# MySQL程序

## 调用MySQL程序

首先将mysql安装文件中的bin目录作为环境变量加载进去

连接mysql数据库

Shell>mysql

Shell>mysql -h localhost -u myname -p password mydb

默认的参数配置：

[--default-auth=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_default-auth)***[plugin](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_default-auth)***

关于使用客户端身份验证插件的提示

[--host=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_host)***[host\_name](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_host)***， -h ***host\_name***

运行服务器的主机。默认值是 localhost。

[--password[=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_password)***[pass\_val](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_password)***[]](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_password)， -p[***pass\_val***]

MySQL帐户的密码。如前所述，密码值是可选的，但如果给出，则密码值之间必须 *没有空格*-p或 [--password=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_password)密码。默认是不发送密码。

[--pipe](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_pipe)， -W

在Windows上，使用命名管道连接到服务器。服务器必须启动， [--enable-named-pipe](https://dev.mysql.com/doc/refman/5.7/en/server-options.html" \l "option_mysqld_enable-named-pipe)并启用命名管道连接。

[--port=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_port)***[port\_num](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_port)***， -P ***port\_num***

用于连接的端口号，用于使用TCP / IP进行的连接。默认端口号是3306。

[--protocol={TCP|SOCKET|PIPE|MEMORY}](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_protocol)

此选项明确指定用于连接到服务器的协议。当其他连接参数通常会导致使用的协议不是您想要的协议时，这很有用。例如，localhost默认情况下，Unix上的连接是使用Unix套接字文件创建的

| [--protocol](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_protocol)**值** | **连接协议** | **允许的操作系统** |
| --- | --- | --- |
| TCP | 到本地或远程服务器的TCP / IP连接 | 所有 |
| SOCKET | Unix套接字文件连接到本地服务器 | 只有Unix |
| PIPE | 命名管道连接到本地或远程服务器 | 仅限Windows |
| MEMORY | 与本地服务器的共享内存连接 | 仅限Windows |
|  |  |  |

[--shared-memory-base-name=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_shared-memory-base-name)***[name](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_shared-memory-base-name)***

在Windows上，使用共享内存名称，用于使用共享内存连接到本地服务器。默认值是MYSQL。共享内存名称区分大小写。

必须启动服务器， [--shared-memory](https://dev.mysql.com/doc/refman/5.7/en/server-options.html" \l "option_mysqld_shared-memory)并启用共享内存连接选项。

[--socket=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_socket)***[file\_name](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_socket)***， -S ***file\_name***

在Unix上，使用Unix套接字文件的名称，用于使用命名管道连接到本地服务器。默认的Unix套接字文件名是 /tmp/mysql.sock。

在Windows上，用于连接到本地服务器的命名管道的名称。默认的Windows管道名称是 MySQL。管道名称不区分大小写。

服务器必须启动， [--enable-named-pipe](https://dev.mysql.com/doc/refman/5.7/en/server-options.html" \l "option_mysqld_enable-named-pipe)并启用命名管道连接。

--ssl\*

[--ssl](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-options.html" \l "option_general_ssl) 如果服务器配置了SSL支持，则使用 开头的选项用于使用SSL建立到服务器的安全连接。有关详细信息，请参见[第6.4.2节“加密连接的命令选项”](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-options.html" \o "6.4.2加密连接的命令选项)。

[--tls-version=](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-options.html" \l "option_general_tls-version)***[protocol\_list](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-options.html" \l "option_general_tls-version)***

客户端允许的加密连接协议。该值是一个包含一个或多个协议名称的逗号分隔列表。可以为这个选项命名的协议取决于用于编译MySQL的SSL库。有关详细信息，请参见 [第6.4.6节“加密连接协议和密码”](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-protocols-ciphers.html" \o "6.4.6加密的连接协议和密码)。

该选项已添加到MySQL 5.7.10中。

[--user=](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_user)***[user\_name](https://dev.mysql.com/doc/refman/5.7/en/connecting.html" \l "option_general_user)***， -u ***user\_name***

您想要使用的MySQL帐户的用户名。默认的用户名ODBC在Windows上，或者您在Unix上的Unix登录名。

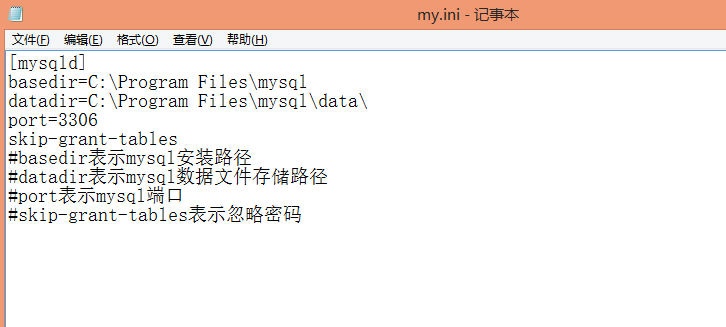
Window:如果要设置这些选项的默认参数的话，建议打开MySQL安装目录下的my.ini文件进行配置.

C:\ProgramData\MySQL\MySQL Server 5.7 位置一般在这里

如果一开始就想指明选项文件，那么应该在mysqld --install --default\_file 里面给出

C:\> "C:\Program Files\MySQL\MySQL Server 5.7\bin\mysqld" --install MySQL --defaults-file=C:\my-opts.cnf

这样，MySQLD作为一个服务就在系统启动的时候自动启动了，不用每次都net start mysqld,net stop myslqd了。



Liunx/unix下一般是在: /ect/my.cnf下进行配置的,任何在命令行中给出的任何长选项都可以在选项文件中中给出，要获取程序的可用选项列表，可以使用--help运行它.

语法规则：

选项文件中，可以省略选项名称中的前两个破折号，并且每行只指定一个选项

，例如--quick与--host=localhost在命令行上，应该被指定为quick与host=localhost

1. 注释:#comment或者;comment
2. [group]要为其设置选项的程序或者组的名称 [client]代表全局
3. opt\_name 相当于--opt\_name
4. Opt\_name=value 设置某个参数的默认值
5. 可以在选项文件中使用!include /home/mydir/myopt.conf包含其他的选项文件
6. 可以在选项文件中max\_allowed\_packet=16777216来设置程序变量

mysql> show variables like 'max%' ; 注释：显示程序变量，大多数程序变量可以在mysqld命令中设置....

+-----------------------+------------+  
| Variable\_name | Value |  
+-----------------------+------------+  
| max\_allowed\_packet | 1048576 |

...  
  
mysql> set max\_allowed\_packet = 1500000;  
Query OK, 0 rows affected (0.03 sec)  
  
mysql> show variables like 'max%' ;

+-----------------------+------------+  
| Variable\_name | Value |  
+-----------------------+------------+  
| max\_allowed\_packet | 1499136 |

## MySQL服务器启动程序

Mysqld ,也被称为mysql服务器，是执行mysql安装中大部分工作的主要程序。

Mysqld\_safe是在unix上启动mysqld服务器的推荐方式，加入了一些安全功能，例如发生错误的时候重启服务器并将运行时信息记录到错误日志中

## MySQL安装相关程序

[4.4.1 comp\_err - 编译MySQL错误消息文件](https://dev.mysql.com/doc/refman/5.7/en/comp-err.html)

[4.4.2 mysql\_install\_db - 初始化MySQL数据目录](https://dev.mysql.com/doc/refman/5.7/en/mysql-install-db.html)

[4.4.3 mysql\_plugin - 配置MySQL服务器插件](https://dev.mysql.com/doc/refman/5.7/en/mysql-plugin.html)

[4.4.4 mysql\_secure\_installation - 改进MySQL安装安全性](https://dev.mysql.com/doc/refman/5.7/en/mysql-secure-installation.html)

[4.4.5 mysql\_ssl\_rsa\_setup - 创建SSL / RSA文件](https://dev.mysql.com/doc/refman/5.7/en/mysql-ssl-rsa-setup.html)

[4.4.6 mysql\_tzinfo\_to\_sql - 加载时区表](https://dev.mysql.com/doc/refman/5.7/en/mysql-tzinfo-to-sql.html)

[4.4.7 mysql\_upgrade - 检查并升级MySQL表](https://dev.mysql.com/doc/refman/5.7/en/mysql-upgrade.html)

## mysql客户端程序

[4.5.1 mysql - MySQL命令行工具](https://dev.mysql.com/doc/refman/5.7/en/mysql.html)

[4.5.2 mysqladmin - 用于管理MySQL服务器的客户端](https://dev.mysql.com/doc/refman/5.7/en/mysqladmin.html)

[4.5.3 mysqlcheck - 一个表维护程序](https://dev.mysql.com/doc/refman/5.7/en/mysqlcheck.html)

[4.5.4 mysqldump - 数据库备份程序](https://dev.mysql.com/doc/refman/5.7/en/mysqldump.html)

[4.5.5 mysqlimport - 数据导入程序](https://dev.mysql.com/doc/refman/5.7/en/mysqlimport.html)

[4.5.6 mysqlpump - 数据库备份程序](https://dev.mysql.com/doc/refman/5.7/en/mysqlpump.html)

[4.5.7 mysqlsh - MySQL Shell](https://dev.mysql.com/doc/refman/5.7/en/mysqlsh.html)

[4.5.8 mysqlshow - 显示数据库，表和列信息](https://dev.mysql.com/doc/refman/5.7/en/mysqlshow.html)

[4.5.9 mysqlslap - 加载仿真客户端](https://dev.mysql.com/doc/refman/5.7/en/mysqlslap.html)

## 5.mysql管理和使用程序

[4.6.1 innochecksum - 离线InnoDB文件校验和实用程序](https://dev.mysql.com/doc/refman/5.7/en/innochecksum.html)

[4.6.2 myisam\_ftdump - 显示全文索引信息](https://dev.mysql.com/doc/refman/5.7/en/myisam-ftdump.html)

[4.6.3 myisamchk - MyISAM表维护实用程序](https://dev.mysql.com/doc/refman/5.7/en/myisamchk.html)

[4.6.4 myisamlog - 显示MyISAM日志文件内容](https://dev.mysql.com/doc/refman/5.7/en/myisamlog.html)

[4.6.5 myisampack - 生成压缩的只读MyISAM表](https://dev.mysql.com/doc/refman/5.7/en/myisampack.html)

[4.6.6 mysql\_config\_editor - MySQL配置实用程序](https://dev.mysql.com/doc/refman/5.7/en/mysql-config-editor.html)

[4.6.7 mysqlbinlog - 处理二进制日志文件的实用程序](https://dev.mysql.com/doc/refman/5.7/en/mysqlbinlog.html)

[4.6.8 mysqldumpslow - 总结慢速查询日志文件](https://dev.mysql.com/doc/refman/5.7/en/mysqldumpslow.html)

## 6.mysql程序开发使用程序

[4.7.1 mysql\_config - 显示编译客户端的选项](https://dev.mysql.com/doc/refman/5.7/en/mysql-config.html)

[4.7.2 my\_print\_defaults - 显示来自选项文件的选项](https://dev.mysql.com/doc/refman/5.7/en/my-print-defaults.html)

[4.7.3 resolve\_stack\_dump - 将数值堆栈跟踪转储解析为符号](https://dev.mysql.com/doc/refman/5.7/en/resolve-stack-dump.html)

# MYSQLD程序

mysqld的帮助程序是mysql --verbose --help

它的输出包含myslqd选项和可配置系统变量的列表,我感觉这里才是真正的程序变量所在的位置，基本都包含在mysqld命令中了。

要查看当前服务器运行时候使用的当前系统变量，请连接到它并执行以下语句:

Mysql>show variables;

要查看正在运行的服务器的一些统计和状态指示器，请执行以下的语句：

Mysql>show status;

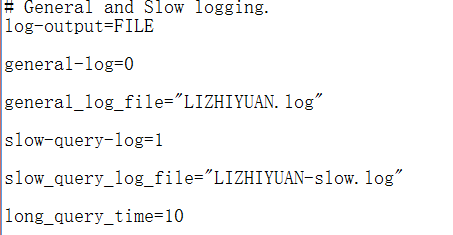
可以选择在mysqld 命令后接--选项或者是--系统变量，但一般还是建议放到选项文件中去。

更多的内容可以参考官网

# mysql日志

1. 选择常规查询和慢速查询日志输出目标

数据库文件的general\_log和slow\_log表存放数据日志，也可以是日志文件，可以选择任一个或两个目的地



服务器启动时的日志控制：

--log-output选项指定日志输出的目的地。该选项本身不会启动日志，它的语法是：--log-output[=value,...],如果--log-output给定值，则该值应该是一个或多个单词table(记录到表),file（记录到文件）或者none（不记录到表或文件）的以逗号分隔的列表

--general\_log开启一般查询

--slow\_query\_log 开启慢速查询

运行时的日志控制：

--log\_output 只是当前的记录目标

--general\_log 开启常规查询

--slow\_query\_log 开启慢查询

--general\_log\_file 一般查询日志的文件名称

--slow\_query\_file 慢速查询日志的文件名称

1. 错误日志



window中，myslqd 使用--log-error,--pid-file,--colose选项，确定是否将错误输出到控制台或者文件。

--console 将错误日志写入控制台

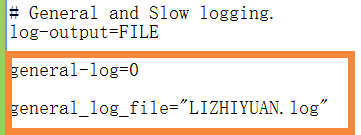
--log-error 没有给出命名文件，会写入host\_name.err数据目录或者是--pid-file指定文件

--log-error 给出命名文件，该文件.err 也会写入错误信息。

unix系统中，myslqd 使用--log-error来确定mysqld是否将错误日志写入控制台或文件，如果是文件，则使用文件名

1. 一般查询日志

记录mysqld正在做什么的记录，服务器在客户端连接或者断开的时候，将信息写入此日志，并记录从客户端收到的每条SQL语句，当你怀疑客户端发生错误，并向确认知道客户端发送给mysqld的时候，这个日志会非常有用。



1. 二进制日志
2. 慢查询日志

慢查询日志由SQL语句组成，这些SQL语句花费了超过long\_query\_time几秒的时间才能执行，并且至少min\_examined\_row\_limit需要检查行。

1. DDL日志