

# MySQL 8.0.34

MySQL — свободная реляционная система управления базами данных (СУБД), разработанная и поддерживаемая корпорацией Oracle. База данных — это специально созданное хранилище важной информации, неотъемлемым атрибутом которого является удобный доступ ко всем хранящимся данным. Для обеспечения этого доступа используются так называемые системы управления базами данных или просто СУБД, представляющие собой особые приложения на базе определенной программной архитектуры.



```
hg$
hg$ mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.17 MySQL Community Server - GPL

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

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> USE test;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW tables;
+-----+
| Tables_in_test |
+-----+
```

## Описание программы

Одной из самых популярных СУБД на сегодняшний день является MySQL, распространяемая свободно (с некоторыми ограничениями). Эта серверная система способна эффективно функционировать во взаимодействии с интернет-сайтами и веб-приложениями. При этом она

проста в освоении, что лишь увеличивает ее популярность, особенно среди «айтишников»-любителей.

Примечательно, что результатом огромной популярности MySQL стало появление в интернете множества руководств по освоению системы, а также огромного количества всевозможных плагинов и расширений, упрощающих работу с этой системой. Это в свою очередь лишь придало системе еще большей популярности.

Несмотря на отсутствие некоторого функционала, имеющегося у других СУБД, MySQL обладает достаточно обширным разнообразием доступных инструментов для создания приложений.

## Установка сервер MySQL в Ubuntu 22.04 LTS Linux

### Шаг 1 — Обновление системы

Важно, чтобы вы обновили систему, выполнив следующие команды apt:

```
sudo apt update
sudo apt list --upgradable # получить список обновлений
sudo apt upgrade
```

### Шаг 2 — Поиск пакетов MySQL сервер

Используйте команду apt-cache или apt, как показано ниже, для поиска серверных и клиентских пакетов MySQL в Ubuntu 22.04 LTS. Например:

```
apt-cache search mysql-server
```

Система вернёт список доступных вариантов, включая сервер и клиент Oracle MySQL 8.xx и MariaDB 10.x в Ubuntu 22.04 LTS:

```
mysql-server - MySQL database server (metapackage depending on the latest version)
mysql-server-8.0 - MySQL database server binaries and system database setup
mysql-server-core-8.0 - MySQL database server binaries
default-mysql-server - MySQL database server binaries and system database setup (metapackage)
default-mysql-server-core - MySQL database server binaries (metapackage)
mariadb-server-10.6 - MariaDB database server binaries
mariadb-server-core-10.6 - MariaDB database core server files
```

Хотите узнать больше о пакете MySQL server под названием mysql-server-8.0? Выполните команду следующую команду apt, прежде чем устанавливать mysql:

```
apt info -a mysql-server-8.0
```

```
Package: mysql-server-8.0
```

```
Version: 8.0.29-0ubuntu0.22.04.2
Priority: optional
Section: database
Source: mysql-8.0
Origin: Ubuntu
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Original-Maintainer: Debian MySQL Maintainers <pkg-mysql-
maint@lists.alioth.debian.org>
Bugs: https://bugs.launchpad.net/ubuntu/+filebug
Installed-Size: 1610 kB
Provides: virtual-mysql-server
Pre-Depends: adduser (>= 3.40), debconf, mysql-common (>= 5.5)
Depends: lsb-base (>= 3.0-10), mysql-client-8.0 (>=
8.0.29-0ubuntu0.22.04.2), mysql-common (>= 5.8+1.0.4~), mysql-server-
core-8.0 (= 8.0.29-0ubuntu0.22.04.2), passwd, perl:any (>= 5.6), psmisc,
debconf (>= 0.5) | debconf-2.0
Recommends: libhtml-template-perl, mecab-ipadic-utf8
Suggests: mailx, tinyca
Conflicts: mariadb-server-10.1, mariadb-server-10.3, mysql-server-5.7,
virtual-mysql-server
Homepage: http://dev.mysql.com/
Task: lamp-server
Download-Size: 1391 kB
APT-Sources: http://archive.ubuntu.com/ubuntu jammy-updates/main amd64
Packages
Description: MySQL database server binaries and system database setup
MySQL is a fast, stable and true multi-user, multi-threaded SQL database
server. SQL (Structured Query Language) is the most popular database query
language in the world. The main goals of MySQL are speed, robustness and
ease of use.
.
This package contains all the infrastructure needed to setup system
databases.

Package: mysql-server-8.0
Version: 8.0.28-0ubuntu4
Priority: optional
Section: database
Source: mysql-8.0
Origin: Ubuntu
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Original-Maintainer: Debian MySQL Maintainers <pkg-mysql-
maint@lists.alioth.debian.org>
Bugs: https://bugs.launchpad.net/ubuntu/+filebug
Installed-Size: 1603 kB
Provides: virtual-mysql-server
Pre-Depends: adduser (>= 3.40), debconf, mysql-common (>= 5.5)
Depends: lsb-base (>= 3.0-10), mysql-client-8.0 (>= 8.0.28-0ubuntu4), mysql-
common (>= 5.8+1.0.4~), mysql-server-core-8.0 (= 8.0.28-0ubuntu4), passwd,
perl:any (>= 5.6), psmisc, debconf (>= 0.5) | debconf-2.0
Recommends: libhtml-template-perl, mecab-ipadic-utf8
```

```
Suggests: mailx, tinyca
Conflicts: mariadb-server-10.1, mariadb-server-10.3, mysql-server-5.7,
virtual-mysql-server
Homepage: http://dev.mysql.com/
Task: lamp-server
Download-Size: 1386 kB
APT-Sources: http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages
Description: MySQL database server binaries and system database setup
MySQL is a fast, stable and true multi-user, multi-threaded SQL database
server. SQL (Structured Query Language) is the most popular database query
language in the world. The main goals of MySQL are speed, robustness and
ease of use.
.
This package contains all the infrastructure needed to setup system
databases.
```

mysql-server-8.0 vs mysql-server-core-8.0

1. **mysql-server-8.0** — этот пакет нужен практически во всех случаях. Он содержит исполняемые файлы сервера баз данных MySQL, клиенты и настройки системной базы данных.
2. **mysql-server-core-8.0** — этот пакет включает двоичные файлы сервера, но не содержит всей инфраструктуры, необходимой для настройки системных баз данных. Так что это более полезно для тех, кто настраивает контейнеры Linux (Docker, LXD и прочие) и не нуждается во всех этих вещах, таких как клиенты mysql.

### Шаг 3 — Установка пакета MySQL 8 сервер

Давайте установим MySQL сервер версии 8.0 на Ubuntu 22.04 LTS:

```
sudo apt install mysql-server-8.0
```

Пример сеанса установки:

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
 libfreetype6
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
 libcgi-fast-perl libcgi-pm-perl libclone-perl libencode-locale-perl
 libevent-pthreads-2.1-7 libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-
 parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl
 libhttp-message-perl libio-html-perl liblwp-mediatypes-perl
 libmecab2 libprotobuf-lite23 libtimedate-perl liburi-perl mecab-ipadic
 mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-
 common mysql-server-core-8.0
```

### Suggested packages:

```
libdata-dump-perl libipc-sharedcache-perl libbusiness-isbn-perl libwww-  
perl mailx tinycs
```

The following NEW packages will be installed:

```
libcgi-fast-perl libcgi-pm-perl libclone-perl libencode-locale-perl  
libevent-pthreads-2.1-7 libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-  
parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl  
libhttp-message-perl libio-html-perl liblwp-mediatypes-perl
```

```
libmecab2 libprotobuf-lite23 libtimedate-perl liburi-perl mecab-ipadic  
mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-  
common mysql-server-8.0 mysql-server-core-8.0
```

0 upgraded, 27 newly installed, 0 to remove and 0 not upgraded.

Need to get 28.6 MB of archives.

After this operation, 240 MB of additional disk space will be used.

Do you want to continue? [Y/n] y

## Установка пароля для учётной записи root

Для установки пароля учётной записи **root**, запустите:

```
sudo mysql
```

Затем выполните sql запрос для установки пароля **My7Pass@Word\_9\_8A\_zE** для учётной записи **'root@localhost'**. В имени учётной записи, после @ указывается с какого адреса будет обращаться пользователь. Для локальной машины указывают **localhost** или **127.0.0.1**, для удалённой машины её IP-адрес. Для доступа из локальной или внешней сети %, но так делать не стоит без особой необходимости:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY  
'My7Pass@Word_9_8A_zE';  
exit
```

## Основные файлы конфигурации и порты MySQL 8.xx

- **mysql.service** — Имя сервиса. Вы можете управлять им с помощью следующих команд systemctl:

```
sudo systemctl start mysql.service  
sudo systemctl stop mysql.service  
sudo systemctl restart mysql.service  
sudo systemctl status mysql.service
```

- **/etc/mysql/** — Основной каталог конфигурации MySQL сервера.
- **/etc/mysql/my.cnf** — Файл конфигурации сервера баз данных MySQL. Отредактируйте файл .my.cnf (\$HOME/.my.cnf), чтобы установить пользовательские параметры. Дополнительные настройки можно переопределить из следующих двух каталогов:

```
/etc/mysql/conf.d/
```

```
/etc/mysql/mysql.conf.d/
```

- **Порт TCP/3306** — TCP/3306 сетевой порт по умолчанию для MySQL сервера и привязывается к **127.0.0.1** по соображениям безопасности. Однако вы можете его изменить, если нужен доступ к VLAN или VPN CIDR. Затем можете получить доступ к серверу MySQL, используя набор сокетов **localhost** в каталоге **/run/mysqld/**.

## Шаг 4 — Защита сервера MySQL 8

По умолчанию пароль не установлен, и необходимо настроить другие параметры. Давайте запустим следующую команду для настройки и защиты MySQL сервера:

```
sudo mysql_secure_installation
```

Программа запросит пароль для пользователя **root**, политику валидации пароля и предполагаемую надёжность пароля:

```
Securing the MySQL server deployment.
```

```
Enter password for user root: My7Pass@Word_9_8A_zE
```

```
VALIDATE PASSWORD COMPONENT can be used to test passwords  
and improve security. It checks the strength of password  
and allows the users to set only those passwords which are  
secure enough. Would you like to setup VALIDATE PASSWORD component?
```

```
Press y|Y for Yes, any other key for No: Y
```

```
There are three levels of password validation policy:
```

```
LOW      Length >= 8  
MEDIUM Length >= 8, numeric, mixed case, and special characters  
STRONG Length >= 8, numeric, mixed case, special characters and dictionary  
file
```

```
Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 2  
Using existing password for root.
```

```
Estimated strength of the password: 100  
Change the password for root ? (Press y|Y for Yes, any other key for No) : Y
```

```
New password: My7Pass@Word_9_8A_zE
```

```
Re-enter new password: My7Pass@Word_9_8A_zE
```

```
Estimated strength of the password: 100  
Do you wish to continue with the password provided?(Press y|Y for Yes, any  
other key for No) : Y
```

Далее удалим анонимного пользователя, отключим удалённый вход пользователя root в систему и удалим тестовую базу данных:

```
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.
```

```
Remove anonymous users? (Press y|Y for Yes, any other key for No) : Y
Success.
```

```
Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.
```

```
Disallow root login remotely? (Press y|Y for Yes, any other key for No) : Y
Success.
```

```
By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.
```

```
Remove test database and access to it? (Press y|Y for Yes, any other key for
No) : Y
```

```
- Dropping test database...
Success.
```

```
- Removing privileges on test database...
Success.
```

```
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.
```

```
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y
Success.
```

```
All done!
```

## Шаг 5 — Включение загрузки MySQL сервера вместе с загрузкой системы

Убедитесь, что MySQL сервер запускается при загрузке системы с помощью следующей команды:

```
sudo systemctl is-enabled mysql.service
```

Если загрузка не включена, введите следующую команду для включения загрузки:

```
sudo systemctl enable mysql.service
```

Проверьте состояние MySQL сервера, следующей командой:

```
sudo systemctl status mysql.service
```

Если всё в порядке, она должна вывести что-то вроде этого:

```
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2022-08-10 23:46:30 UTC; 2min 19s ago
     Process: 1498 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
    Main PID: 1506 (mysqld)
      Status: "Server is operational"
      Tasks: 39 (limit: 4575)
     Memory: 361.5M
        CPU: 1.090s
     CGroup: /system.slice/mysql.service
            └─1506 /usr/sbin/mysqld

Aug 10 23:46:30 nixcraft-mysql-8 systemd[1]: Starting MySQL Community Server...
Aug 10 23:46:30 nixcraft-mysql-8 systemd[1]: Started MySQL Community Server.
```

## Шаг 6 — Запуск/Остановка/Перезапуск MySQL сервера

Используя параметры командной строки, мы можем управлять MySQL сервером в Ubuntu 22.04 LTS. Давайте запустим сервер, если он ещё не запущен:

```
sudo systemctl start mysql.service
```

Для остановки MySQL сервера выполните следующую команду:

```
sudo systemctl stop mysql.service
```

Если MySQL сервер нужно перезапустить выполните:

```
sudo systemctl restart mysql.service
```

Посмотреть журнал службы MySQL можно командой **journalctl**:

```
sudo journalctl -u mysql.service -xe
```



Она выведет что-то вроде:

```
May 10 05:09:01 ubuntu-nixcraft systemd[1]: Starting MySQL Community
Server...
Subject: A start job for unit mysql.service has begun execution
Defined-By: systemd
Support: http://www.ubuntu.com/support
A start job for unit mysql.service has begun execution.
The job identifier is 597.
May 10 05:09:01 ubuntu-nixcraft systemd[1]: Started MySQL Community Server.
Subject: A start job for unit mysql.service has finished successfully
Defined-By: systemd
Support: http://www.ubuntu.com/support
A start job for unit mysql.service has finished successfully.
The job identifier is 597.
```

Файл журнала ошибок MySQL сервера по умолчанию расположен в **/var/log/mysql/error.log**.  
Посмотреть последние записи можно с помощью команды **tail**:

```
sudo tail -f /var/log/mysql/error.log
```

```
2022-05-10T05:08:59.396952Z 7 [System] [MY-013172] [Server] Received
SHUTDOWN from user boot. Shutting down mysqld (Version:
8.0.29-0ubuntu0.22.04.2).
2022-05-10T05:08:59.399628Z 0 [System] [MY-011323] [Server] X Plugin ready
for connections. Bind-address: '127.0.0.1' port: 33060, socket:
/var/run/mysqld/mysqlx.sock
2022-05-10T05:09:00.873507Z 0 [System] [MY-010910] [Server]
/usr/sbin/mysqld: Shutdown complete (mysqld 8.0.29-0ubuntu0.22.04.2)
(Ubuntu).
2022-05-10T05:09:01.640964Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld
(mysqld 8.0.29-0ubuntu0.22.04.2) starting as process 1463
2022-05-10T05:09:01.652378Z 1 [System] [MY-013576] [InnoDB] InnoDB
initialization has started.
2022-05-10T05:09:01.771700Z 1 [System] [MY-013577] [InnoDB] InnoDB
initialization has ended.
2022-05-10T05:09:01.942385Z 0 [Warning] [MY-010068] [Server] CA certificate
ca.pem is self signed.
2022-05-10T05:09:01.942503Z 0 [System] [MY-013602] [Server] Channel
mysql_main configured to support TLS. Encrypted connections are now
supported for this channel.
2022-05-10T05:09:01.974967Z 0 [System] [MY-010931] [Server]
/usr/sbin/mysqld: ready for connections. Version: '8.0.29-0ubuntu0.22.04.2'
socket: '/var/run/mysqld/mysqld.sock' port: 3306 (Ubuntu).
2022-05-10T05:09:01.974988Z 0 [System] [MY-011323] [Server] X Plugin ready
for connections. Bind-address: '127.0.0.1' port: 33060, socket:
```

```
/var/run/mysqld/mysqlx.sock
```

## Шаг 7 — Войдите на MySQL 8 сервер для тестирования

На данный момент мы узнали, как установить, настроить, защитить и запустить/остановить MySQL 8 сервер в Ubuntu 22.04 LTS. Пришло время зайти на сервер как root (администратор MySQL). Синтаксис команды следующий:

```
mysql -u {user} -p
mysql -u {user} -h {remote_server_ip} -p
mysql -u root -p
```

Пример сессии:

```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.30-0ubuntu0.22.04.1 (Ubuntu)

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>
```

Выполните команду **STATUS**, которая отображает версию и другую информацию о MySQL сервере:

```
STATUS;
```

```
mysql Ver 8.0.30-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))

Connection id:          14
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server version:         8.0.30-0ubuntu0.22.04.1 (Ubuntu)
Protocol version:      10
Connection:             Localhost via UNIX socket
```

```

Server charset: utf8mb4
Db charset: utf8mb4
Client charset: utf8mb4
Conn. charset: utf8mb4
UNIX socket: /var/run/mysqld/mysqld.sock
Binary data as: Hexadecimal
Uptime: 5 min 40 sec

```

```

Threads: 2 Questions: 20 Slow queries: 0 Opens: 142 Flush tables: 3
Open tables: 61 Queries per second avg: 0.058
-----

```

Также версию MySQL сервера, можно посмотреть следующей командой:

```
SHOW VARIABLES LIKE "%version%";
```

```

+-----+-----+
| Variable_name | Value |
+-----+-----+
| admin_tls_version | TLSv1.2,TLSv1.3 |
| immediate_server_version | 999999 |
| innodb_version | 8.0.30 |
| original_server_version | 999999 |
| protocol_version | 10 |
| replica_type_conversions | |
| slave_type_conversions | |
| tls_version | TLSv1.2,TLSv1.3 |
| version | 8.0.30-0ubuntu0.22.04.1 |
| version_comment | (Ubuntu) |
| version_compile_machine | x86_64 |
| version_compile_os | Linux |
| version_compile_zlib | 1.2.12 |
+-----+-----+
13 rows in set (0.00 sec)

```

## Шаг 8 — Создание новой базы данных MySQL и пользователя/пароля

Создадим новую базу данных mydemodb:

```
CREATE DATABASE mydemodb;
```

Затем создадим пользователя **vivekappusr** для базы данных **mydemodb** с паролем **aa09dd995C72\_5355a598fc7D8ab1230a**:

```
CREATE USER 'vivekappusr'@'%' IDENTIFIED BY
'aa09dd995C72_5355a598fc7D8ab1230a';
```

Установим права:

```
GRANT SELECT, INSERT, UPDATE, DELETE ON mydemodb.* TO 'vivekappusr'@'%';
```

И конечно предоставим все привилегии на базу данных **mydemodb**:

```
GRANT ALL PRIVILEGES ON mydemodb.* TO 'vivekappusr'@'%';
```

Посмотрим пользователей MySQL и привилегии пользователя **vivekappusr** командами:

```
SELECT USER,host FROM mysql.user;  
SHOW GRANTS FOR vivekappusr;
```

Войдём под новым пользователем в MySQL:

```
mysql -u vivekappusr -p mydemodb  
mysql -u vivekappusr -h localhost -p mydemodb
```

Где,

- -u vivekappusr : Имя пользователя
- -h localhost : Соединение с хостом localhost
- -p : Запрос пароля
- mydemodb : Соединение с базой данных: mydemodb

## Шаг 9 — Конфигурация сервера MySQL 8

Откройте файл `/etc/mysql/mysql.conf.d/mysqld.cnf` с помощью текстового редактора, например:

```
sudo vim /etc/mysql/mysql.conf.d/mysqld.cnf
```

Отредактируйте раздел `mysqld` в соответствии с вашими потребностями. Детальную информацию по параметрам можно узнать из [документации](#).

```
[mysqld]  
pid-file = /var/run/mysqld/mysqld.pid  
socket   = /var/run/mysqld/mysqld.sock  
datadir  = /var/lib/mysql  
log-error = /var/log/mysql/error.log
```

Далее, я добавляю доступ по сети:

```
# server LAN/VLAN IP and port  
bind_address = 10.147.164.6  
port = 3306  
skip_external_locking  
skip_name_resolve  
max_allowed_packet = 256M
```

```
max_connect_errors = 1000000
```

Немного тонкой настройки баз данных для проекта

```
# InnoDB
default_storage_engine = InnoDB
innodb_buffer_pool_instances = 1
innodb_buffer_pool_size = 512M
innodb_file_per_table = 1
innodb_flush_log_at_trx_commit = 0
innodb_flush_method = 0_DIRECT
innodb_log_buffer_size = 16M
innodb_log_file_size = 512M
innodb_stats_on_metadata = 0
innodb_read_io_threads = 64
innodb_write_io_threads = 64

# MyISAM Settings (set if you are using MyISAM)
key_buffer_size = 32M

low_priority_updates = 1
concurrent_insert = 2

# Connection Settings
max_connections = 100

back_log = 512
thread_cache_size = 100
thread_stack = 192K

interactive_timeout = 180
wait_timeout = 180

# Buffer Settings
join_buffer_size = 4M
read_buffer_size = 3M
read_rnd_buffer_size = 4M
sort_buffer_size = 4M
```

Настройки таблицы в соответствии с потребностями проекта:

```
# Table Settings (see below for open file limits)
table_definition_cache = 40000
table_open_cache = 40000
open_files_limit = 60000

max_heap_table_size = 128M
tmp_table_size = 128M

# Search Settings
```

```
ft_min_word_len = 3
```

Настройка ведения логов:

```
# Logging
log_error = /var/lib/mysql/mysql_error.log
log_queries_not_using_indexes = 1
long_query_time = 5
slow_query_log = 0 # Disabled for production
slow_query_log_file = /var/lib/mysql/mysql_slow.log
```

Настройка **mysqldump** для бэкапов:

```
[mysqldump]
quick
quote_names
max_allowed_packet
```

### Настройка максимального количества открытых файлов сервера MySQL(количество файловых дескрипторов)

Для загруженного MySQL 8 сервера вам необходимо настроить параметры максимального количества открытых файлов с помощью **systemd**. В противном случае вы получите сообщение об ошибке «Ну удалось увеличить количество max\_open\_files больше, чем XXXX ». Запустите:

```
sudo systemctl edit mysql.service
```

Вы увидите следующий текст:

```
### Lines below this comment will be discarded
### /lib/systemd/system/mysql.service
# # MySQL systemd service file
#
# [Unit]
# Description=MySQL Community Server
# After=network.target
#
# [Install]
# WantedBy=multi-user.target
#
# [Service]
# Type=notify
# User=mysql
# Group=mysql
# PIDFile=/run/mysqld/mysqld.pid
# PermissionsStartOnly=true
# ExecStartPre=/usr/share/mysql/mysql-systemd-start pre
```

```
# ExecStart=/usr/sbin/mysqld
# TimeoutSec=infinity
# Restart=on-failure
# RuntimeDirectory=mysqld
# RuntimeDirectoryMode=755
# LimitNOFILE=10000
#
# # Set environment variable MYSQLD_PARENT_PID. This is required for restart.
# Environment=MYSQLD_PARENT_PID=1
```

Добавьте свою конфигурацию между:

```
### Anything between here and the comment below will become the new contents
of the file
```

```
### Lines below this comment will be discarded
```

Например (замените **1800000** на желаемое значение. Для максимального поддерживаемого значения используйте **LimitNOFILE=infinity** вместо **LimitNOFILE=1800000**):

```
### Editing /etc/systemd/system/mysql.service.d/override.conf
### Anything between here and the comment below will become the new contents
of the file
```

```
[Service]
LimitNOFILE=1800000
```

```
### Lines below this comment will be discarded
```

```
### /lib/systemd/system/mysql.service
# # MySQL systemd service file
#
# [Unit]
# Description=MySQL Community Server
# After=network.target
#
# [Install]
# WantedBy=multi-user.target
#
# [Service]
# Type=notify
# User=mysql
# Group=mysql
# PIDFile=/run/mysqld/mysqld.pid
# PermissionsStartOnly=true
# ExecStartPre=/usr/share/mysql/mysql-systemd-start pre
# ExecStart=/usr/sbin/mysqld
# TimeoutSec=infinity
```

```
# Restart=on-failure
# RuntimeDirectory=mysqlld
# RuntimeDirectoryMode=755
# LimitNOFILE=10000
#
# # Set environment variable MYSQLD_PARENT_PID. This is required for restart.
# Environment=MYSQLD_PARENT_PID=1
```

Создайте или отредактируйте **/etc/sysctl.d/100-custom.conf** и добавьте:

```
fs.nr_open=1800000
```

Обновите изменения:

```
sudo sysctl -p /etc/sysctl.d/100-custom.conf
```

Затем перезагрузите и перезапустите службу mysql:

```
sudo systemctl daemon-reload
sudo systemctl restart mysql
```

Проверьте:

```
mysql -u root -p -e 'SHOW GLOBAL VARIABLES LIKE "open_files_limit";'
```

Пример вывода:

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| open_files_limit | 1800000 |
+-----+-----+
```

Также можете воспользоваться следующей командой, чтобы запросить текущий лимит открытых файлов MySQL сервера в Ubuntu 22.04 LTS:

```
cat /proc/$(cat /var/run/mysqld/mysqld.pid)/limits
## ИЛИ ##
grep 'open files' /proc/$(cat /var/run/mysqld/mysqld.pid)/limits
```

Выведет в консоль:

```
Max open files          1800000          1800000          files
```

## Достоинства и недостатки



Переходим к обзору сильных и слабых сторон MySQL от Oracle.

Плюсы:

- Простота в использовании.
- Обширный функционал.
- Безопасность.
- Масштабируемость.
- Скорость.

Минусы:

- Недостаточная надежность.
- Низкая скорость разработки.

## Ссылки и Примечания

[Сайт разработчика](#)

[Страница загрузки](#)

[Краткое руководство по использованию репозитория MySQL APT](#)

[Репозиторий MySQL APT](#)

[Источник: «How to install MySQL server on Ubuntu 22.04 LTS Linux»](#)

[MySQL Documentation](#)

[База данных PHP MySQL](#)

## Файлы для загрузки

[Установщики MySQL 8.0.34](#)



Windows (x86, 64-разрядная версия), установщик MSI

Linux — универсальный (glibc 2.28) (x86, 64-разрядная версия), архив TAR

Ubuntu / Debian (независимо от архитектуры), пакет DEB

From:

<https://book51.ru/> - **book51.ru**

Permanent link:

[https://book51.ru/doku.php?id=software:development:ps\\_pycharm:mysql:mysql](https://book51.ru/doku.php?id=software:development:ps_pycharm:mysql:mysql)

Last update: **2024/02/25 20:10**

