

# MySQL

ВВЕДЕНИЕ И ОСНОВЫ

## СОДЕРЖАНИЕ

<b>СОДЕРЖАНИЕ</b>	<b>2</b>
<b>ВВЕДЕНИЕ</b>	<b>2</b>
<b>ОБЩЕЕ ОПИСАНИЕ СУБД MYSQL</b>	<b>4</b>
ОСНОВЫ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ	4
ИСТОРИЯ, ПРИМЕНЕНИЕ И ОСОБЕННОСТИ MYSQL	6
<b>УСТАНОВКА MYSQL</b>	<b>9</b>
<b>ОБЗОР УТИЛИТ И КОНФИГУРАЦИИ MYSQL</b>	<b>24</b>
<b>НАЧАЛО РАБОТЫ С MYSQL</b>	<b>27</b>
КЛИЕНТ КОМАНДНОЙ СТРОКИ	27
MYSQL SHELL	29
MYSQL WORKBENCH	30

## ВВЕДЕНИЕ

- Современный мир информационных технологий невозможно представить без применения баз данных. Уже само название, “информационные технологии”, говорит нам о том, что в основе данной области знаний лежит **информация**, которую необходимо правильно хранить, а также правильно структурировать для быстрого и удобного доступа при дальнейшем использовании. Именно этой деятельностью – хранением и структурированием информации – и занимаются базы данных и сопутствующие им технологии (т.н. СУБД – системы управления базами данных).
- При приеме на работу знание баз данных может не являться основным условием – скорее всего это будет какой-либо язык программирования (Python, Java, JavaScript, PHP) или знания в той области, которой занимается компания-наниматель (маркетинг, бухгалтерия, статистика). Однако в списке дополнительных требований почти всегда можно найти умение использовать ту или иную СУБД. Это показывает глобальную актуальность обозреваемой технологии – где бы вы ни работали, вы сможете использовать знания баз данных для продвижения своей карьеры, т.к. такие знания дают вам преимущество над другими соискателями или работниками.
- На что же может рассчитывать человек, хорошо знакомый с базами данных? Его карьерные перспективы можно разделить на три условных блока:
  1. **Специалист в области данных (Data Scientist)**. Он должен уметь извлекать необходимую информацию из самых разнообразных источников, видеть логические связи в системе собранной информации и на основе количественного анализа разрабатывать эффективные бизнес-решения. В Amazon, например, такой специалист может разрабатывать систему контекстной рекламы.<sup>1</sup>
  2. **Backend разработчик (Backend Developer)**. Должен уметь проектировать правильную структуру базы данных и создавать запросы на вставку, удаление, обновление и получение информации из базы, а также взаимодействовать с фронтенд-разработчиками с целью правильного отображения полученной информации конечному пользователю.

---

<sup>1</sup> Шилдс, У. SQL: быстрое погружение. СПб: Питер, 2022. С. 15.

### 3. Администратор баз данных (Database Administrator - DBA).

Отвечает за разработку требований к базе данных и ее использование. Администраторы баз данных играют ключевую роль при приобретении программных средств, а также несут ответственность за контроль доступа к базам данных.

- Существует огромное количество СУБД и их подвидов, но в данный момент глобально СУБД можно разделить на две части – реляционные и нереляционные системы. Реляционные системы не только хранят данные, но и позволяют контролировать связи (отношения) между сущностями, хранимыми в базе данных. Это означает, что система помогает разработчику структурировать данные для их правильного использования (чтобы данные случайно не были отнесены к чужой сущности, чтобы данные, которые уже не нужны, автоматически удалялись и т.д.). Кроме того, все реляционные СУБД поддерживают язык запросов SQL (хотя каждая конкретная СУБД может иметь некоторые отличия от стандарта – т.н. диалект), который позволяет взаимодействовать с хранимыми данными. Нереляционные СУБД не накладывают ограничения на хранимые данные, их главная задача – хранить что-угодно и это что-угодно быстро отдать при запросе. Можно понять, что реляционные СУБД предназначены для долговременного хранения важных данных, потеря которых крайне нежелательна, а нереляционные системы хорошо показывают себя при работе с неструктурированной информацией с небольшим сроком жизни.
- В рамках данного курса будет рассматриваться реляционная СУБД MySQL. Во-первых, реляционные СУБД заметно популярнее своих нереляционных коллег.<sup>2</sup> Во-вторых, сама MySQL является одной из самых популярных представительниц своего вида<sup>3</sup> – она давно зарекомендовала как надежное средство для всех типов проектов, является свободным программным обеспечением (хотя есть и платные дополнения), но одновременно поддерживается гигантом современной индустрии – компанией Oracle. В-третьих, MySQL обладает набором хороших дополнительных инструментов, который сильно облегчает работу с базами данных – главным из них является графическая оболочка MySQL Workbench, при помощи которой можно не только делать запросы, но также проектировать структуру базы, а также визуализировать связи между сущностями внутри базы.

---

<sup>2</sup> <https://learnsql.com/blog/most-popular-databases-2023>

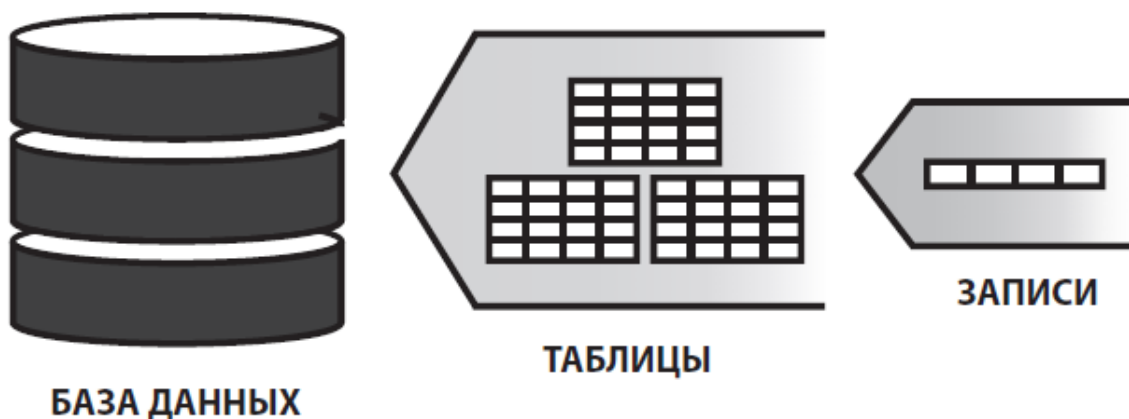
<sup>3</sup> Там же.

## ОБЩЕЕ ОПИСАНИЕ СУБД MYSQL

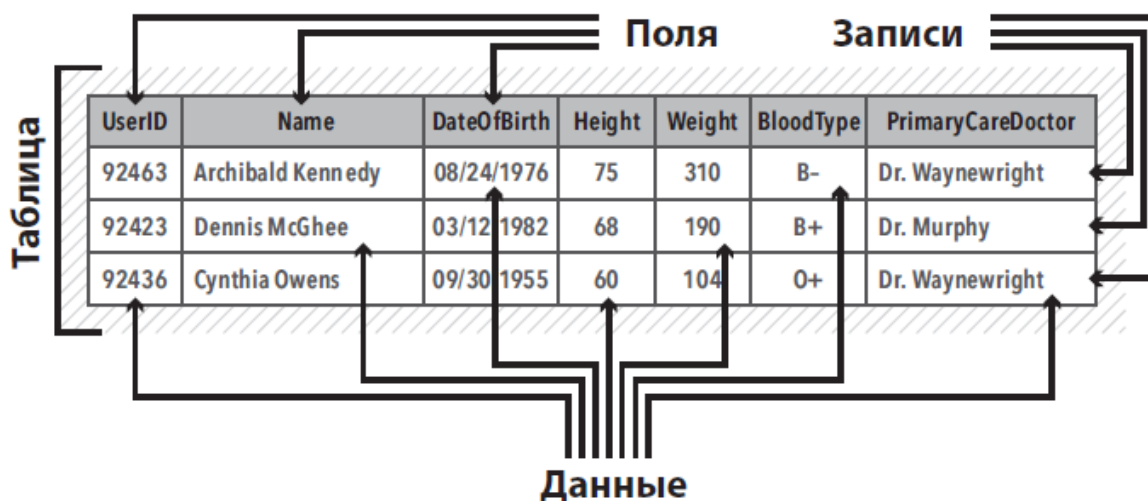
### ОСНОВЫ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ

- Перед тем, как обратиться непосредственно к MySQL, особое внимание стоит обратить на те принципы, которые лежат в основе этой СУБД – т.е. на принципы реляционных баз данных.
- Автором этих принципов является британский математик Эдгар Франк Кодд (1923 – 2003), который в 1970 году предложил и, что особо важно, математически обосновал (т.е. придал своей идее научный фундамент) ряд новых способов хранения информации, получивших название реляционная модель данных.
- Пока не все положения реляционной модели нам будут понятны, рассмотрим только основные из них:
  - данные хранятся в таблицах, состоящих из столбцов (также они могут называться полями, колонками или атрибутами) и строк (также называются записями, рядами или кортежами);
  - на пересечении каждого столбца и строки стоит только одно значение;
  - у каждого столбца есть своё имя, которое служит его названием, и все значения в одном столбце имеют один тип;
  - столбцы располагаются в определённом порядке, который определяется при создании таблицы, в отличие от строк, которые располагаются в произвольном порядке;
  - в таблице может не быть ни одной строчки, но обязательно должен быть хотя бы один столбец;
  - запросы к базе данных возвращают результат в виде таблиц, которые тоже могут выступать как объект запросов;
- Из приведенных выше положений Э.Ф. Кодда можно понять, что в реляционных базах данных (в том числе в MySQL) информация должна храниться в таблицах, каждая из которых заполняется строками (записями), в свою очередь все строки каждой конкретной таблицы должны иметь одинаковую структуру – т.е. набор столбцов в строго заданном порядке со строго

определенными типами данных. Само собой, значения данных могут быть различными – самое главное, чтобы их типы (число, строка, дата и т.д.) были одинаковыми.



Базы данных состоят из таблиц. Таблицы состоят из записей.<sup>4</sup>



Таблицы состоят из записей (строк). Записи состоят из полей (столбцов), которые расположены в строго заданном порядке и имеют одинаковые типы (*Name* – всегда строка, *Height* – всегда число, *DateOfBirth* – всегда дата и т.д.).<sup>5</sup>

<sup>4</sup> Шилдс, У. SQL: быстрое погружение. С. 22.

<sup>5</sup> Там же. С. 23.

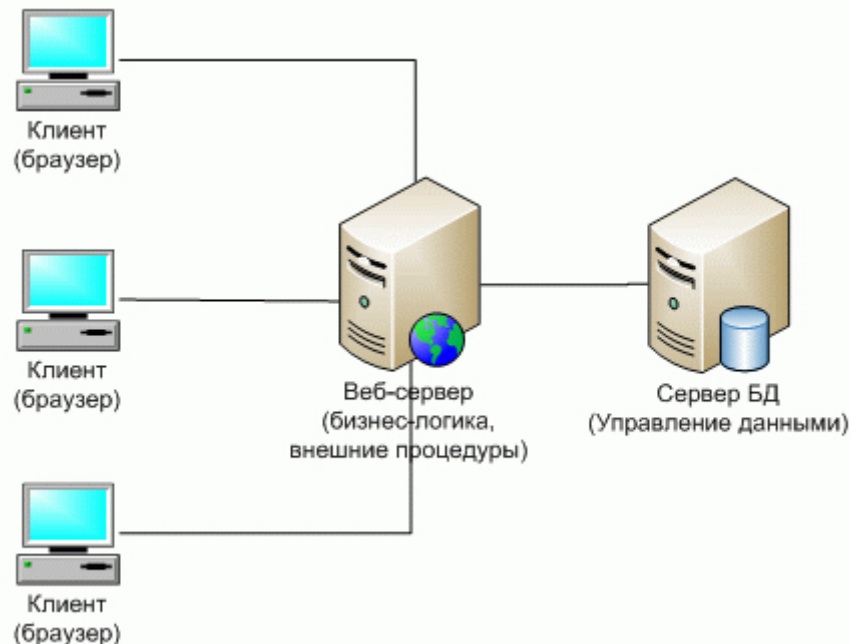
## **ИСТОРИЯ, ПРИМЕНЕНИЕ И ОСОБЕННОСТИ MYSQL**

- Первая версия СУБД MySQL была выпущена 23 мая 1995 года работниками шведской компании TCX DataKonsult для внутреннего использования. По одной из самых распространенных версий, название СУБД состоит из двух частей – My (имя дочери главного разработчика Микаэля Видениуса)<sup>6</sup> и SQL (Structured Query Language – структурный язык запросов, разные диалекты которого используются для работы с реляционными базами данных). Разработка оказалась удачной, и уже в 1996 году была создана компания MySQL AB, которая занималась исключительно MySQL и ее продвижением.
- Особенно большую популярность MySQL начала обретать с 2001 года, когда в версии 3.23 была внедрена (пока в качестве экспериментальной) технология InnoDB, которая позволяла использовать такие важнейшие инструменты реляционных баз данных как транзакции и ограничения внешнего ключа (в дальнейшем мы подробно разберем эти инструменты). Начиная с версии 4.0 (2003 год) InnoDB начала использоваться в MySQL уже полноценно.
- В 2005 году компания Oracle приобрела компанию Innobase, которой изначально принадлежала технология InnoDB, что вызвало беспокойство по поводу дальнейшего будущего MySQL, т.к. Oracle некоторое время рассматривала MySQL AB как своего конкурента. Однако в 2006 году было достигнуто соглашение, согласно которому MySQL AB было позволено продолжать использовать InnoDB.
- В 2008 году компания MySQL AB была продана компании Sun Microsystems. Однако уже через год (в 2009) уже сама компания Sun Microsystems была продана компании Oracle, которая с того времени и по сей день владеет СУБД MySQL. В данный момент MySQL является зрелой, стабильной и серьезной технологией, которая используется такими корпорациями как Netflix, Toyota, Uber, Walmart, Airbnb, PayPal и многими другими. MySQL Community Edition является абсолютно бесплатной и открытой для всех видов доработок версией данного продукта, в свою очередь MySQL Enterprise Edition, техническую поддержку и сопровождение которой осуществляет Oracle, распространяется платно.

---

<sup>6</sup> <https://dev.mysql.com/doc/refman/8.0/en/history.html>

- В глобальном масштабе СУБД MySQL является типичным примером программы, которая использует клиент-серверную архитектуру. Это означает, что приложение, которое использует MySQL, не обязано находиться на том же компьютере, где установлена сама MySQL. В общем случае СУБД установлена на одном компьютере (как правило, она использует порт 3306 для прослушивания входящих запросов), который называется сервером, а приложение развернуто на другом компьютере (клиенте), который соединен с первым компьютером сетью.
- Приложение, написанное на некоем языке программирования, имеет в своей структуре определенный набор драйверов или библиотек, которые позволяют языку программирования отправлять запросы (на получение или изменение данных) к MySQL серверу и получать от него ответы в том формате, который может быть обработан в контексте приложения.
- Однако само приложение и СУБД могут быть расположены на одном и том же компьютере. Это не слишком хорошо с точки зрения безопасности, но в целях экономии средств часто используется на небольших и даже средних проектах.



**Место сервера с базой данных в правильной клиент-серверной архитектуре (в качестве приложения выступает сайт).**



- На внутреннем уровне архитектура СУБД MySQL также весьма любопытна. Таблицы баз данных обычно хранятся в файлах, но также есть способ хранить информацию и в памяти компьютера. Однако самое важное здесь понимание того, что MySQL позволяет использовать разные способы хранения и структурирования данных (т.н. движки) даже если эти данные хранятся исключительно в файлах или исключительно в памяти. Понятно, что будет меняться скорость вставки и получения данных, но также может меняться тот функционал, который доступен при работе с таблицами.
- Все движки MySQL будут подробно рассмотрены в следующих разделах курса, однако для примера можно сказать, что при использовании движка MyISAM пользователю СУБД не будут доступны т.н. ограничения по внешнему ключу (т.е. в определенные столбцы таблицы можно вставлять только те данные, которые существуют в определенных столбцах другой таблицы). Такие ограничения возможны только при использовании уже упоминавшегося движка InnoDB. Если пользователь СУБД забудет о версии движка, то может быть нарушена логика работы всей базы данных – при запросах в приложение будут приходить неправильные данные.
- Особую роль информация из предыдущего пункта начинает играть в том случае, если приходится использовать СУБД MySQL ниже версии 5.5 (хотя данная версия и вышла в 2010 году, иногда с ней по-прежнему можно столкнуться). Именно начиная с этой версии движком по умолчанию стал InnoDB, а до этого в таком качестве использовался MyISAM.<sup>7</sup> Таким образом, создавалась ситуация, при которой по умолчанию был отключен важный функционал, оказывающий влияние на правильность взаимодействия с данными.
- К другим особенностям устройства СУБД MySQL можно отнести максимальное теоретическое количество столбцов в одной таблице – 4096 (хотя фактическое количество может быть меньше). Количество строк в таблицах обычно ограничено не внутренними настройками СУБД, а лимитами операционной системы – однако есть ряд рекомендаций, которым лучше следовать (например, разделять таблицы, если их размер превысил 1 ТБ).<sup>8</sup>

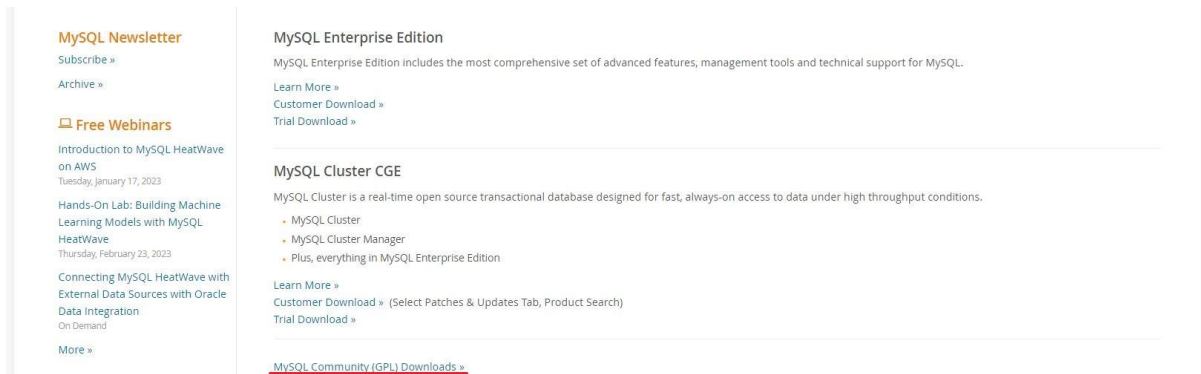
---

<sup>7</sup> Шварц, Б., Зайцев, П., Ткаченко, В. MySQL по максимуму. Оптимизация, репликация, резервное копирование. СПб: Питер, 2018. С. 61.

<sup>8</sup> <https://dev.mysql.com/doc/mysql-reslimits-excerpt/8.0/en/limits.html>

## УСТАНОВКА MYSQL

- Чтобы начать работать с MySQL, надо сначала установить эту СУБД на свой компьютер. Для этого следует перейти по ссылке <https://www.mysql.com/downloads/> и внизу страницы нажать на надпись **"MySQL Community (GPL) Downloads"**:



- На следующей странице необходимо выбрать тот продукт, который нам нужен в данный момент. Если у нас на компьютере установлена операционная система Windows, выбираем пункт **"MySQL Installer for Windows"**:



- После очередного перехода попадаем на страницу, где нам предстоит выбрать из 2 вариантов – выбираем нижний из них (тот, который, имеет наибольший размер и файл которого называется **mysql-installer-community-8.0.31.0.msi**) :

## MySQL Community Downloads

MySQL Installer

**MySQL Installer 8.0.31**

Select Operating System: Microsoft Windows Looking for previous GA versions?

Operating System	Version	Size	Action
Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.31.0.msi)	8.0.31	5.5M	<a href="#">Download</a>
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.31.0.msi)	8.0.31	431.7M	<a href="#">Download</a>

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

ORACLE © 2022 Oracle

[Privacy / Do Not Sell My Info](#) | [Terms of Use](#) | [Trademark Policy](#) | [Cookie Preferences](#)

- На последней странице игнорируем предложение о регистрации или входе в систему и сразу же начинаем скачивать MySQL, нажав на надпись **"No thanks, just start my download."** :

## MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

[Login »](#)  
using my Oracle Web account

[Sign Up »](#)  
for an Oracle Web account

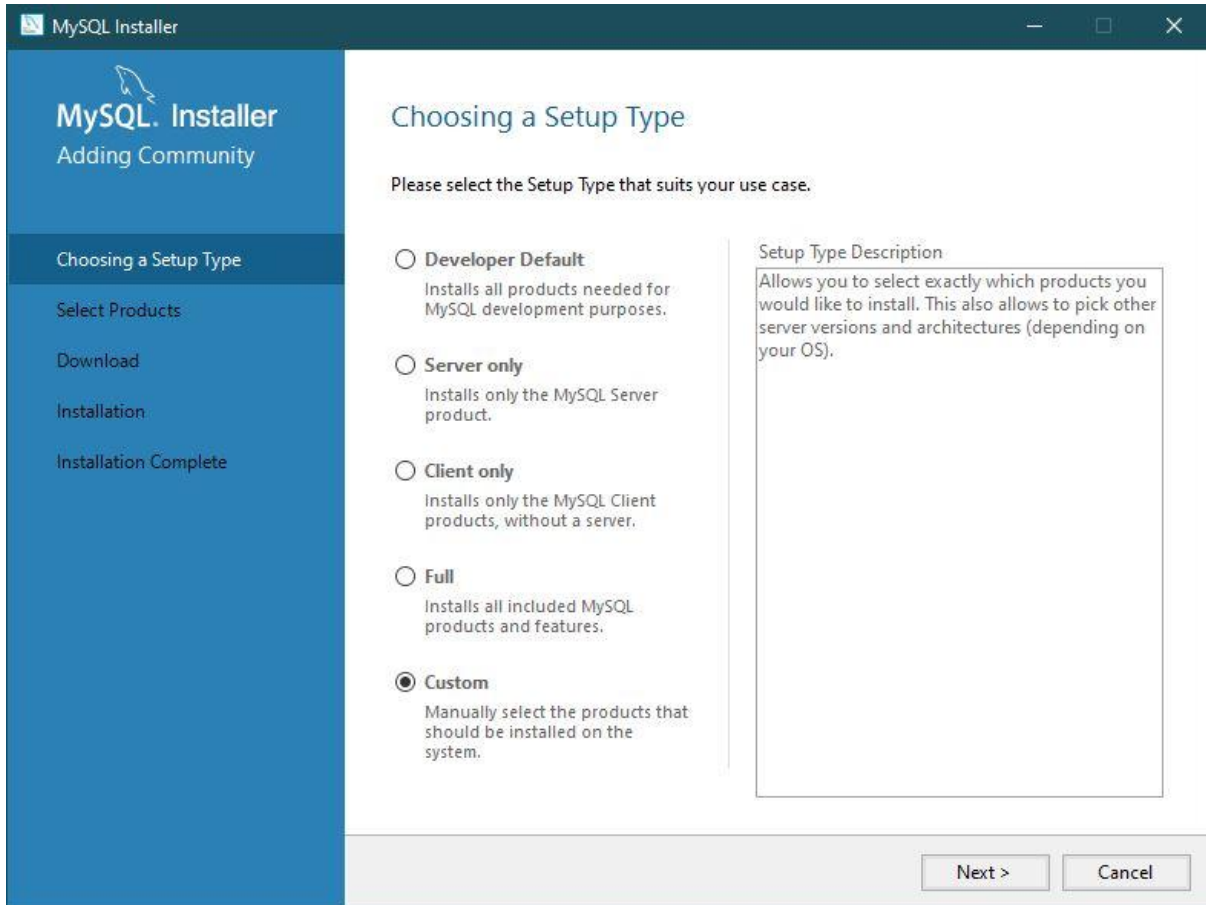
MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can sign up for a free account by clicking the Sign Up link and following the instructions.

[No thanks, just start my download.](#)

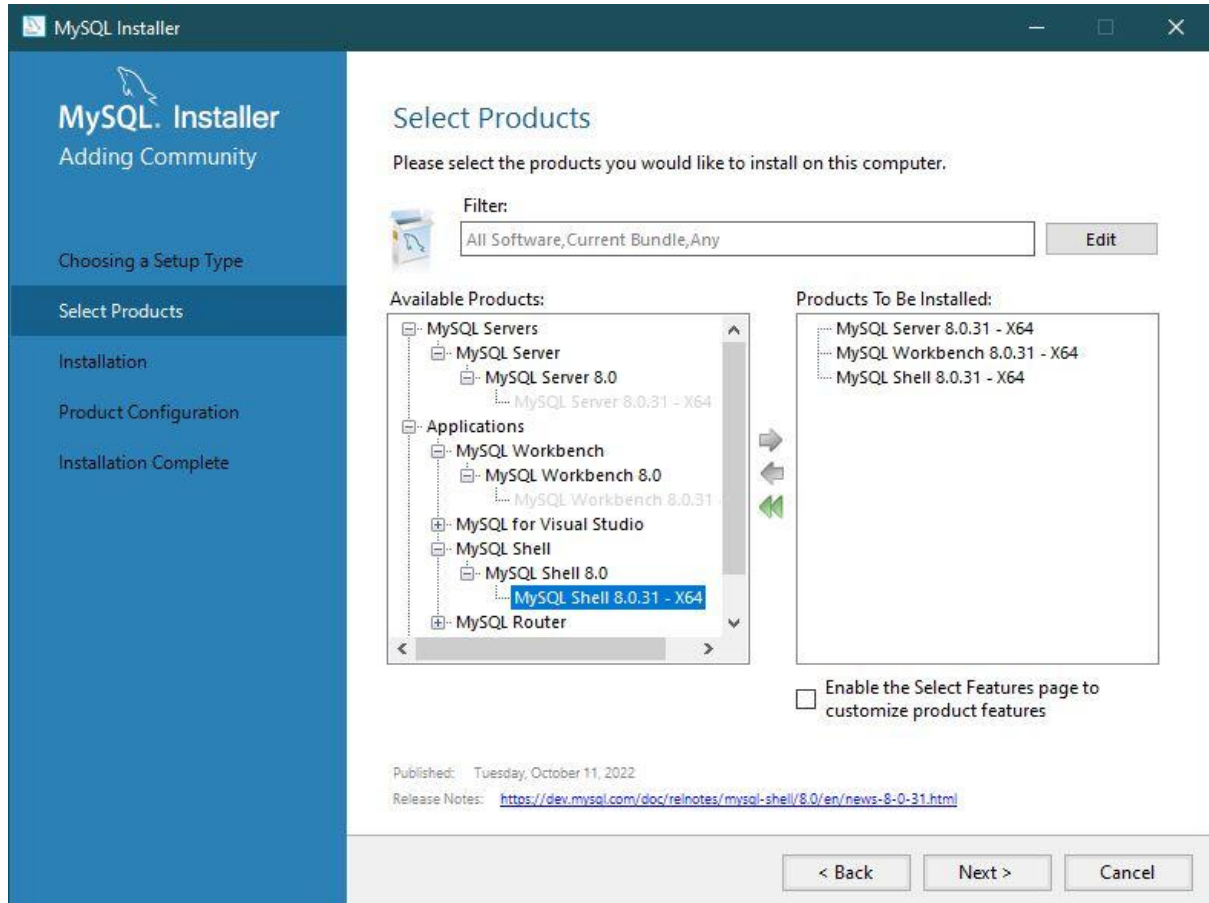
ORACLE © 2022 Oracle

[Privacy / Do Not Sell My Info](#) | [Terms of Use](#) | [Trademark Policy](#) | [Cookie Preferences](#)

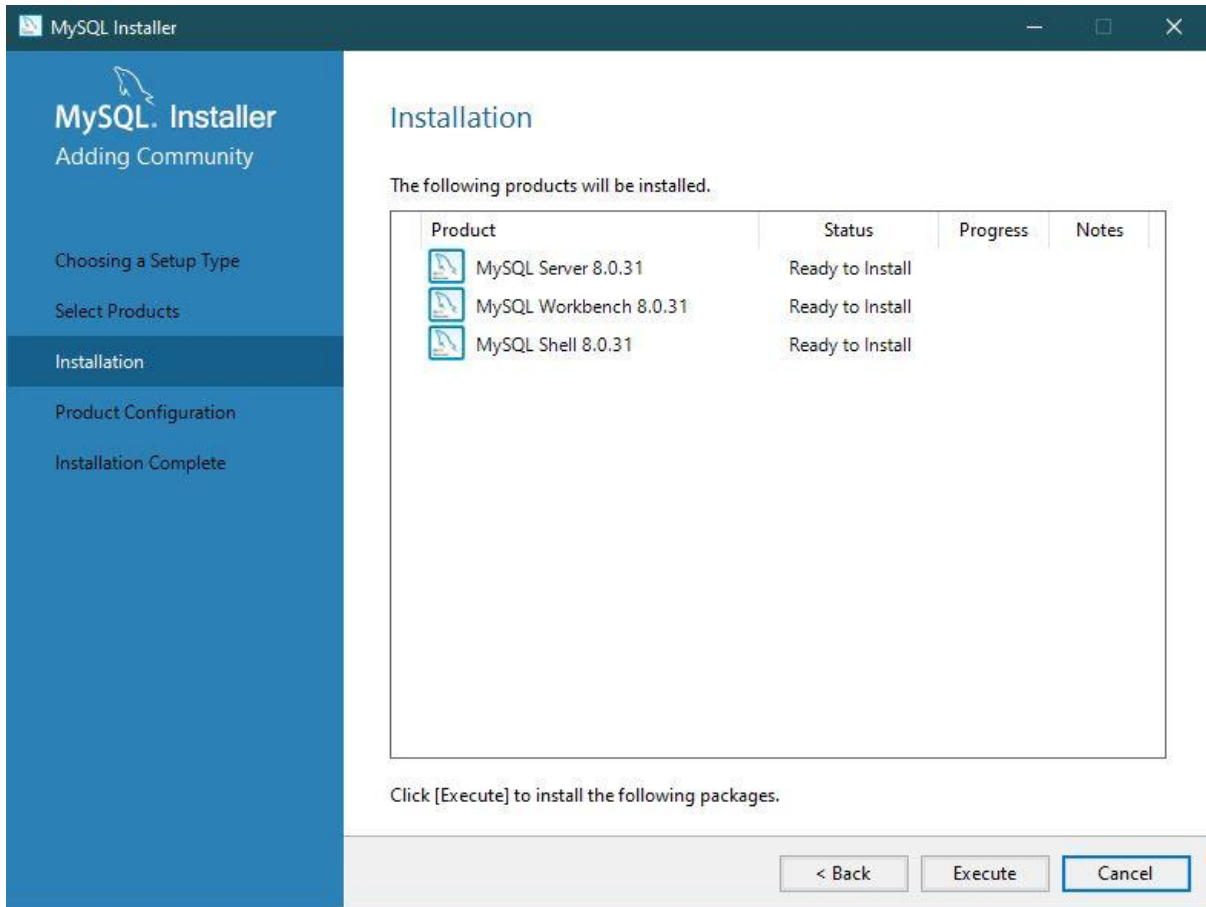
- После того, как инсталляционный файл был скачан, сделаем на нем двойной клик – подождем пока откроется первый шаг инсталляции, после этого в качестве типа установки выбираем “Custom”, а затем нажимаем кнопку “Next”:



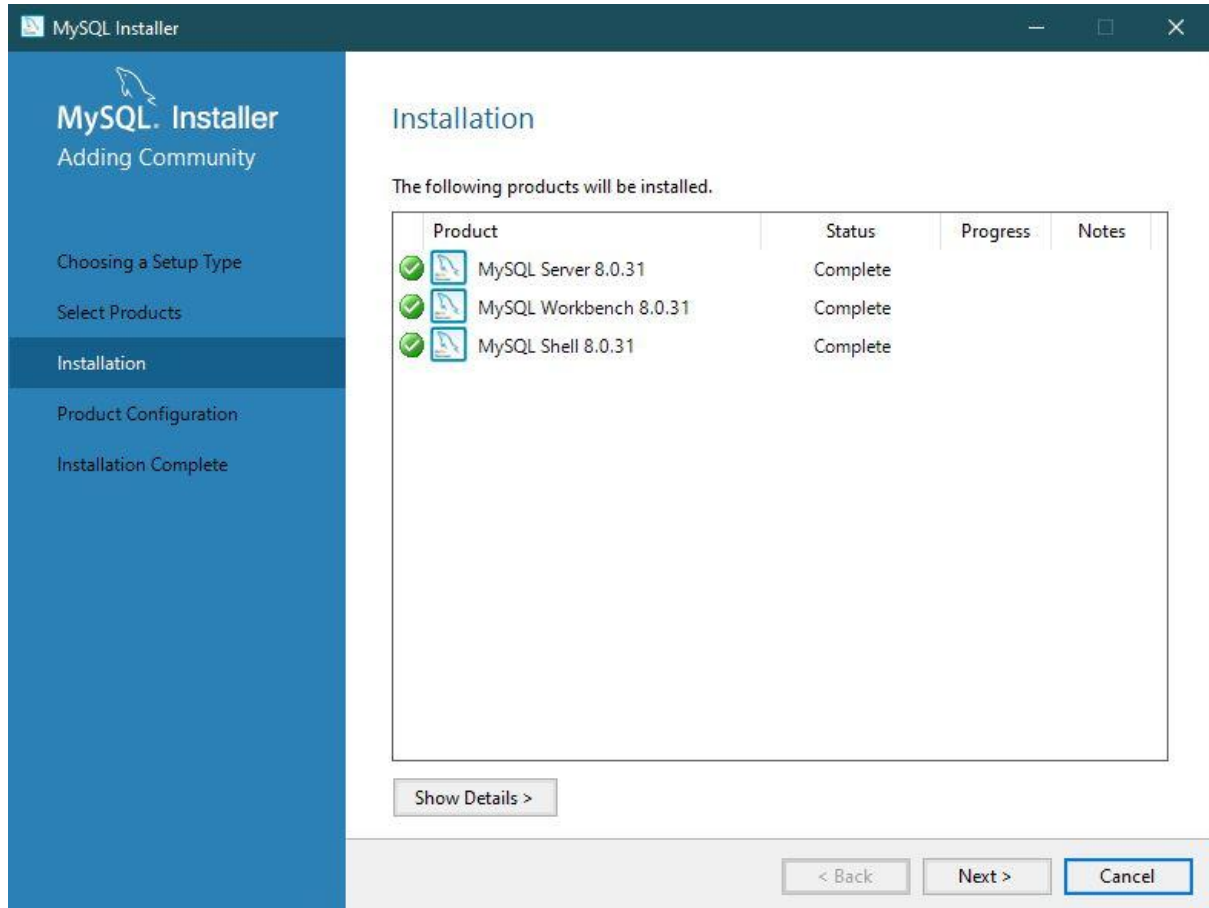
- На следующем шаге выберем только необходимые для нас компоненты – это MySQL Server, MySQL Workbench и MySQL Shell (если они не видны, следует раскрыть дерево с продуктами). Затем также нажимаем кнопку “Next”:



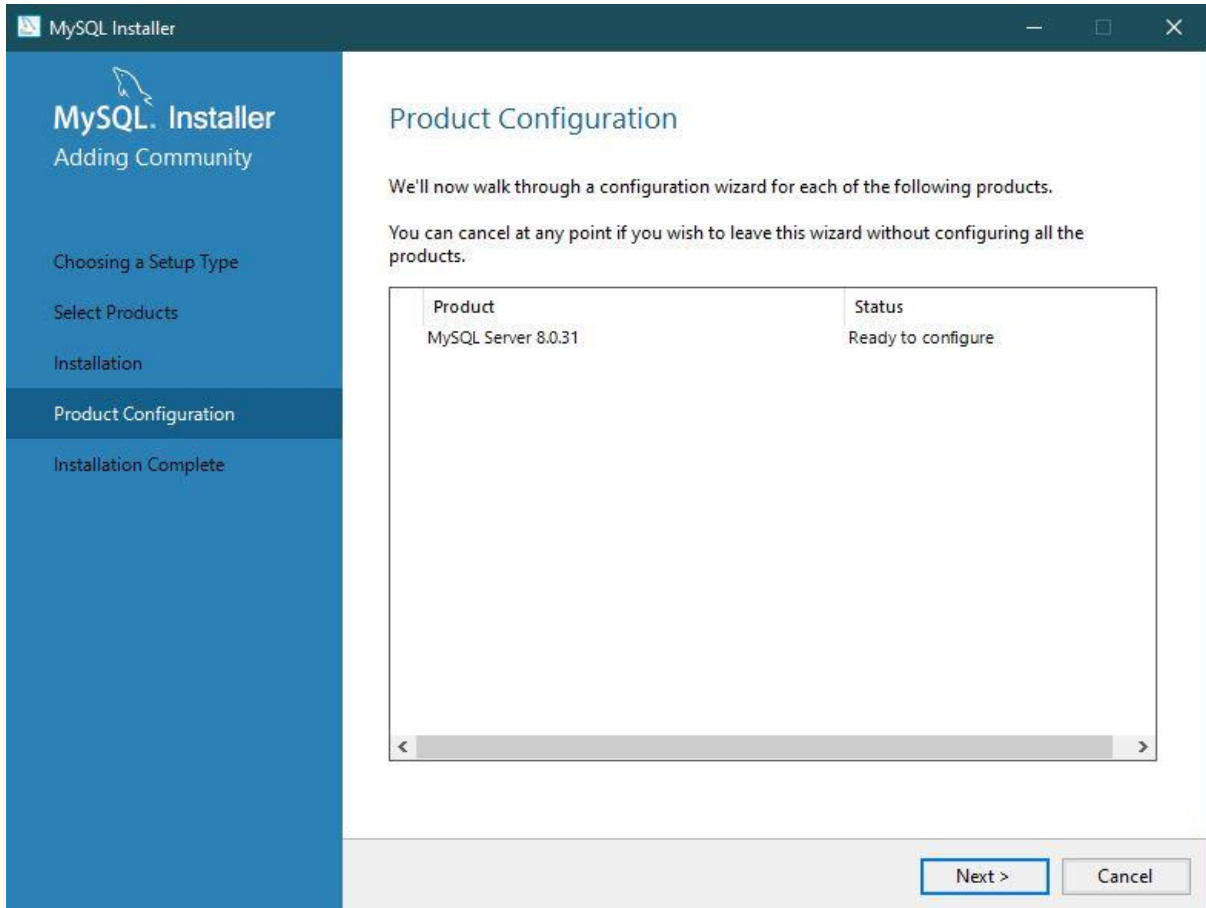
- На следующем шаге нажимаем кнопку “Execute” и ожидаем завершения установки всех выбранных компонентов:



- После успешной установки все компоненты получают статус "Complete". После того, как это произойдет, следует опять нажать кнопку "Next":

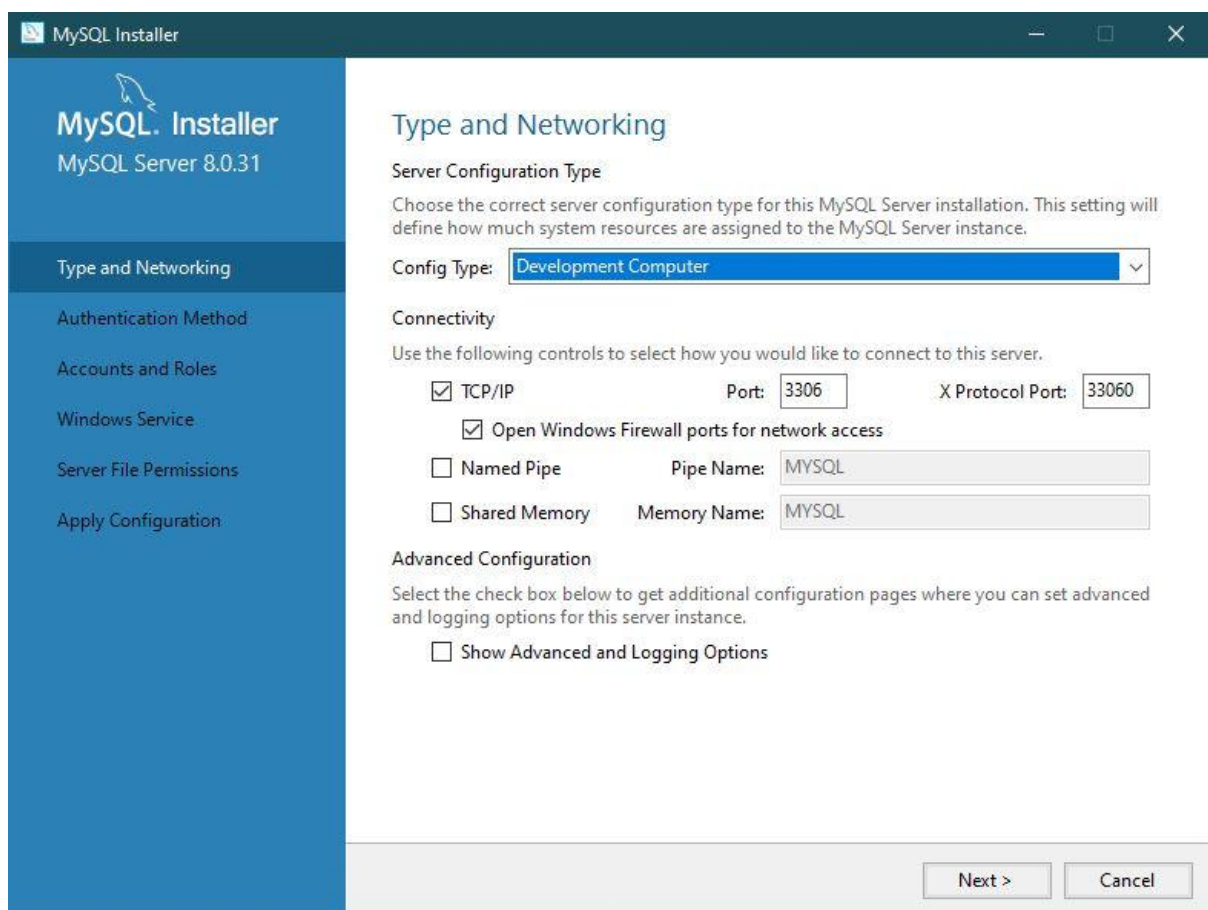


- На следующей странице показаны те компоненты, которым требуется дополнительная настройка (в данном случае это будет только MySQL Server). В очередной раз нажимаем кнопку "Next":

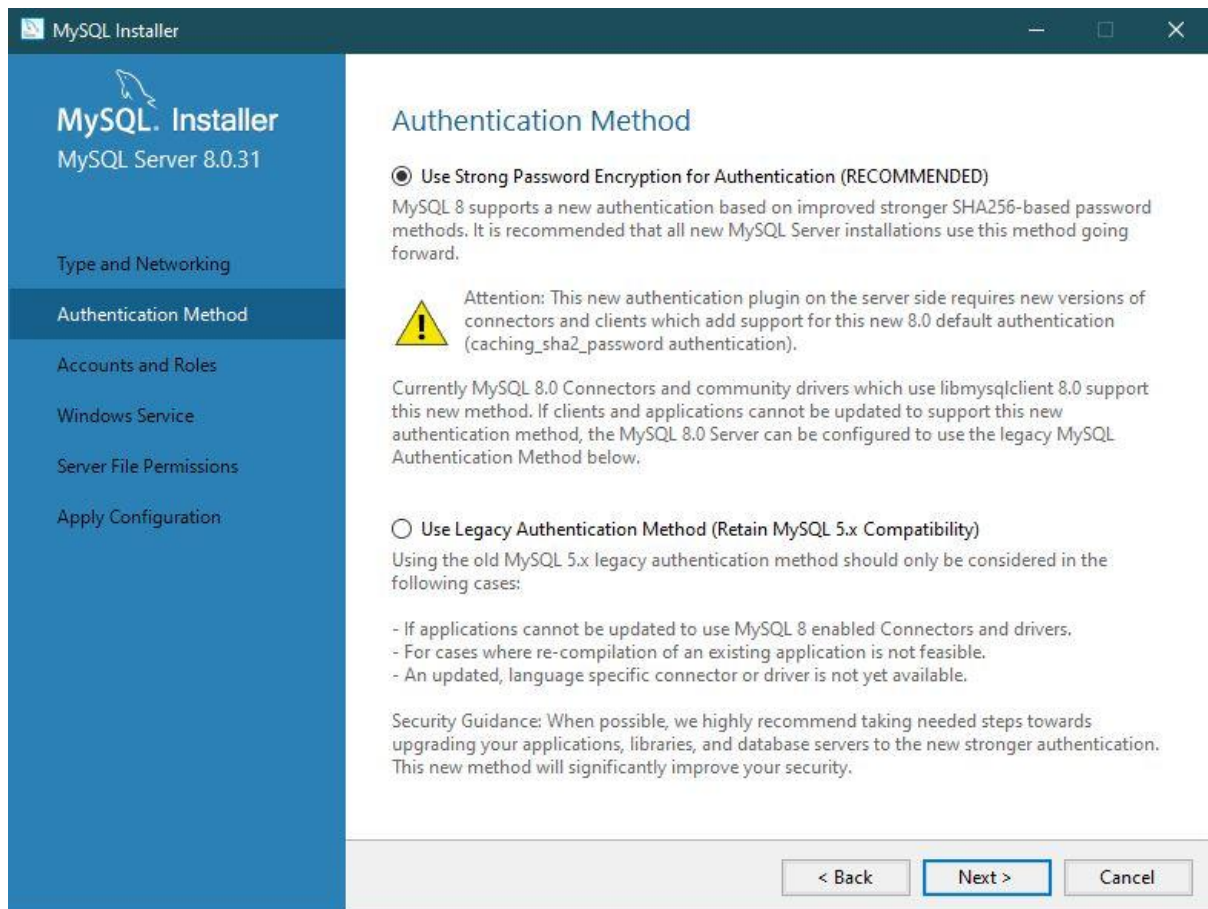




- В открывшемся окне убедимся, что настройки отрегулированы следующим образом:
  - **Config Type** - Development Computer
  - **TCP/IP** - включен (будем общаться с MySQL по сети)
  - **Port** - 3306
  - **X Protocol Port** - 33060 (для MySQL Shell и т.п.)
  - **Open Windows Firewall ...** - включен
  - Все остальные опции должны быть выключены
- После всех манипуляций нажимаем кнопку "Next":



- На следующей странице выбираем сильное шифрование паролей (**Use Strong Password Encryption for Authentication**) и опять нажимаем кнопку "Next":



- На следующей странице необходимо ввести пароль (и его подтверждение) для основного пользователя (также известен как **root**) – условимся выбрать слово **secret** в качестве пароля. После заполнения двух полей (пароля и его подтверждения) нажимаем кнопку "Next":

**MySQL Installer**  
MySQL Server 8.0.31

Type and Networking  
Authentication Method  
**Accounts and Roles**  
Windows Service  
Server File Permissions  
Apply Configuration

### Accounts and Roles

**Root Account Password**  
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password strength: **Weak**

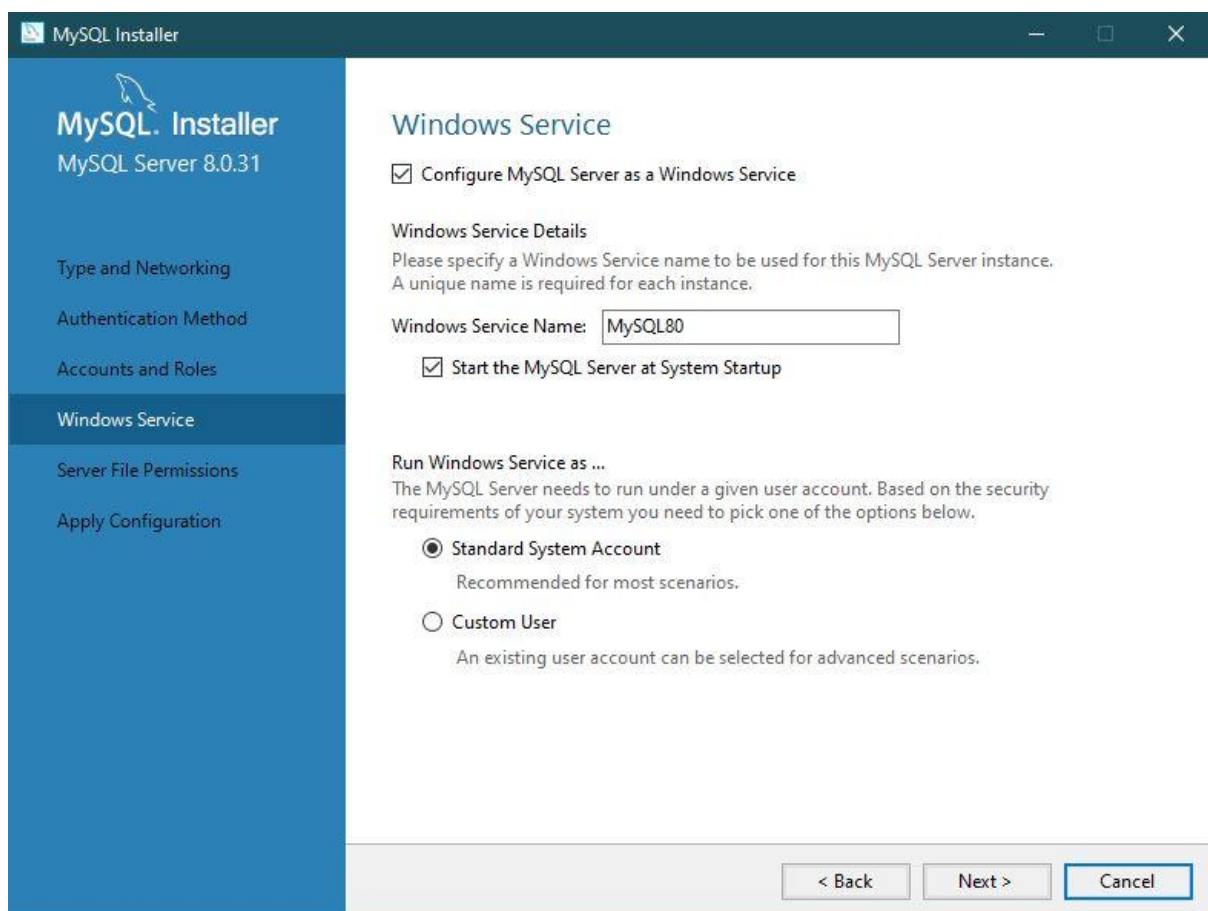
**MySQL User Accounts**  
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role
-----------------	------	-----------

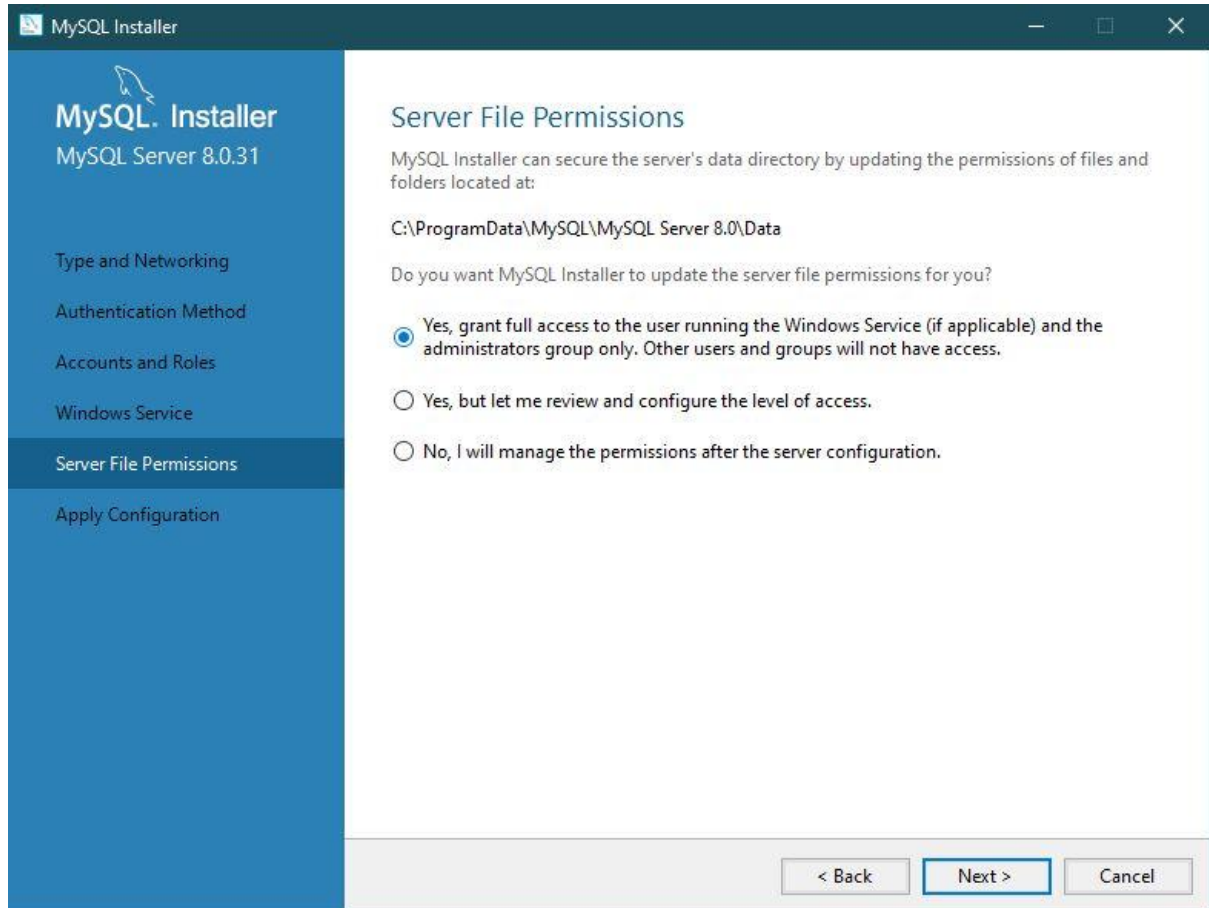
Add User  
Edit User  
Delete

< Back   Next >   Cancel

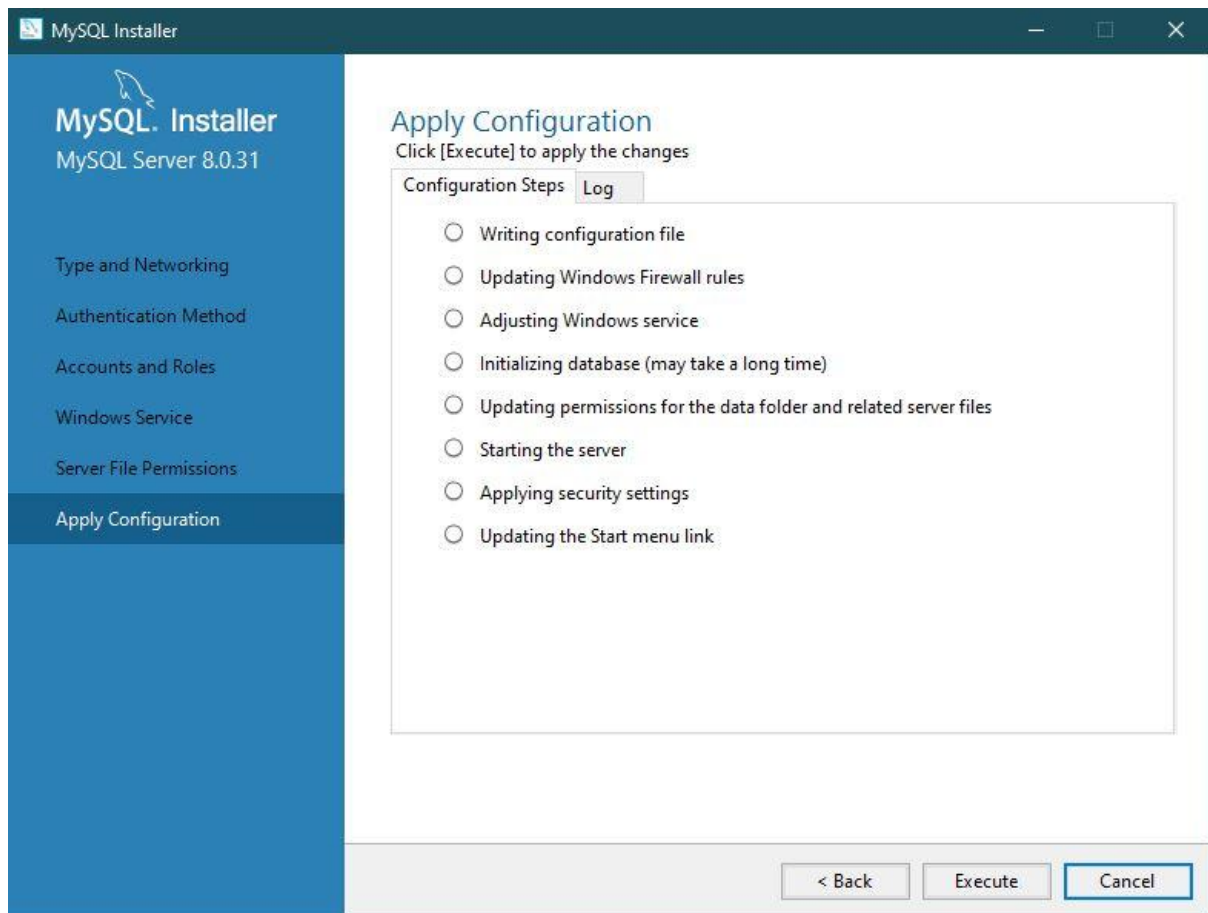
- На этом шаге мы должны настроить режим работы MySQL. Отрегулируем настройки следующим образом:
  - **Configure MySQL Server as a Windows Service** - включен
  - **Windows Service Name** - MySQL80
  - **Start the MySQL Server at System Startup** - по желанию
  - **Run Windows Service as** - Standard System Account
- После всех манипуляций нажимаем кнопку "Next":



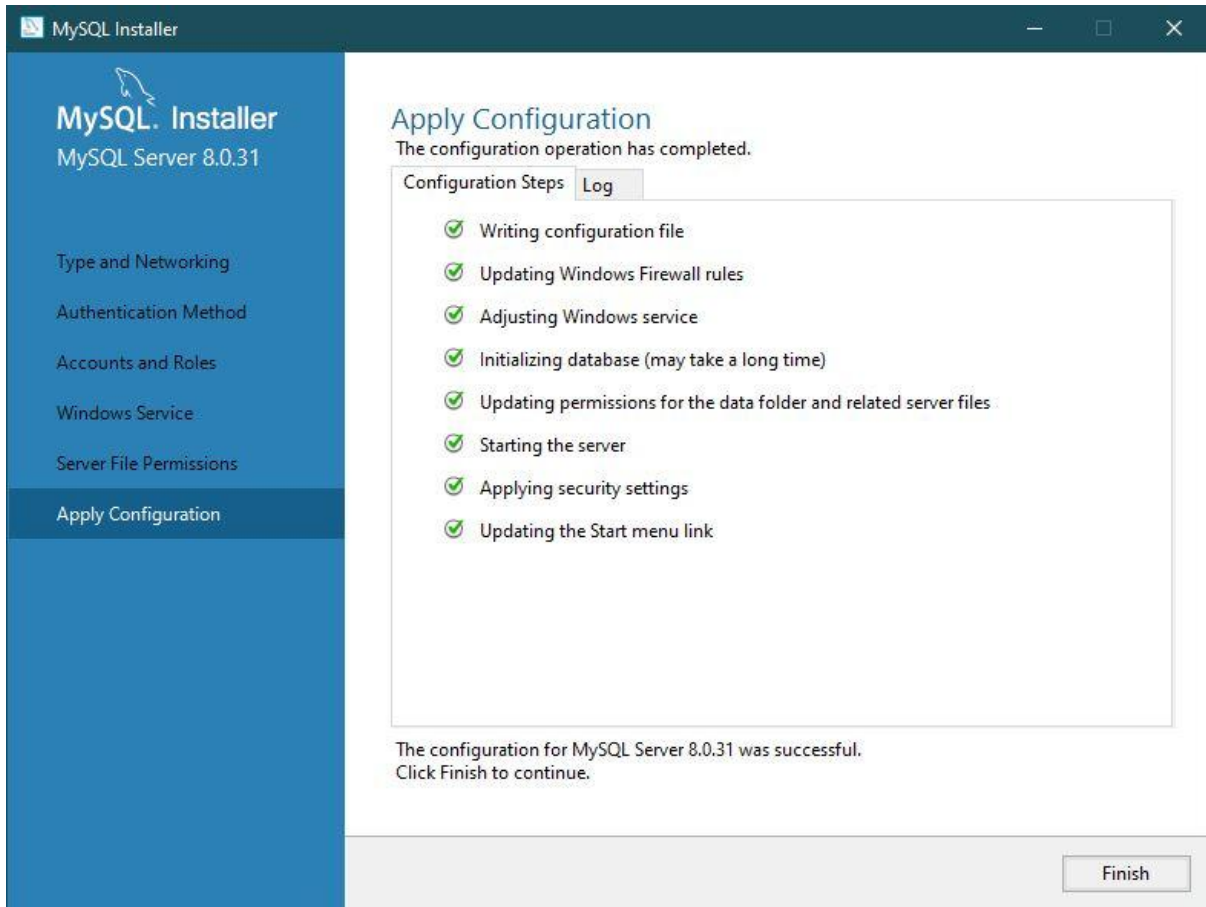
- На следующей странице настроим права на файлы, используемые в контексте MySQL – выберем первый вариант **Yes, grant full access to the user running Windows Service** ... . После этого нажимаем кнопку "Next":



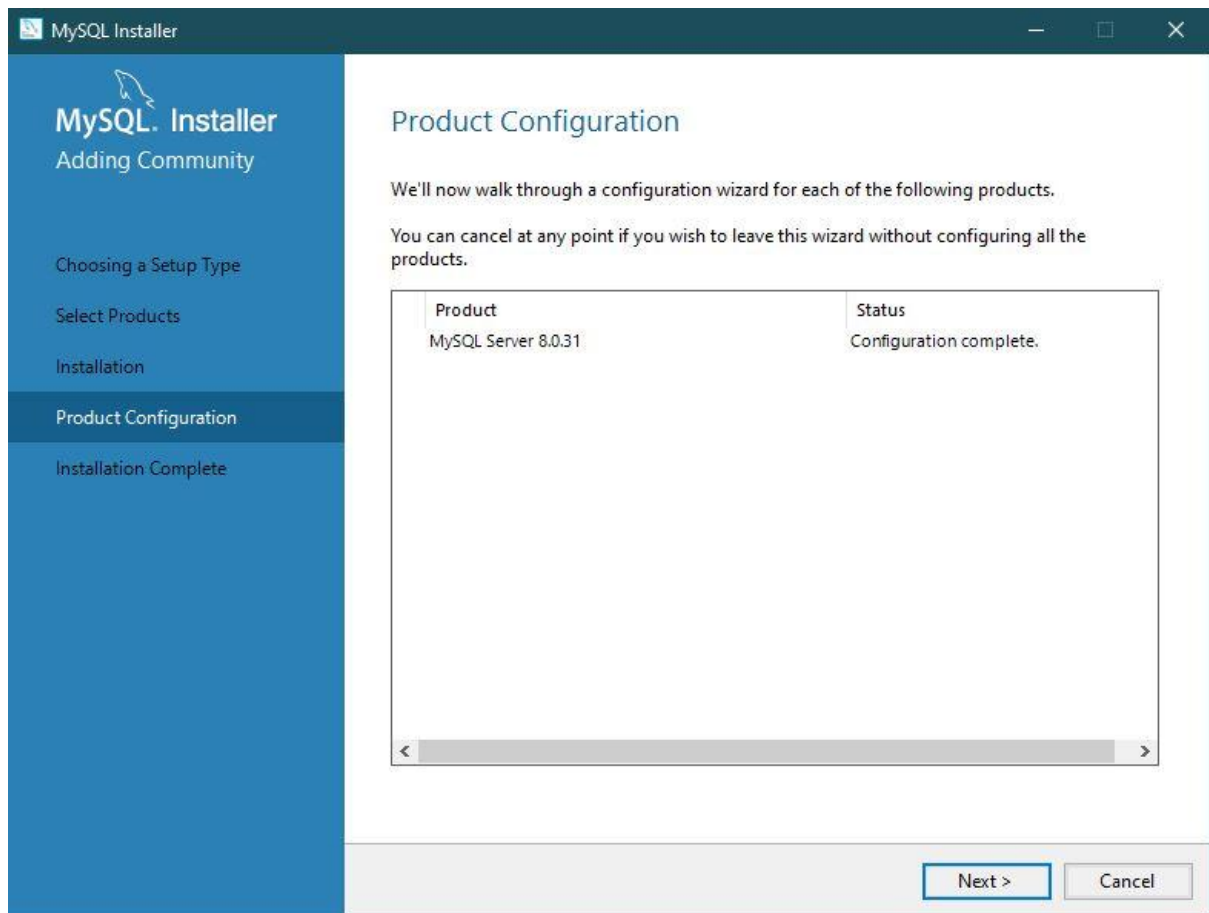
- На следующем шаге нас попросят подтвердить все выбранные ранее конфигурации – если нас все устраивает (а нас все устраивает), нажимаем кнопку “Execute”:



- После того, как все настройки будут успешно применены, нажимаем кнопку "Finish":

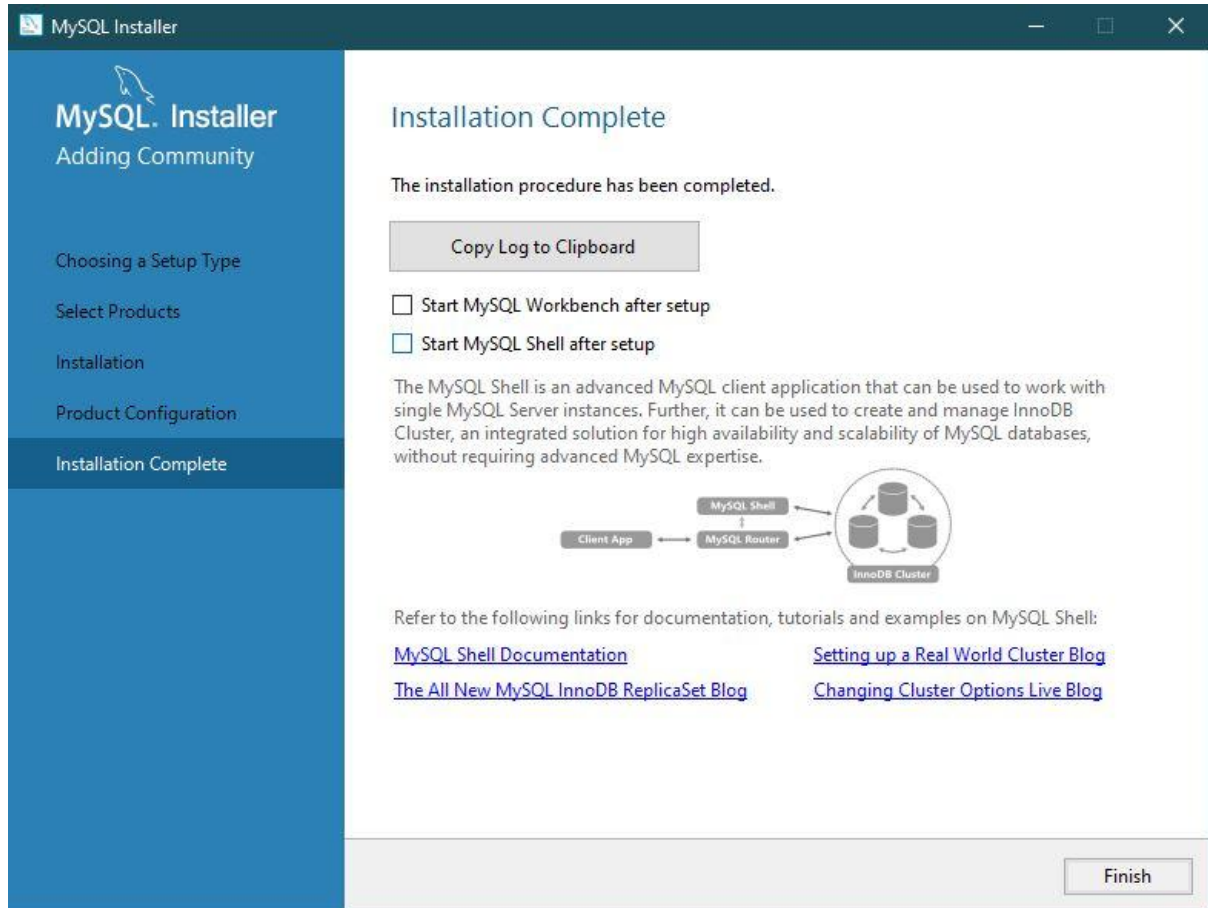


- На следующем шаге мы увидим, что MySQL успешно сконфигурирован. Порадуемся этому факту, а затем в очередной раз нажмем кнопку "Next":





- На финальном шаге уберем галочки с настроек **Start MySQL Workbench after setup** и **Start MySQL Shell after setup** (мы запустим их сами), и нажмем кнопку "Finish" – на этом установка MySQL и сопутствующих программ будет завершена:



## ОБЗОР УТИЛИТ И КОНФИГУРАЦИИ MYSQL

- При установке на операционную систему Windows и выборе путей по умолчанию, основные файлы MySQL будут находиться в директории **C:\Program Files\MySQL\MySQL Server 8.0\bin** – далее рассмотрим подробно расположенные там утилиты.
- Сама MySQL запускается при помощи файла **mysqld.exe** – если при установке нашей СУБД был выбран вариант установки в качестве сервиса операционной системы, то в контексте сервиса на фоне запускается именно этот файл.
- Если попытаться запустить **mysqld.exe** без всяких дополнительных настроек, то произойдет ошибка – программа попытается найти некие директории и настройки, которых не существует. Поэтому при запуске файлу **mysqld.exe** в качестве параметра с именем **--defaults-file** принято передавать путь к файлу с настройками по умолчанию (позже мы поговорим об этом файле подробнее), например вот так:  
**.\mysqld.exe --defaults-file="C:\ProgramData\MySQL\MySQL Server 8.0\my.ini"**
- Особый интерес представляет файл **mysqldump.exe** – при помощи данного файла можно делать резервные копии баз данных, отдельных таблиц, игнорировать таблицы и т.д. – в следующих разделах курса резервное копирование будет рассмотрено гораздо подробнее.
- Также интересен файл **mysql.exe** – он позволяет запустить так называемый клиент командной строки, т.е. программу, которая предоставляет возможность взаимодействовать с СУБД MySQL при помощи командной строки. Чтобы все это заработало, при запуске необходимо указать логин и пароль (предположим, что логин **root**, а пароль **secret**):  
**.\mysql.exe -uroot -psecret**
- Если запустить **mysql.exe** без дополнительных параметров, то ошибок не будет, однако этот файл может подхватить некоторые настройки операционной системы, которые негативно скажутся на работе с данными. Например, при неудачной кодировке командной строки на экране при выборке информации могут отобразиться не буквы и цифры, а непонятные иероглифы. Чтобы этого избежать, при запуске

необходимо прописать кодировку работы клиента с базой данных (в какой кодировке вставляем и получаем данные), а также файл дополнительных настроек, которые будут способствовать правильному функционированию клиента:

```
.\mysql.exe --defaults-file="C:\ProgramData\MySQL\MySQL
Server 8.0\my.ini" --default-character-set=utf8mb4 -uroot
-psecret
```

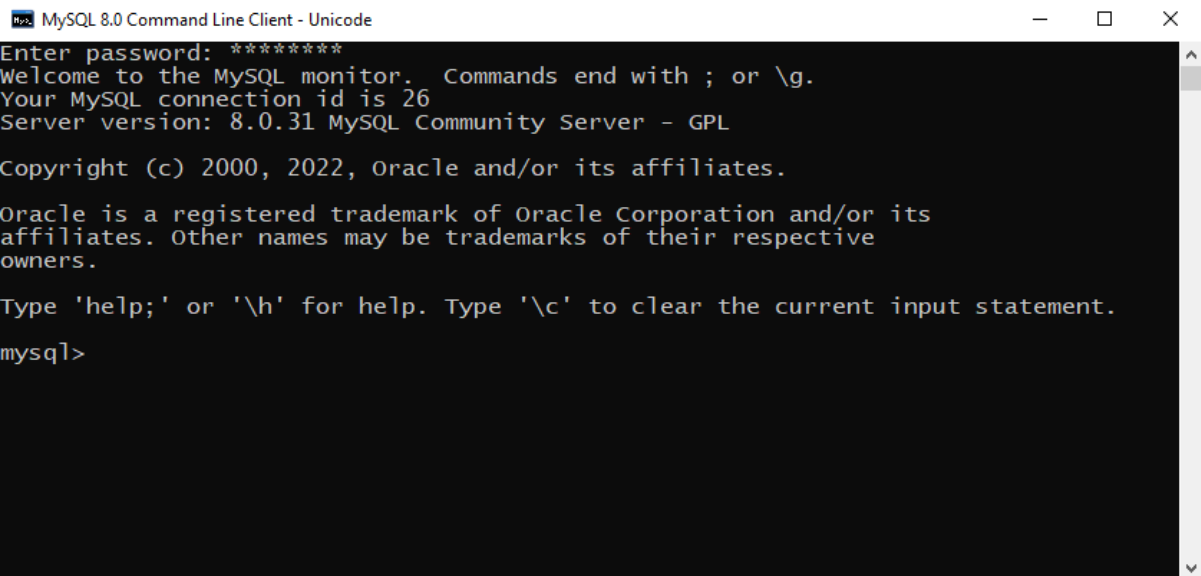
- Можно обратить внимание, что мы уже второй раз обращаемся к таинственному файлу **my.ini** (мы это уже делали, когда запускали саму СУБД). Во-первых, стоит сказать, что **my.ini** находится в директории "**C:\ProgramData\MySQL\MySQL Server 8.0**", которую MySQL (в рамках операционной системы Windows) по умолчанию использует для хранения баз данных и таблиц (они размещены в папке **Data**). Во-вторых, **my.ini** как раз является главным конфигурационным файлом, который включает в себя настройки как для самой MySQL, так и настройки для тех клиентских программ, которые будут работать с MySQL. Например, если в секции **[mysql]** вставить настройку **default-character-set=utf8mb4**, то при старте **mysql.exe** отпадает необходимость прописывать параметр **default-character-set** – он будет взят из файла с настройками.
- Единственным исключением из настроек в версии MySQL 8.0 является кодировка, в которой создаются базы данных и таблицы – даже если нигде ничего не будет указано явно, сама СУБД будет создавать свои сущности в кодировке **utf8mb4**.<sup>9</sup> Это поведение можно изменить, если при создании баз данных и таблиц явно указывать кодировку, кроме того, есть возможность в файле **my.ini** в секции **[mysqld]** вставить настройку **character-set-server** с нужной кодировкой, например: **character-set-server=latin1**
- После всего описанного выше может возникнуть вопрос – есть ли возможность подгружать конфигурационный файл **my.ini** автоматически – т.е. без прописывания каждый раз дополнительных параметров. Ответ – да. Есть несколько директорий, которые MySQL сканирует при старте на предмет наличия конфигурационного файла. Чтобы упростить себе жизнь, легче всего файл **my.ini** разместить в той же директории где установлена сама MySQL – **C:\Program Files\MySQL\MySQL Server 8.0**

<sup>9</sup> <https://dev.mysql.com/doc/refman/8.0/en/charset.html>

## НАЧАЛО РАБОТЫ С MYSQL

### КЛИЕНТ КОМАНДНОЙ СТРОКИ

- Классический вариант работы с MySQL – использование клиента командной строки. Можно запустить его воспользовавшись инструкцией из прошлой главы, однако есть более легкий способ (если мы работаем на Windows) – надо всего лишь найти среди установленных программ ярлык **MySQL 8.0 Command Line Client - Unicode** и нажать на него. Затем в открывшемся окне следует ввести пароль **secret** – в результате мы окажемся внутри MySQL.



```
MySQL 8.0 Command Line Client - Unicode
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 26
Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

- Даже если не учитывать работу с базами данных MySQL имеет множество полезных команд, которые сильно облегчают жизнь пользователю этой СУБД, например:
  - **\s** – полная информация о MySQL сервере
  - **\q** – отключение от mysql сервера
  - **\c** – отмена еще недописанной до конца команды
  - **\T** имя-файла – включение журналирования всех запросов
  - **\t** выключения журналирования
- Полный список команд можно получить выполнив команду **\h**

- Однако самая главная возможность – работать с базами данных. Создадим нашу первую базу выполнив следующий запрос (в данном случае вместо слова **DATABASE** можно использовать **SCHEMA** – в MySQL эти два термина являются псевдонимами) :

```
CREATE DATABASE first;
```

- Затем проверим то, что база была действительно создана – для этого выполним вот это (в полученном списке должна находиться база данных first):

```
SHOW DATABASES;
```

- Если все прошло успешно – поздравляю! Вы создали свою первую базу данных.
- Если мы хотим создать базу данных только в том случае, если ее нет, то запрос должен выглядеть следующим образом:

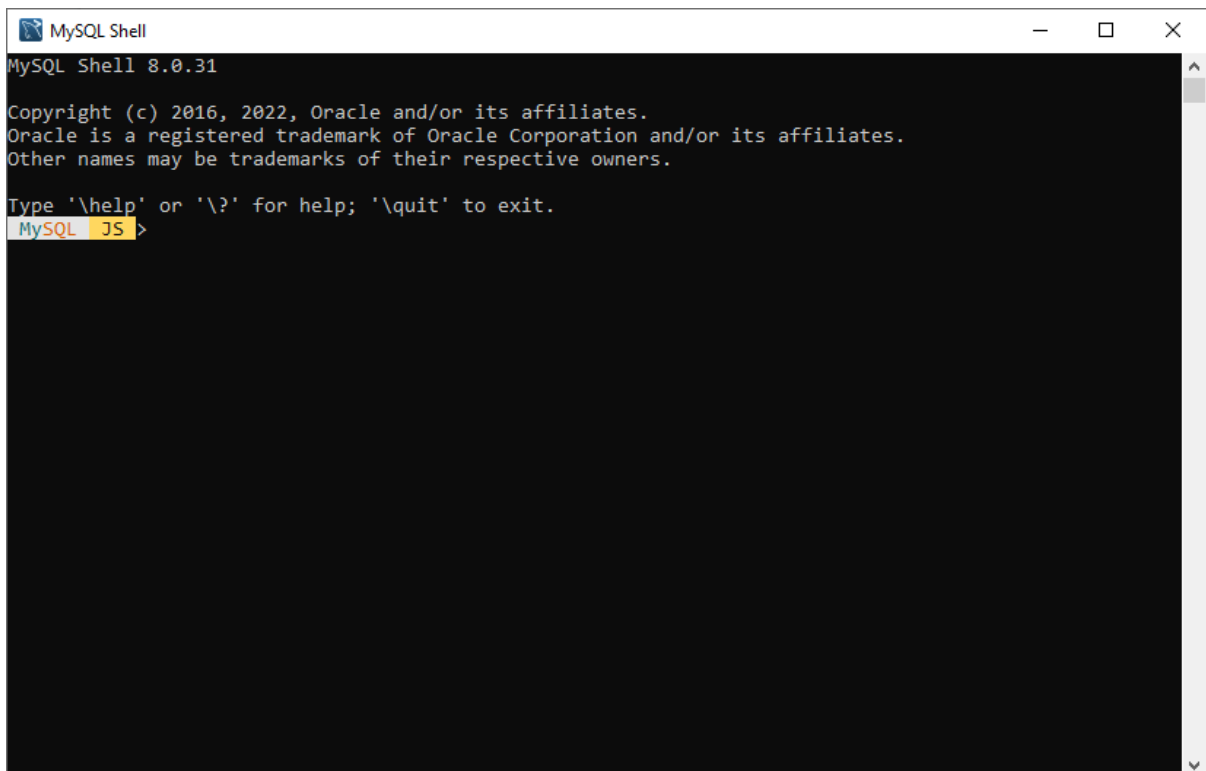
```
CREATE DATABASE IF NOT EXISTS first;
```

- Если мы хотим, чтобы база данных была создана не в кодировке по умолчанию, а в какой-либо другой кодировке, то полная команда будет выглядеть так (применим кодировку latin1):

```
CREATE DATABASE IF NOT EXISTS first DEFAULT CHARSET=latin1;
```

## MYSQL SHELL

- Существует более продвинутый вариант клиента командной строки – MySQL Shell. Данный вид взаимодействия с MySQL не только поддерживает команды SQL, но и позволяет применять такие языки программирования как Python и JavaScript для работы с базами данных. Более того, именно в данном режиме можно на примере посмотреть NoSQL возможности MySQL (в дальнейшем мы углубимся в эту тему).
- Для подключения найдем в списке установленных программ найдем ярлык MySQL Shell и нажмем на него. В результате увидим следующее окно:

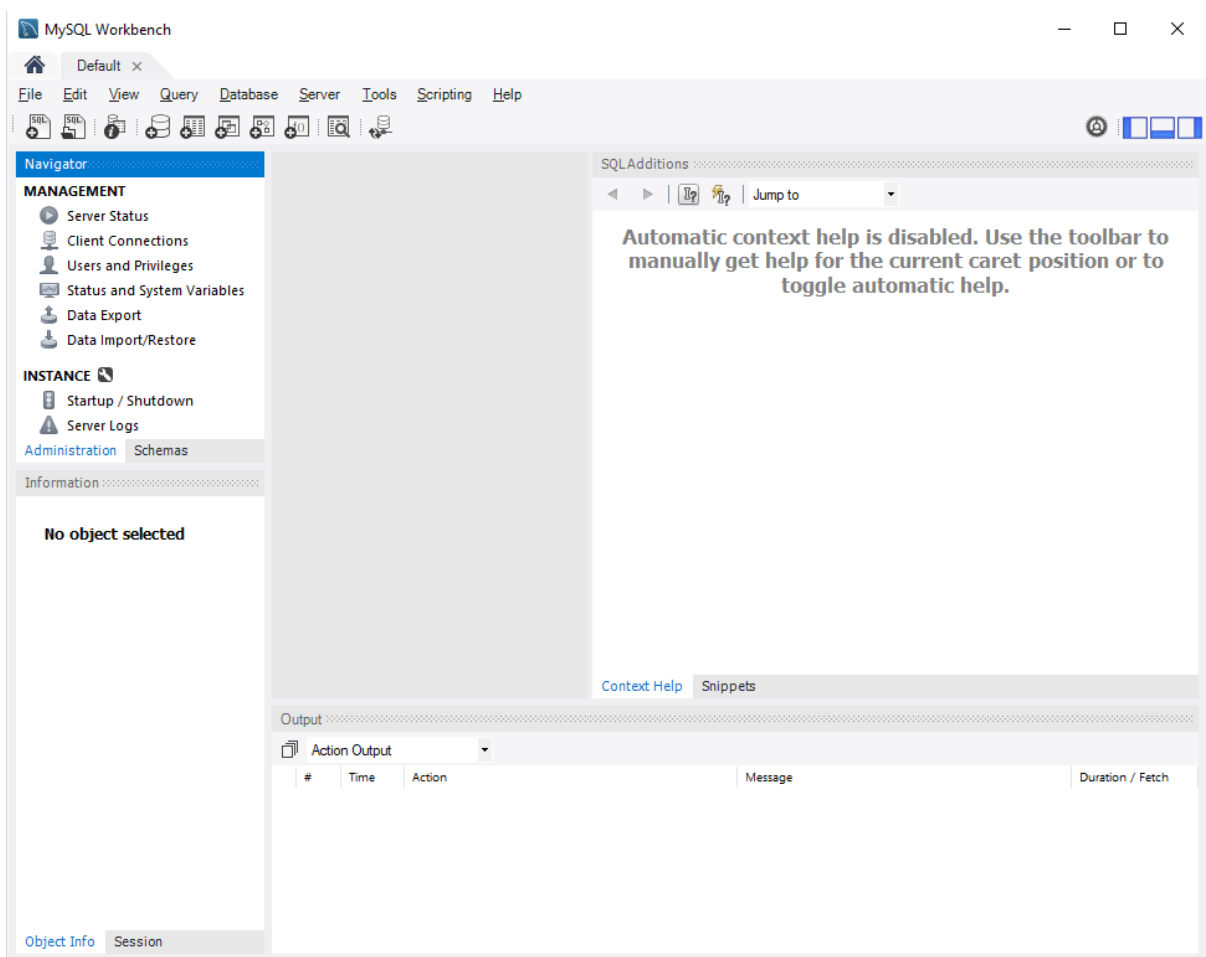


- Затем переключимся в режим SQL, выполнив команду **\sql**
- Затем, подключимся к MySQL, введя следующую строку: **\c root@localhost**, а на предложение ввести пароль напомним **secret**. Если мы все сделаем без ошибок, то опять окажемся внутри нашей СУБД и сможем создать уже вторую базу данных:

```
CREATE DATABASE second;
```

## MYSQL WORKBENCH

- Третий вариант работы с MySQL – MySQL Workbench – работает не с командной строкой, а больше полагается на графические средства взаимодействия с СУБД.
- Запускаем MySQL Workbench и в открывшемся окне нажимаем плюс в левой части (рядом с надписью MySQL Connections), затем заполняем поля **Connection Name** (можно дать любое название), **Connection Method** – Standard (TCP/IP), **Hostname** – 127.0.0.1, **Port** – 3306, **Username** – root. После подтверждения нашего выбора нажатием кнопки ОК, нажимаем мышкой на созданное соединение, вводим пароль **secret** и оказываемся в красивой графической оболочке СУБД MySQL.



- Чтобы создать свою третью базу данных, в левой части нажимаем кнопку **Create new schema ...** (четвертая слева), затем в качестве Name вводим third и нажимаем кнопку **Apply**. В новом окне подтверждаем создание базы еще раз.