

前言

为了MySQL8.0.28安装教程我竟然在MySQL官方文档逛了一天，至此献给想入门MySQL8.0的初学者。以目前最新版本的MySQL8.0.28为例进行安装与初步使用的详细讲解，面向初学者的详细教程。无论是Windows还是Linux上安装，咱都会。**这也许是迄今为止全网最最最详细的MySQL8.0.28的安装与使用教程。**

温故而知新，可以为师矣。咱缺的是学习的途径吗？答案并不是，而是缺乏学习的方法。官方文档是很详细的，而且是权威的，其它的那些书籍、博文啊都是基于官方文档以及自己的使用经验总结的。写这篇文章的用意，希望大家可以总结自己的学习方法，善于利用官方文档来提升自己。

建议初学者多在命令行窗口下进行练习，熟能生巧。达到一定的熟练程度，再借助客户端工具提高我们的工作效率。最终目的是啥？活下去呗，提高捞金能力。当然开个玩笑，回到正题，接着往下看。

```
D:\work\mysql-8.0.28-win64\bin>mysql -uroot -p -P 3307 1、登录到MySQL，指定端口，默认为3306
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.28 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> select version(); 2、查看数据库版本，目前最新版本为8.0.28
+-----+
| version() |
+-----+
| 8.0.28    |
+-----+
1 row in set (0.00 sec)

mysql> show databases; 1、查看初始化安装默认有哪些数据库
+-----+
| Database |
+-----+
| information_schema |
| mysql         |
| performance_schema |
| sys           |
+-----+
4 rows in set (0.00 sec)

mysql> create database TEST; 2、创建数据库TEST
Query OK, 1 row affected (0.01 sec)

mysql> use test; 3、切换数据库，并使用TEST数据库
Database changed
mysql> create table STUDY(ID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,NAME VARCHAR(16))ENGINE=InnoDB;
Query OK, 0 rows affected (0.02 sec) 4、创建一张STUDY表，并指定默认存储引擎为InnoDB

mysql> INSERT INTO STUDY VALUES(1,'梦梦');
Query OK, 1 row affected (0.01 sec) 5、在创建的STUDY表中插入一条演示数据

mysql> SELECT * FROM STUDY; 6、查询STUDY表，显示出我们插入的数据，基本的数据库操作就完成了。
+----+-----+
| ID | NAME |
+----+-----+
| 1  | 梦梦 |
+----+-----+
```

从下载到安装，再到忘记密码解决方案。一步步使用命令行窗口学会基本操作，然后使用客户端远程连接工具。最后配合时下比较火热的Java语言进行演示如何使用JDBC连接最新的MySQL8.0数据库，以及执行查询返回结果。

正文

咱也不多哔哔，直接上干货，要的就是实用性和性价比！

一、MySQL8.0.28下载

可以下载msi文件一键安装或者解压版zip文件进行命令行初始化安装。

MySQL官网下载地址

<https://dev.mysql.com/downloads/mysql/>

1、Windows版本下载

在Windows下可以选择下载msi文件或者解压版zip文件。**一般使用，选择我使用紫色框线选中的即可。**关于下面的Debug Test Suite，是带有很多的测试套件在里面，对于有测试需求的人员可以进行下载。

MySQL Community Downloads

MySQL Community Server

General Availability (GA) Releases Archives

MySQL Community Server 8.0.28

Select Operating System:
Microsoft Windows

Looking for previous GA versions?

Recommended Download:

MySQL Installer for Windows

All MySQL Products. For All Windows Platforms. In One Package.

Starting with MySQL 5.5 the MySQL Installer package replaces the standalone self packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI

Go to Download Page >

Other Downloads:

Windows (x86, 64-bit), ZIP Archive (mysql-8.0.28-winx64.zip)	8.0.28	211.7M	Download
Windows (x86, 64-bit), ZIP Archive Debug Binaries & Test Suite (mysql-8.0.28-winx64-debug-test.zip)	8.0.28	506.3M	Download

2、Linux版本下载

根据自己需要的Linux发行版版本适配的MySQL进行选择。比如，我个人选择的是自己比较熟悉的Redhat7系列进行下载。同样有bundle版本，包含了一些插件和依赖在里面，便于使用rpm包安装。安装单个的server服务，需要安装其它的依赖包比较繁琐。对于初学者，**建议直接下载RPM bundle版本**。我偏不，就要折腾。那也行，请接着往下看，一样提供了详细安装步骤。

MySQL Community Downloads

MySQL Community Server

[General Availability \(GA\) Releases](#) [Archives](#)

MySQL Community Server 8.0.28

Select Operating System: 选择你需要的Linux发行版
Red Hat Enterprise Linux / Oracle Linux

Select OS Version: 选择Linux发行版的版本，比如Redhat7系列
Red Hat Enterprise Linux 7 / Oracle Linux 7 (x86_64-bit)

Looking for previous GA versions?

Install Using Yum:

MySQL Yum Repository

Supported Platforms:

- Red Hat Enterprise Linux / Oracle Linux
- Fedora

Download Now »

Download Packages:

RPM Bundle (mysql-8.0.28-1.el7.x86_64.rpm-bundle.tar)	8.0.28 804.0M	Download
RPM Package, MySQL Server (mysql-community-server-8.0.28-1.el7.x86_64.rpm)	8.0.28 451.2M	Download

3、注意事项

一般人可能没仔细看，官方会提示登录，加了button按钮字体非常显眼，而下面的我需要立即下载则字体很小。所以注意了，选择下面的No thanks，我需要立即下载。选择的是社区版本，免费提供下载。

MySQL Community Downloads 社区版是免费提供下载的

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. 选择不登录即可下载

二、MySQL8.0安装

初学者尝鲜，建议在Windows下安装。

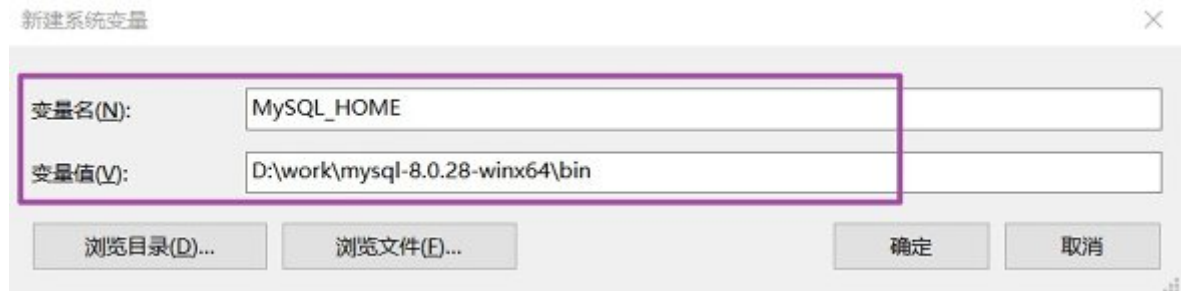
一般情况，默认安装一个MySQL版本服务实例。也不排除预算有限，在同台服务器上安装多个实例进行测试。默认端口3306，如果在发布（生产）环境建议修改默认端口，达到不让别人一下就猜到的目的。接下来安装测试多个MySQL服务版本共存一个操作系统下，只针对于Windows下安装多个服务（没有使用虚拟机工具，真机环境下测试）。Linux下有便捷的yum源以及apt方式安装，一键安装所需依赖，但

也有比较繁琐的rpm包安装。

1、Windows下安装

配置环境变量，编辑系统环境变量，控制面板>所有控制面板项>系统>高级系统配置>系统环境变量：

```
#变量名
MySQL_HOME
#变量值
D:\work\mysql-8.0.28-winx64\bin
```



msi文件安装就不介绍了，傻瓜式的一键安装，注意选择路径。

主要介绍解压版本zip的安装与实例化：

```
mysql-8.0.28-winx64.zip
```

1.1、解压上面准备的安装包mysql-8.0.28-winx64.zip

在解压后的D:\work\mysql-8.0.28-winx64目录下新增my.ini文件，默认解压版是没有的。然后加入如下配置：

```
[client]
# 设置mysql客户端默认字符集
default-character-set=utf8

[mysqld]
# 设置3307端口,有多个服务,为了不冲突修改默认的3306端口为3307
port=3307
# 设置mysql的安装目录
basedir=D:\\work\\mysql-8.0.28-winx64
# 设置 mysql数据库存放目录
datadir=D:\\work\\mysql-8.0.28-winx64\\data
# 允许最大连接数
max_connections=20
# 服务端使用的字符集默认为8比特编码的latin1字符集
character-set-server=utf8
# 创建新表时将使用的默认存储引擎
default-storage-engine=INNODB
```

1.2、实例化

以管理员身份运行CMD命令窗口，切换到mysql解压后的目录下：

```
-- 第一步执行d:，切换到D盘
d:
-- 第二步执行cd命令，切换到个人安装mysql的bin目录下
cd D:\work\mysql-8.0.28-winx64\
```

设置为空密码，去掉不必要的麻烦。

```
mysqld --initialize-insecure
```

1.3、安装服务

进入到解压后MySQL的bin目录下，执行安装服务命令：

```
cd D:\work\mysql-8.0.28-winx64\bin
mysqld install
mysqld install --service -mysql8
```

如果没有安装多个服务，使用mysqld install即可。可以不用指定服务名，默认的服务名为MySQL。

1.4、服务命令使用，需要管理员身份运行CMD命令，注意看我的路径是在bin目录下执行的

我没有配置MySQL8的系统环境变量，所以都在MySQL的bin目录中执行命令。

启动服务net start mysql

```
# 启动服务
net start mysql
D:\work\mysql-8.0.28-winx64\bin>net start mysql8
MySQL8 服务正在启动 .
MySQL8 服务已经启动成功。
```

停止服务net stop mysql

```
# 停止服务
net stop mysql
D:\work\mysql-8.0.28-winx64\bin>net stop mysql8
MySQL8 服务正在停止..
MySQL8 服务已成功停止。
```

删除服务sc delete mysql

```
# 删除服务
sc delete mysql
# 或者mysqld remove删除服务，需要在mysql的bin目录下执行mysqld命令
mysqld remove mysql
```

1.5、注意事项

```
# 安装服务指定了服务名为MySQL8
计算机\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\MySQL8
# 或者是MySQL
计算机\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\MySQL
```


安装服务指定了服务名为MySQL8。在下面的多实例服务共存也提到过，需要将原始残留的注册表删掉，重启电脑，再进行安装即可。

出现了丢失MSVCR120.dll，缺少组件，安装以下组件解决

vcredist_x64.exe

vcredist_x86.exe

[Download Visual C++ Redistributable Packages for Visual Studio 2013 from Official Microsoft Download Center](#)

注意：使用了mysql -initialize，密码是随机生成的，在mysql的错误日志中可以找到

例如我的日志（mysql的data中以.err结尾的文件）

```
A temporary password is generated for root@localhost: 6hk20yueza=M
```

修改密码的命令

```
ALTER USER 'root'@'localhost' IDENTIFIED BY '新密码'
```

之所以在Windows下介绍的如此详细，是因为我们平时的工作环境更多的是在Windows下进行的。就算使用Linux环境一般也是使用虚拟机配合Linux发行版，再就是云服务器了。MySQL的一些命令都熟悉了，Linux下安装还能难倒你吗？直接翻一翻官方文档即可。

2、Linux下安装

建议初学者不要像我这样去安装rpm包，**你可以直接下载rpm bundle，或者使用mysql官方的yum源**。个人有多年Linux使用经验，以及有一定的实际工作经验；知道如何判断哪些包是必须的，以及哪些需要被替换掉。

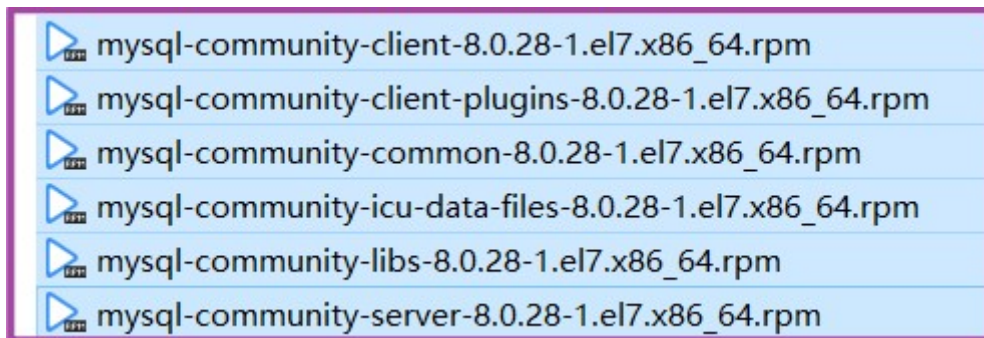
一定要注意Linux操作系统的权限问题，权限在最小范围内满足即可。

2.1、准备好安装包

直接在官网下载，或者使用wget命令下载都可以，同样可以使用官网的yum源进行安装。亦或是使用apt命令获取安装。至于为什么一些Linux发行版将MySQL从默认中移除了，因为MySQL被Oracle收购后存在闭源的风险。取而代之的是她的妹妹MariaDB，这也是为什么我在安装的时候提到了MariaDB。

2.2、安装rpm包

系统会提示哪些是需要的依赖包，所以我事先**准备需要的依赖包**。



```
[mysql@localhost ~]$ rpm -ivh mysql-community-server-8.0.28-1.el7.x86_64.rpm
警告: mysql-community-server-8.0.28-1.el7.x86_64.rpm: 头V4 RSA/SHA256 Signature,
密钥 ID 3a79bd29: NOKEY
错误: 依赖检测失败:
```

```

mysql-community-client(x86-64) >= 8.0.11 被 mysql-community-server-8.0.28-1.e17.x86_64 需要
mysql-community-common(x86-64) = 8.0.28-1.e17 被 mysql-community-server-8.0.28-1.e17.x86_64 需要
mysql-community-icu-data-files = 8.0.28-1.e17 被 mysql-community-server-8.0.28-1.e17.x86_64 需要

[root@localhost mysql]# rpm -ivh mysql-community-icu-data-files-8.0.28-1.e17.x86_64.rpm
警告: mysql-community-icu-data-files-8.0.28-1.e17.x86_64.rpm: 头V4 RSA/SHA256 Signature, 密钥 ID 3a79bd29: NOKEY
准备中... ##### [100%]
正在升级/安装...
1:mysql-community-icu-data-files-8.##### [100%]
[root@localhost mysql]# rpm -ivh mysql-community-common-8.0.28-1.e17.x86_64.rpm
警告: mysql-community-common-8.0.28-1.e17.x86_64.rpm: 头V4 RSA/SHA256 Signature, 密钥 ID 3a79bd29: NOKEY
准备中... ##### [100%]
正在升级/安装...
1:mysql-community-common-8.0.28-1.e##### [100%]

[root@localhost mysql]# rpm -ivh mysql-community-client-8.0.28-1.e17.x86_64.rpm
警告: mysql-community-client-8.0.28-1.e17.x86_64.rpm: 头V4 RSA/SHA256 Signature, 密钥 ID 3a79bd29: NOKEY
错误: 依赖检测失败:
mysql-community-client-plugins = 8.0.28-1.e17 被 mysql-community-client-8.0.28-1.e17.x86_64 需要
mysql-community-libs(x86-64) >= 8.0.11 被 mysql-community-client-8.0.28-1.e17.x86_64 需要

```

2.2.1、安装依赖包，然后使用 `rpm -qa | grep mysql` 查询哪些被安装了。怎么传到服务器上，简单一点scp命令即可。

```

$ scp mysql-community-server-8.0.28-1.e17.x86_64.rpm mysql@192.168.245.147:/home/mysql
mysql@192.168.245.147's password:
mysql-community-server-8.0.28-1.e17.x86_64.rpm 100% 451MB 62.9MB/s 00:07

```

利用Git bash使用scp命令传输到虚拟机环境

```

rpm -ivh mysql-community-common-8.0.28-1.e17.x86_64
rpm -ivh mysql-community-libs-8.0.28-1.e17.x86_64
rpm -ivh mysql-community-icu-data-files-8.0.28-1.e17.x86_64
rpm -ivh mysql-community-client-8.0.28-1.e17.x86_64
rpm -ivh mysql-community-client-plugins-8.0.28-1.e17.x86_64
[root@localhost mysql]# rpm -qa | grep mysql

```

2.2.2、Redhat7系列需要卸载原有的mariadb-libs，替换为mysql-community-libs依赖

```
$ yum remove mariadb-libs
```

2.2.3、正式安装server

```
$ rpm -ivh mysql-community-server-8.0.28-1.e17.x86_64.rpm
```

查看安装的版本**Ver 8.0.28 for Linux on x86_64**

```
[mysql@localhost ~]$ mysqladmin --version
mysqladmin ver 8.0.28 for Linux on x86_64 (MySQL Community Server - GPL)
```

2.2.4、赋予mysql安装目录所有者为mysql用户，rpm包默认安装后的路径在/var/lib/mysql。在授予mysql用户所有者和所属组权限之后，你可以使用mysql用户进行登录或者启动服务与关闭服务。

```
[mysql@localhost ~]$ mysql -uroot -p 01、使用mysql用户登录
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.28 MySQL Community Server - GPL
Copyright (c) 2000, 2022, Oracle and/or its affiliates.
02、显示出mysql版本
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 mysql 03、切换到mysql数据库，便于修改一些权限
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
```

```
#添加mysql组
$ groupadd mysql
#新增mysql用户到mysql组中
$ useradd -g mysql mysql
#修改mysql用户密码
$ mysql passwd
#修改所有者
$ chown -R mysql:mysql /var/lib/mysql
```

tips：你也可以将mysql用户加入到/etc/sudoers配置文件中，限制mysql用户使用的权限。

2.3、初始化

设置密码为空，后续登录可修改密码

```
$ mysqld --initialize-insecure
```

2.4、启动服务与查看服务状态

```
[root@localhost mysql]# systemctl start mysqld 01、启动mysqld服务
[root@localhost mysql]# systemctl status mysqld 02、查看mysqld服务状态
• mysqld.service - MySQL Server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
   Active: active (running) since 三 2022-02-02 14:25:51 CST; 6s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
   Process: 5125 ExecStartPre=/usr/bin/mysqld_pre_systemd (code=exited, status=0/SUCCESS)
   Main PID: 5202 (mysqld)
    Status: "Server is operational"
     Tasks: 38
    Memory: 505.9M
   CGroup: /system.slice/mysqld.service
           └─5202 /usr/sbin/mysqld

2月 02 14:25:44 localhost systemd[1]: Starting MySQL Server...
2月 02 14:25:51 localhost systemd[1]: Started MySQL Server.
```

Redhat7系列使用命令启动MySQL服务


```
$ systemctl start mysqld
```

设置开机自启

```
$ systemctl enable mysqld
```

关闭服务

```
$ systemctl stop mysqld
```

重启服务

```
$ systemctl restart mysqld
```

登录mysql

```
$ mysql -uroot -p
```

2.5、设置防火墙

加入mysql服务以及需要的端口3306

```
$ firewall-cmd --zone=public --add-port=3306/tcp --permanent  
$ firewall-cmd --zone=public --add-service=mysql --permanent  
$ firewall-cmd --reload
```

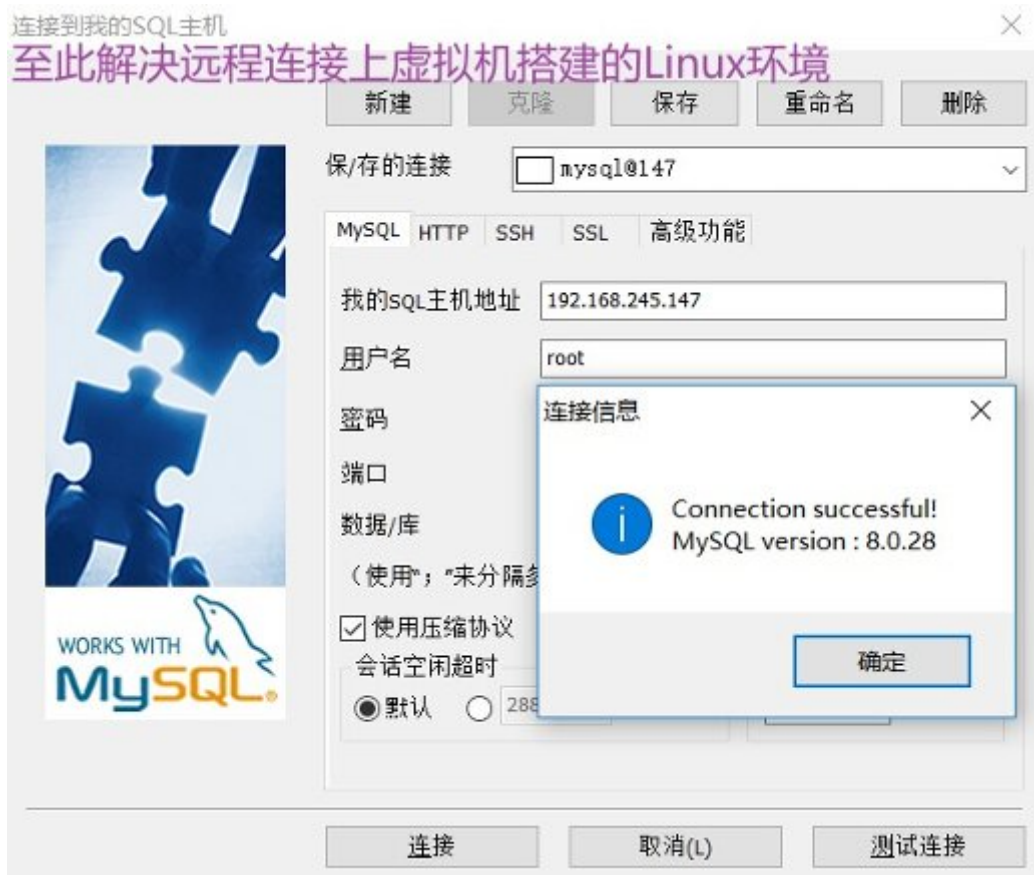
或者临时关闭防火墙测试

```
$ systemctl stop firewalld.service
```

2.6、测试远程登录

开启防火墙，加入了3306/tcp协议规则，加入了mysql服务规则。设置了以前旧的密码缓存验证规则mysql_native_password，解决 caching_sha2_password 验证插件无法被加载的问题。

```
mysql> CREATE USER 'root'@'%' IDENTIFIED BY 'Mysql@123456'; -- 第一步创建用户  
mysql> GRANT ALL ON *.* TO root@'%'; -- 第二步授权  
mysql> ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY  
'Mysql@123456'; -- 第三步修改密码验证方式  
mysql> flush privileges; -- 第四步刷新权限
```



3、关于忘记密码解决方案

很多小伙伴估计都遇到过设置密码后，结果忘记密码了。本文的解决方案，完全适用目前最新版本MySQL8.0.28，**亲自测试验证过**。

参考[MySQL8.0官方文档](#)以及[stackoverflow解决方法](#)。结果兜兜转转回到了跳过登录密码权限验证，8.0版本之前的方法失效了，咱没跟上MySQL更新的步伐。

1、关闭MySQL服务:

```
$ systemctl stop mysqld
```

2、设置MySQL环境选项参数，跳过权限表验证

```
$ systemctl set-environment MYSQLD_OPTS="--skip-grant-tables"
```

或者在/etc/my.cnf文件中添加，是一样的效果。**最后记得去掉跳过验证**。Windows下在my.ini文件中加入skip-grant-tables。

```
[mysqld]
skip-grant-tables
```

3、使用了刚刚的设置启动mysql

```
$ systemctl start mysqld
```

4、登录到root

```
$ mysql -u root
```

5、使用命令更新root用户密码

```
mysql> UPDATE mysql.user SET authentication_string = PASSWORD('MyNewPassword')
-> WHERE User = 'root' AND Host = 'localhost';
mysql> FLUSH PRIVILEGES;
mysql> quit
```

a、这种方式可能行不通

```
UPDATE user SET authentication_string = PASSWORD('123456') WHERE User = 'root'
AND Host = 'localhost';
```

b、采取将密码先置空

```
update user set authentication_string = '' where user = 'root';
```

6、修改密码，解决方案，设置更强的密码规则即可

```
As mentioned my shokulei in the comments, for 5.7.6 and later, you should use
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY '123456';
Or you'll get a warning
```

MySQL8.0出现这种情况，请设置更安全的密码规则比如设置密码为：Mysql@123456，即可成功。

```
ERROR 1819 (HY000): Your password does not satisfy the current policy
requirements
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'Mysql@123456';
```

7、关闭mysql服务

```
$ systemctl stop mysqld
```

8、重置前面设置的mysql环境变量参数

```
$ systemctl unset-environment MYSQLD_OPTS
```

9、再次启动mysql

```
$ systemctl start mysqld
```

最终成功登录到mysql

```
$ mysql -u root -p
```

三、MySQL8.0使用

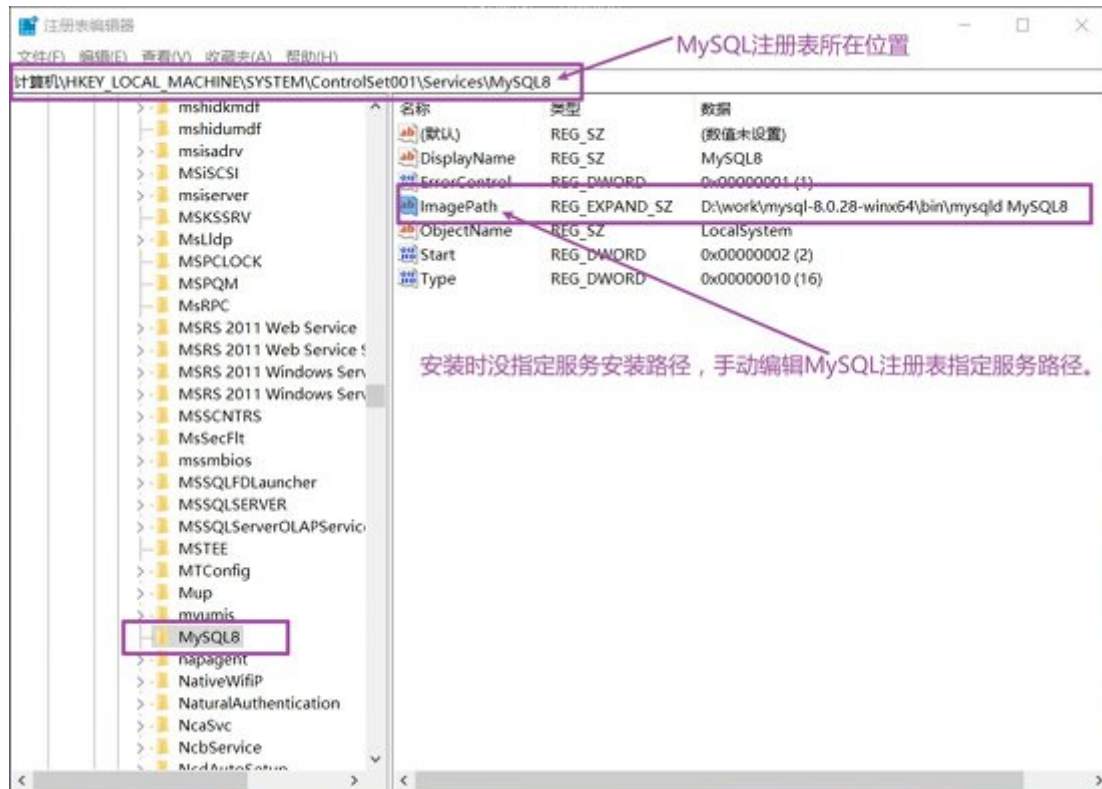
主要基于Windows10进行说明的，一些命令同样适用于Linux。

1、Windows多个MySQL服务实例共存

注意修改注册表路径，解决启动MySQL服务意外停止的情况，提示1067还是1068来着。

```
net start mysql8
net start mysql
```

为了测试演示最新版本，我将服务名改成了MySQL8。



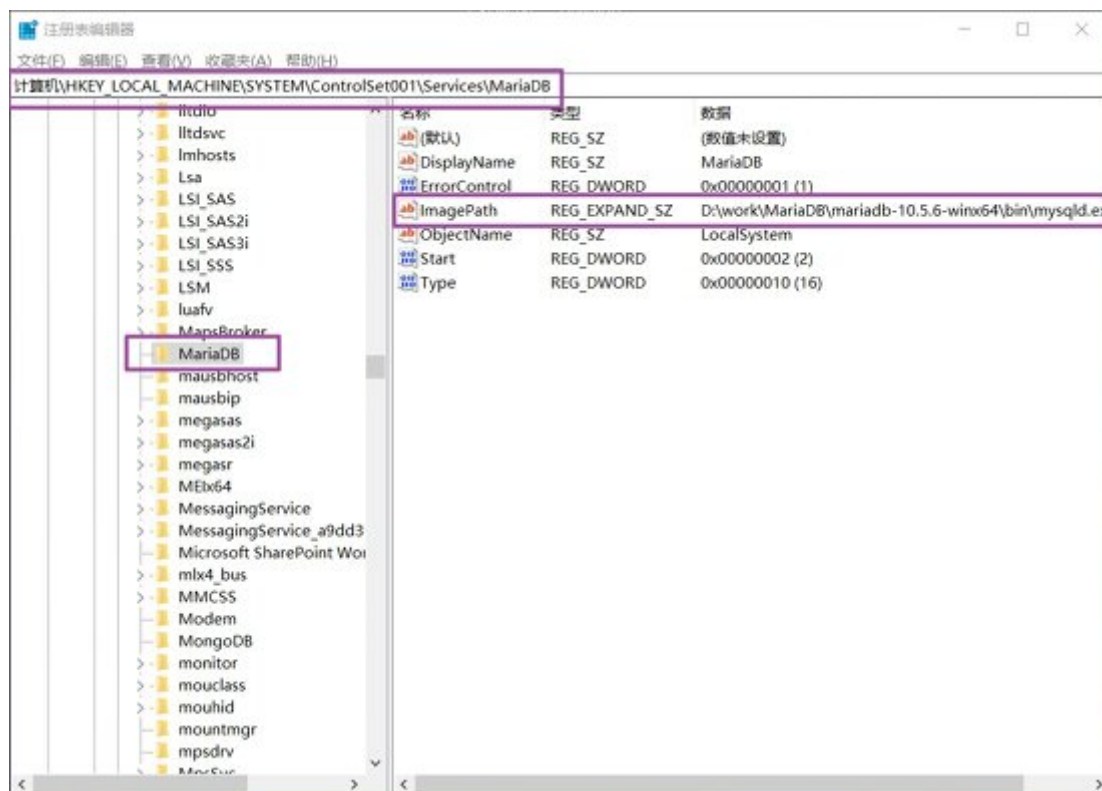
安装服务指定了服务名为MySQL8

计算机\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\MySQL8

或者是MySQL

计算机\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\MySQL

我之前安装的MariaDB10.5.6。我想继续使用MariaDB，又想体验最新版的MySQL8.0.28，选择这样处理。



1.1、登录并指定端口3307，默认为3306，我的MariaDB已经占用了3306端口。个人测试演示多个实例共存改了端口为3307。

注意：在Windows下使用cmd命令窗口以管理员身份运行登录，没有配置环境变量也没关系，切换到MySQL安装的bin目录下执行命令。

```
-- 第一步执行d:，切换到D盘
d:
-- 第二步执行cd命令，切换到个人安装mysql的bin目录下
cd D:\work\mysql-8.0.28-win64\bin
-- 执行登录命令，并指定端口
mysql -uroot -p -P 3307
-- 查询数据库版本
```sql
mysql> select version();
+-----+
| version() |
+-----+
| 8.0.28 |
+-----+
1 row in set (0.00 sec)
```
```



```

D:\work\mysql-8.0.28-win64\bin>mysql -uroot -p -P 3307 1、登录到MySQL，指定端口，默认为3306
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.28 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> select version(); 2、查看数据库版本，目前最新版本为8.0.28
+-----+
| version() |
+-----+
| 8.0.28    |
+-----+
1 row in set (0.00 sec)

```

总结一下：

- 第一步执行d:，切换到D盘；
- 第二步执行cd命令，切换到个人安装mysql的bin目录下；
- 第三步执行登录命令，并指定端口登录到mysql；
- 最后进行简单的交互，并查询数据库版本。

1.2、初步使用命令行模式进行交互

```

mysql> show databases; 1、查看初始化安装默认有哪些数据库
+-----+
| Database |
+-----+
| information_schema |
| mysql        |
| performance_schema |
| sys          |
+-----+
4 rows in set (0.00 sec)

mysql> create database TEST; 2、创建数据库TEST
Query OK, 1 row affected (0.01 sec)

mysql> use test; 3、切换数据库，并使用TEST数据库
Database changed
mysql> create table STUDY(ID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,NAME VARCHAR(16))ENGINE=InnoDB;
Query OK, 0 rows affected (0.02 sec) 4、创建一张STUDY表，并指定默认存储引擎为InnoDB

mysql> INSERT INTO STUDY VALUES(1,'梦梦');
Query OK, 1 row affected (0.01 sec) 5、在创建的STUDY表中插入一条演示数据

mysql> SELECT * FROM STUDY; 6、查询STUDY表，显示出我们插入的数据，基本的数据库操作就完成了。
+----+-----+
| ID | NAME |
+----+-----+
| 1  | 梦梦 |
+----+-----+

```

2、权限设置

2.1、参考官方文档

设置远程登录权限，以及密码校验规则，与安装数据库版本默认使用的默认认证插件有关

参考MySQL官方文档：<https://dev.mysql.com/doc/refman/8.0/en/upgrading-from-previous-series.html#upgrade-caching-sha2-password>

```
01、Authentication plugin 'caching_sha2_password' is not supported
02、Authentication plugin 'caching_sha2_password' cannot be loaded:
dlopen(/usr/local/mysql/lib/plugin/caching_sha2_password.so, 2):
image not found
03、Warning: mysqli_connect(): The server requested authentication
method unknown to the client [caching_sha2_password]
```

使用上面这种密码缓存验证算法。描述到验证插件不支持caching_sha2_password，不能被加载，服务连接请求提出警告认证方法无法识别客户端。**通俗一点解释**：在使用SQLyog、MySQL workbench等客户端连接时，密码验证规则是不被允许的。需要更换验证方式，或者其它方式解决验证。下面将会给出解决方案。

2.2、修改root用户密码

使用alter user语句修改用户密码：

```
ALTER USER 'root'@'localhost' IDENTIFIED BY '新密码';
```

2.3、创建普通用户并授权（开发人员或者DBA使用的比较频繁）

初学者可以先忽略授权这一步，使用root用户将基本功练扎实。还没入门，就没这头皮发麻的授权模式给弄崩溃了，哈哈。

授权命令GRANT，撤销权限命令REVOKE。创建用户并授权参考官方文档：

<https://dev.mysql.com/doc/refman/8.0/en/roles.html>

创建角色：**CREATE ROLE命令**

```
CREATE ROLE 'app_developer', 'app_read', 'app_write';
```

授予角色权限：**GRANT命令**

```
GRANT ALL ON app_db.* TO 'app_developer';
GRANT SELECT ON app_db.* TO 'app_read';
GRANT INSERT, UPDATE, DELETE ON app_db.* TO 'app_write';
```

创建用户：**CREATE USER命令**

```
CREATE USER 'dev1'@'localhost' IDENTIFIED BY 'dev1pass';
CREATE USER 'read_user1'@'localhost' IDENTIFIED BY 'read_user1pass';
CREATE USER 'read_user2'@'localhost' IDENTIFIED BY 'read_user2pass';
CREATE USER 'rw_user1'@'localhost' IDENTIFIED BY 'rw_user1pass';
```

授权给创建的用户：**GRANT命令**

```
GRANT 'app_developer' TO 'dev1'@'localhost';
GRANT 'app_read' TO 'read_user1'@'localhost', 'read_user2'@'localhost';
GRANT 'app_read', 'app_write' TO 'rw_user1'@'localhost';
```

你还可以在my.ini或者my.cnf配置文件中指定设置：

```
[mysqld]
mandatory_roles='role1,role2@localhost,r3@%.example.com'
```

同样可以在命令模式下使用SET命令设置：

```
SET PERSIST mandatory_roles = 'role1,role2@localhost,r3@%.example.com';
```

检查角色dev1的权限，检查权限比较多，我就不一一列举。详情可以参考官方文档。

```
mysql> SHOW GRANTS FOR 'dev1'@'localhost';
+-----+
| Grants for dev1@localhost |
+-----+
| GRANT USAGE ON *.* TO `dev1`@`localhost` |
| GRANT `app_developer`@`%` TO `dev1`@`localhost` |
+-----+
```

创建用户并授权，授权全部权限：

```
CREATE USER 'old_app_dev'@'localhost' IDENTIFIED BY 'old_app_devpass';
GRANT ALL ON old_app.* TO 'old_app_dev'@'localhost';
```

锁定用户：锁定：LOCK，盲猜解锁就是UNLOCK

```
ALTER USER 'old_app_dev'@'localhost' ACCOUNT LOCK;
```

授权给新的开发账号权限，授权部分权限：

```
CREATE USER 'new_app_dev1'@'localhost' IDENTIFIED BY 'new_password';
GRANT 'old_app_dev'@'localhost' TO 'new_app_dev1'@'localhost';
```

以上提供官方文档进行参考，与其东找找西找找，不如翻阅官方文档更直接更准确。咱缺的不是学习的途径，而是缺乏学习的方法。

3、测试创建用户

3.1、创建普通用户并授权远程登录

创建一个普通用户test

```
mysql> CREATE USER 'test'@'localhost' IDENTIFIED BY '123456';
Query OK, 0 rows affected (0.00 sec)
```

授予用户test在本地（localhost）的权限，**只给查询权限**（SELECT），授权所有（ALL）

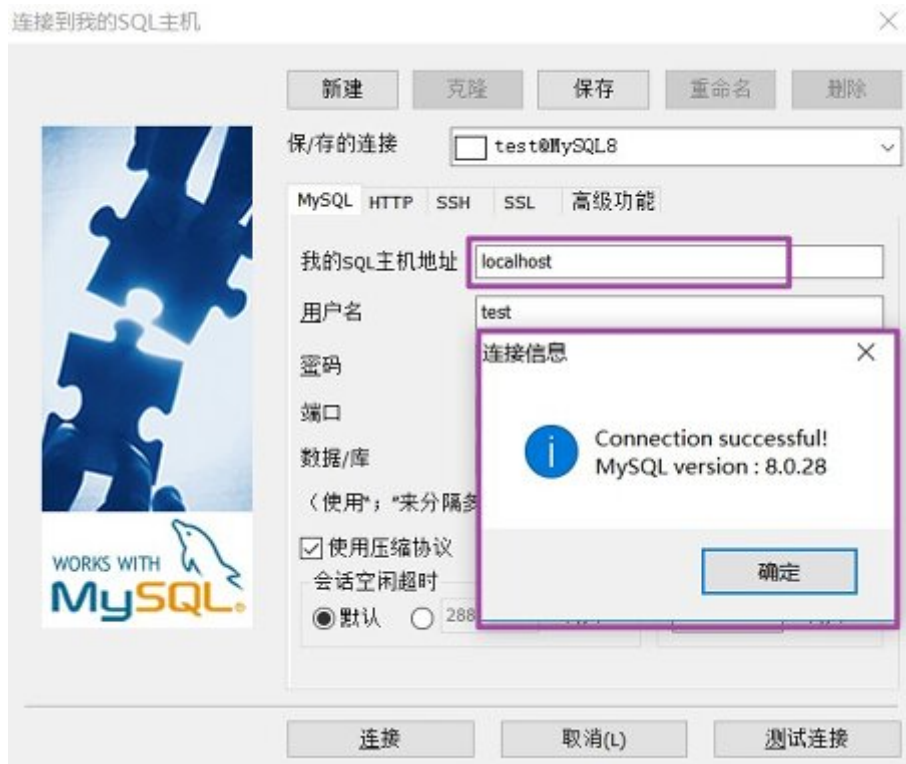
```
mysql> GRANT SELECT ON *.* TO test@'localhost';
Query OK, 0 rows affected (0.01 sec)
```

3.2、给root用户授权

创建root用户授权给所有IP都能登录，以及修改密码缓存认证方式。

```
mysql> CREATE USER 'root'@'%' IDENTIFIED BY 'Mysql@123456';
mysql> GRANT ALL ON *.* TO root@'%';
mysql> ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY
'Mysql@123456';
mysql> flush privileges;
```

在第三方工具中验证登录结果，在localhost下可以登录成功：



目前只给了查询 (select) 权限，验证插入 (insert) 权限：

```
mysql> insert into study values(7,'美柑');
ERROR 1142 (42000): INSERT command denied to user 'test'@'localhost' for table
'study'
```

这是在SQLyog工具下进行验证的，建议初学者多在命令行窗口下进行练习，熟能生巧。



3.2、授权root用户远程登录，MySQL8.0授权方式

用户授权，在MySQL8.0版本中变得更加严格，以前MySQL5.6或者5.7版本中可以执行授权的方式有了变化。经过个人亲测，操作如下。

MySQL8.0授权方式，记得使用flush privileges刷新权限

```
mysql> CREATE USER 'root'@'%' IDENTIFIED BY 'Mysql@123456'; -- 第一步创建用户
mysql> GRANT ALL ON *.* TO root@'%'; -- 第二步授权
mysql> ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY 'Mysql@123456'; -- 第三步修改密码验证方式
mysql> flush privileges; -- 第四步刷新权限
```

修改密码认证方式（8.0默认使用的是sha2算法缓存认证），**第一种解决方案如下**，这只是其中一种解决方案，亲测有效。

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '123456';
ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY '123456';
```

第二种解决方案：在my.ini或者my.cnf配置文件加入如下配置，重启服务并加载配置文件。经过测试没有生效，似乎没有读取到配置文件，但奇怪的是我设置的3307端口和默认存储引擎以及编码格式是生效的。（在官网看到的解决方案）

```
[mysqld]
default_authentication_plugin=mysql_native_password
```

MySQL8.0官方文档默认设置的**认证缓存算法是caching_sha2_password**

```
ALTER USER user(用户) IDENTIFIED WITH caching_sha2_password BY 'password';
```

MySQL8.0之前的授权方式（5.6或者5.7都支持这种方式授权）


```
GRANT ALL PRIVILEGES ON *.* TO '你的用户名'@'你的IP地址' IDENTIFIED BY '设置的密码'
WITH GRANT OPTION;
```

示例：授权root户，所有IP都可连接。

```
GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' IDENTIFIED BY '123456' WITH GRANT
OPTION;
```

刷新权限

```
flush privileges;
```

4、如何高效的使用自带官方文档

登录到MySQL8，指定3307端口，或者使用默认端口登录。

```
mysql -uroot -p -P 3307
mysql -uroot -p
```

使用帮助命令，以? create contents形式查找系统帮助命令。

```
? create contents;
? create user;
? create database;
? create table;
? select;
? insert;
? update;
? delete;
URL: https://dev.mysql.com/doc/refman/8.0/en/select|insert|update|delete.html
```

在使用本地的帮助文档时，你会发现系统自动提示了官方文档的地址 <https://dev.mysql.com/doc>。

示例：查询创建表的帮助命令? create table只展示了一部分内容。

```
mysql> ? create table
Name: 'CREATE TABLE'
Description:
Syntax:
CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tbl_name
    (create_definition,...)
    [table_options]
    [partition_options]
data_type:
    (see https://dev.mysql.com/doc/refman/8.0/en/data-types.html)
index_type:
    USING {BTREE | HASH}
index_option: {
    KEY_BLOCK_SIZE [=] value
    | index_type
    ...
}
URL: https://dev.mysql.com/doc/refman/8.0/en/create-table.html
```

- Name：查看的帮助命令名称
- Description：描述
- Syntax：示例
- data_type：支持的数据类型
- index_type：可以使用的索引类型

我只列举了部分进行说明，更详细的可以自己测试。

在创建用户、数据库以及建表和字段全部采取的大写，因为在Linux和Unix下对大写敏感的，并不是MySQL本身对大小写敏感。

1、创建数据库

```
CREATE DATABASE TEST;
USE TEST;
```

2、创建表，可以通过ENGINE指定表的存储引擎，mysql5.6以及之后的版本默认为InnoDB存储引擎。

```
CREATE TABLE STUDY(
    ID INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
    NAMES VARCHAR(64) NOT NULL
)ENGINE=MyISAM;
```

3、插入数据

```
INSERT INTO STUDY VALUES(1, 'mysql目前最新版本mysql8.0.28');
```

4、查询数据

```
SELECT * FROM STUDY;
```

5、修改数据

```
UPDATE STUDY S SET S.NAMES='mysql默认的存储引擎是InnoDB' WHERE S.ID=1;
```

6、删除全部数据

```
DELETE FROM STUDY;
```

至此基本的创建用户、创建数据库、增删改查都会使用了。

四、MySQL连接工具

做了超链接，方便去官网获取。

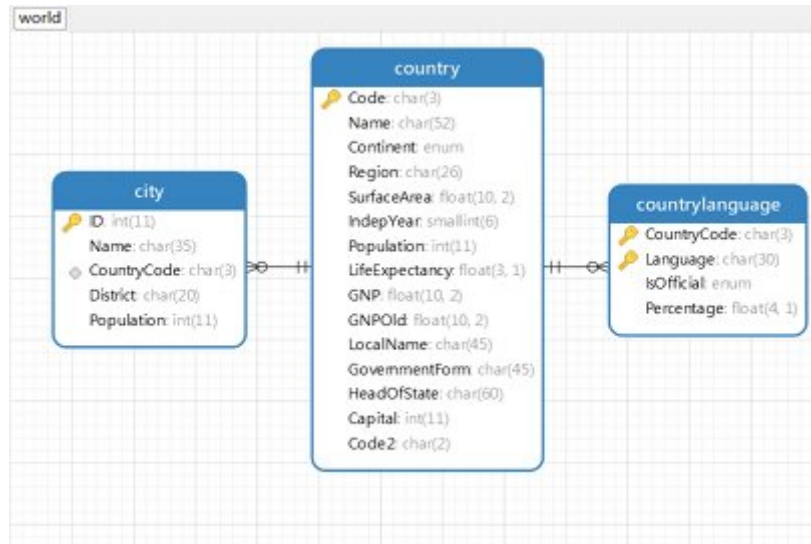
- phpMyAdmin
- MYSQL workbench
- SQLyog

推荐几个比较常用的工具：[phpMyAdmin](#)、[SQLyog](#)、[MySQL Workbench](#)、Navicat可视化工具进行连接操作。工具的使用是其次的，更重要的在于对MySQL命令语句的运用。

tips：包含了SQLyog，还整理了部分安装包以及MySQL官方提供的**sakila**、**world**示例哟！

链接：<https://pan.baidu.com/s/11glZKxoTG5BCCcoXdVJRg>。提取码：ntu7

给出一个使用Navicat逆向生成的示例数据库world的模型：



如果真的要使用到建物理模型：推荐你学习Sybase PowerDesigner设计工具的使用，而且需要了解关系数据库设计遵循的三范式。现在数据库设计最多满足3NF，普遍认为范式过高，虽然具有对数据关系更好的约束性，但也导致数据关系表增加而令数据库IO更易繁忙，原来交由数据库处理的关系约束现更多在数据库使用程序中完成。

五、MySQL之JDBC

1、官方connector-j

MySQL8.0的maven安装JDBC：<https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-installing-maven.html>

JDBC连接驱动管理：<https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-usagenotes-connect-drivermanager.html>

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

// Notice, do not import com.mysql.cj.jdbc.*
// or you will have problems!

public class LoadDriver {
    public static void main(String[] args) {
        try {
            // The newInstance() call is a work around for some
            // broken Java implementations
            Class.forName("com.mysql.cj.jdbc.Driver").newInstance();
        } catch (Exception ex) {
            // handle the error
        }
    }
}

Connection conn = null;
...
try {
    conn = DriverManager.getConnection("jdbc:mysql://localhost/test?" +
                                       "user=root&password=123456");
    // Do something with the Connection
}
```

```

...
} catch (SQLException ex) {
    // handle any errors
    System.out.println("SQLException: " + ex.getMessage());
    System.out.println("SQLState: " + ex.getSQLState());
    System.out.println("VendorError: " + ex.getErrorCode());
}

```

2、JDBC测试连接MySQL8.0数据库

2.1、maven配置

设置pom.xml配置文件，使用MySQL最新的版本8.0.28进行连接测试。maven的镜像仓库，可以使用阿里的镜像源地址。

```

<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.28</version>
</dependency>

<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.28</version>
    <exclusions>
        <exclusion>
            <groupId>com.google.protobuf</groupId>
            <artifactId>protobuf-java</artifactId>
        </exclusion>
    </exclusions>
</dependency>

```

2.2、编写Java代码

使用编辑器sts (Spring Tool Suite4或者IDEA)

创建普通的maven项目或者springboot项目，然后配置pom.xml。

目的：使用纯JDBC测试，或者ORM框架mybatis、JPA、或者hibernate都行，最终达到对数据库进行最基本的增删改查。

```

package com.example.demo.dao;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

public class TestConnMySQL8 {

```

```

    public static void main(String[] args) throws ClassNotFoundException,
SQLException {
        TestSQLConnMySQL();
    }
    private static final Logger log =
LoggerFactory.getLogger(TestConnMySQL8.class);
    //初始化参数
    static Connection conn = null;
    static PreparedStatement ps = null;
    static ResultSet rs = null;
    /**
     * @throws SQLException
     * @throws ClassNotFoundException
     */
    private static void TestSQLConnMySQL() throws SQLException,
ClassNotFoundException {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            /**
             * 1. 获取连接参数url,username,password,默认端口是3306
             * MySQL: url ="jdbc:mysql://127.0.0.1:3306/test";
             */
            /** MySQL拼接url */
            String url = "jdbc:mysql://192.168.245.147:3306/TEST?
useUnicode=true&characterEncoding=utf-8";
            String username = "root";
            String password = "Mysql@123456";
            //获取连接
            conn = DriverManager.getConnection(url, username, password);
            if(null != conn) {
                log.info("connect database success...");
            }else {
                log.error("connect database failed...");
            }
            //查询数据库
            String sql = "SELECT * FROM STUDY";
            // 3.通过preparedStatement执行SQL
            ps = conn.prepareStatement(sql);
            // 4.执行查询,获取结果集
            rs = ps.executeQuery();
            // 5.遍历结果集,前提是你的数据库创建了表以及有数据
            while (rs.next()) {
                //对应数据库表中字段类型Int使用getInt, varchar使用getString
                System.out.println("ID:" + rs.getInt("ID"));
                System.out.println("姓名: " + rs.getString("NAMES"));
            }
        } finally {
            // 6.关闭连接 释放资源
            rs.close();
            ps.close();
            conn.close();
        }
    }
}

```

在sts编辑工具连接并返回测试结果


```
Console Problems Progress Servers
sterminated> TestConnMySQL8 [Java Application] D:\work\Java\FF\jdk8\jdk1.8.0_151\bin\javaw.exe (2022年2月2日 下午7:12:43)
Wed Feb 02 19:12:43 CST 2022 WARN: Establishing SSL connection without server's identity
19:12:45.004 [main] INFO com.example.demo.dao.TestConnMySQL8 - connect database success.
连接成功!!!
ID:1
姓名: mysql默认的存储引擎是InnoDB
```

总结

以上就是本次MySQL8.0.28安装与使用的全部内容，希望能对你的工作与学习有所帮助。感觉写的好，就拿出你的一键三连。在公众号上更新的可能要快一点，目前还在完善中。**能看到这里的，都是帅哥靓妹**。如果感觉总结的不到位，也希望能留下您宝贵的意见，我会在文章中进行调整优化。



原创不易，转载也请标明出处和作者，尊重原创。不定期上传到github或者gitee。认准龙腾万里sky，如果看见其它平台不是这个ID发出我的文章，就是转载的。**MySQL系列文章**：《**MySQL开发篇，存储引擎的选择真的很重要吗？**》已经上传至github和gitee仓库SQL-study。个人github仓库地址，一般会先更新PDF文件，然后再上传markdown文件。如果访问github太慢，可以使用gitee进行克隆。

tips：使用hexo搭建的静态博客也会定期更新维护。

作者：[龙腾万里sky](#)