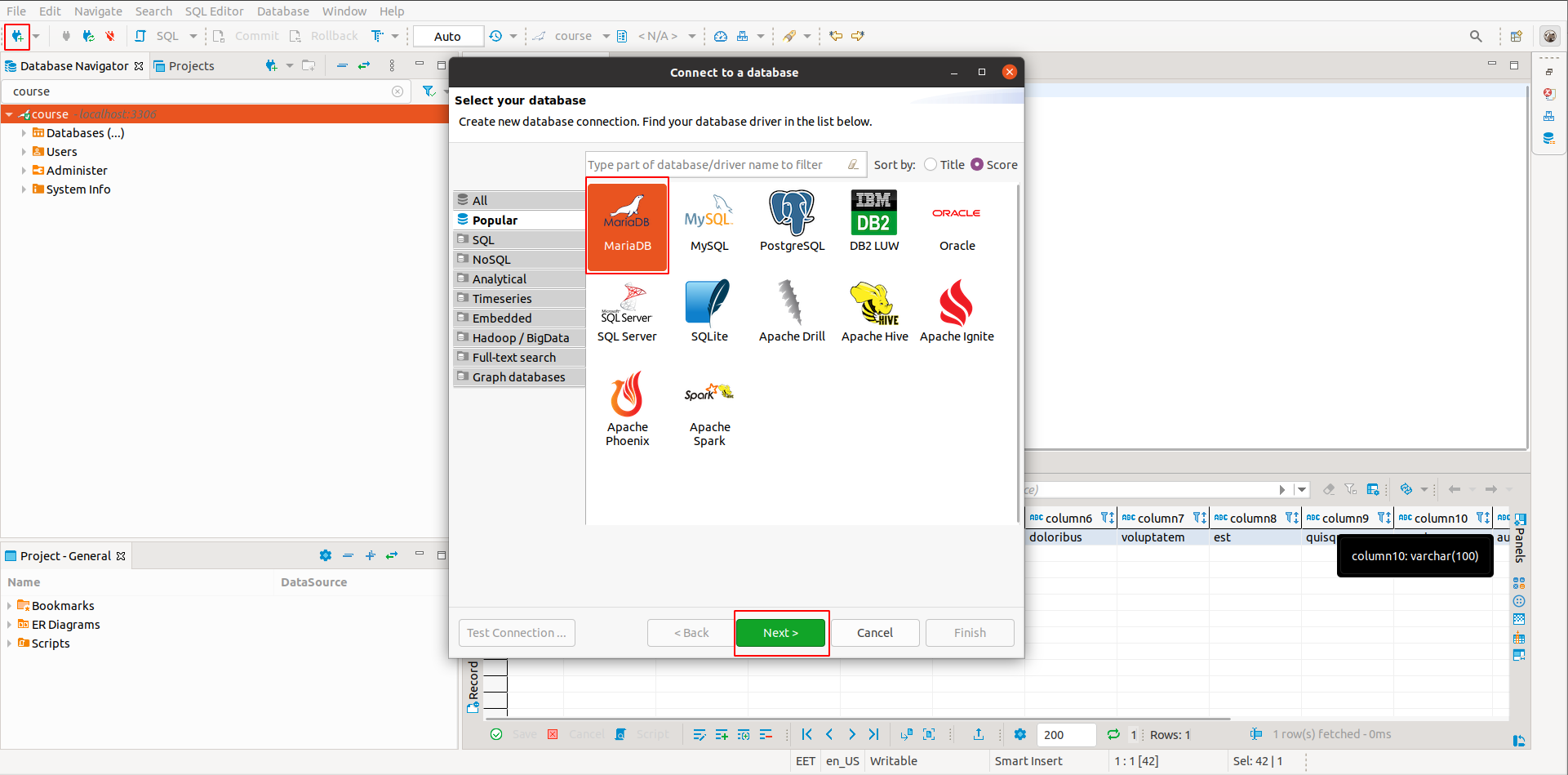


Но он сильно привязан к MySQL, если на работе попросят подключиться к другой базе, вы получите ошибку, поэтому я использую его только для проектирования базы.

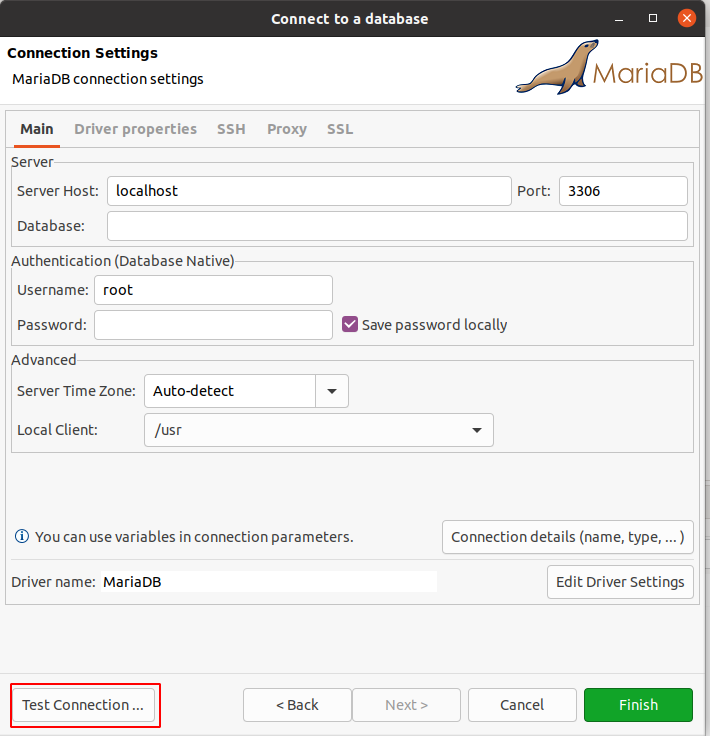
Рекомендую использовать **DBeaver** работает с большинством баз.

Скачиваем с сайта разработчиков <https://dbeaver.io/download/>

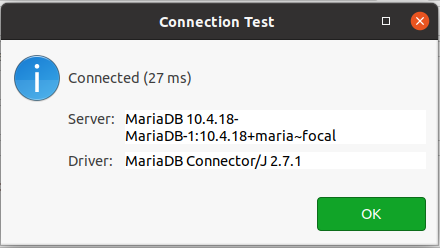
Теперь нам нужно подключиться к нашей базе.



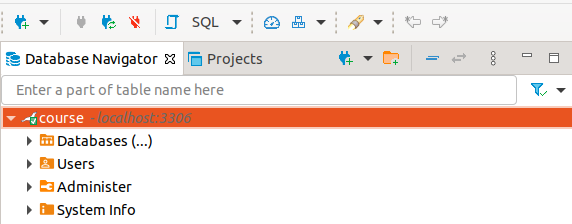
Начальные настройки уже должны быть правильные, если вы устанавливали пароль впишите его в поле password.



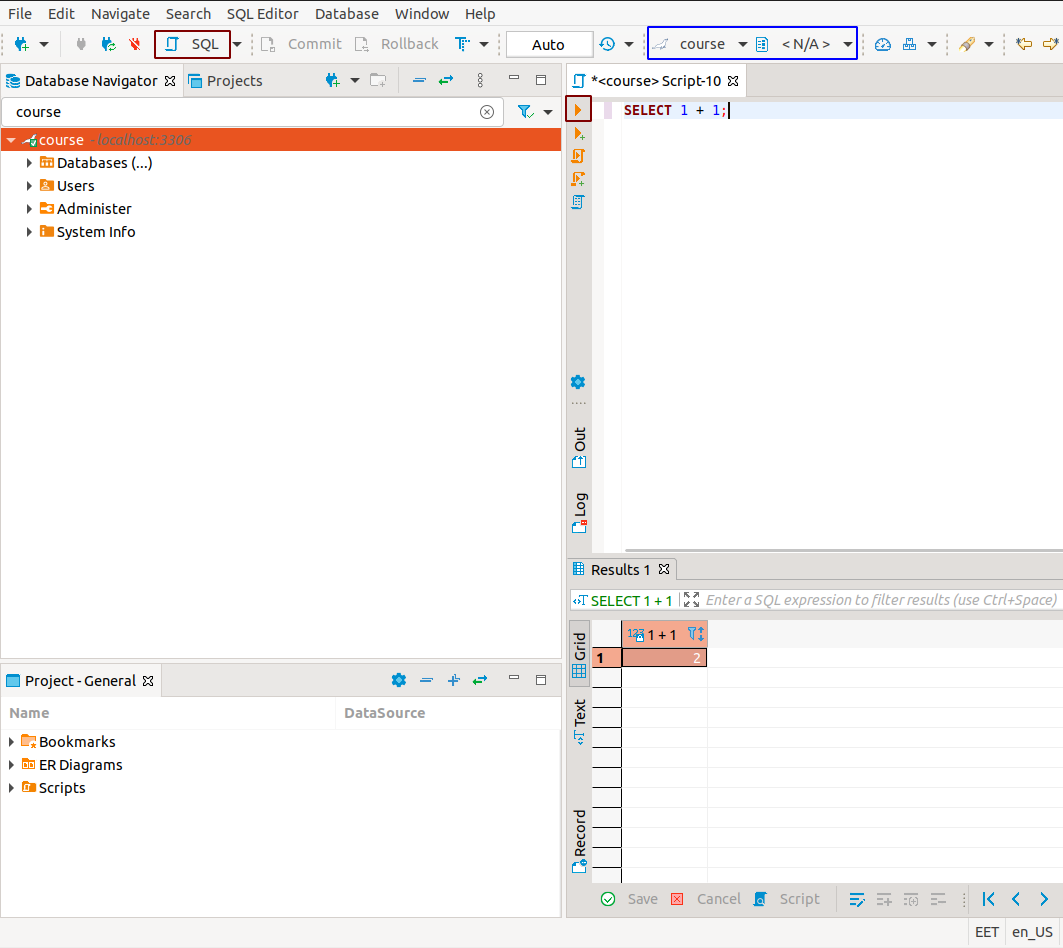
Нажимаем Test Connection должно быть следующее сообщение.



Если все хорошо у вас в списке появиться сервер, откройте его, у вас он будет называться localhost

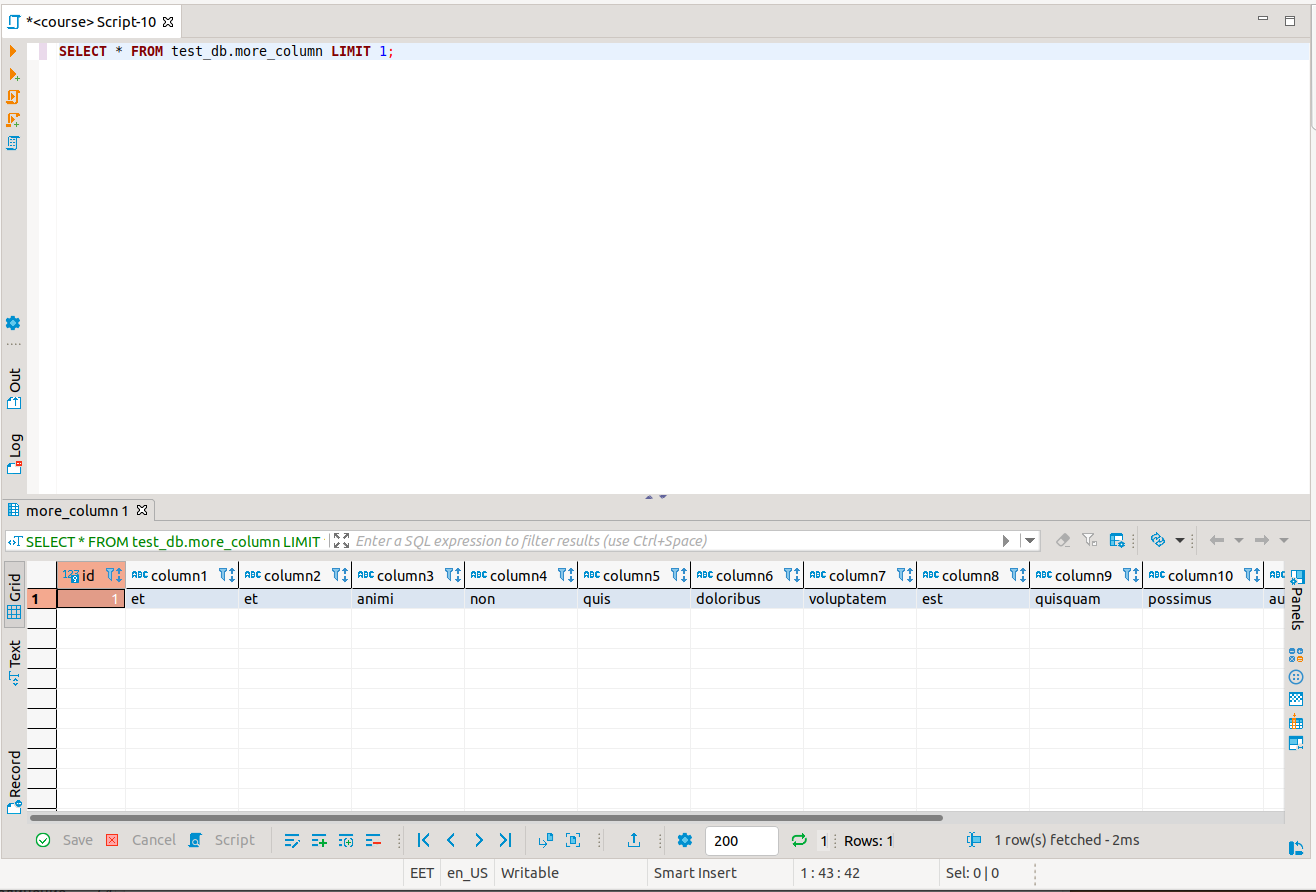


Давайте сделаем первый запрос, нажимаем на значок SQL, обращаем внимание подключены ли мы в серверу, я выделил синей рамкой, пишем запрос и нажимаем на значок похожий на плей.



И теперь мы сразу видим результат такой же как получали в консоли и так же у нас остался сам запрос который мы можем модифицировать если результат нас не устроил.

Помните запрос который мы выводили через **\G** так как ответ не помещался в консоли, тут такой проблемы нет.



Тут есть скролл и можно просмотреть весь результат.