

实验一 MySQL 的安装与使用

一、实验目的

1. 熟悉 MySQL 的数据库的安装与配置过程
2. 熟悉 Navicat 及 workbench 软件

二、实验原理和内容

1. 安装 MySQL 并配置能启动
2. 安装 Navicat 软件并连接 MySQL（如果版权问题，可以不安装本软件）
3. 安装 workbench 软件，并使用该软件，完成数据库的 ER 图的创建，和数据库的创建。

三、实验要求

1. 下载 mysql，根据要求步骤完成 mysql 数据库的安装。
 - (1) 只需要安装 MySQL server 即可。
 - (2) 安装 mysql 系统服务：mysqld -install（如果安装程序已经在 Windows 服务里添加了 MySQL 服务，就不需要做这一步）
 - (3) 将 MySQL 的安装路径添加到环境变量 path 里；
 - (4) 配置 MySQL。（有些不需要配置，可以尝试一下。如果 MySQL 正常启动，就不要配置）在 my.ini 里配置：

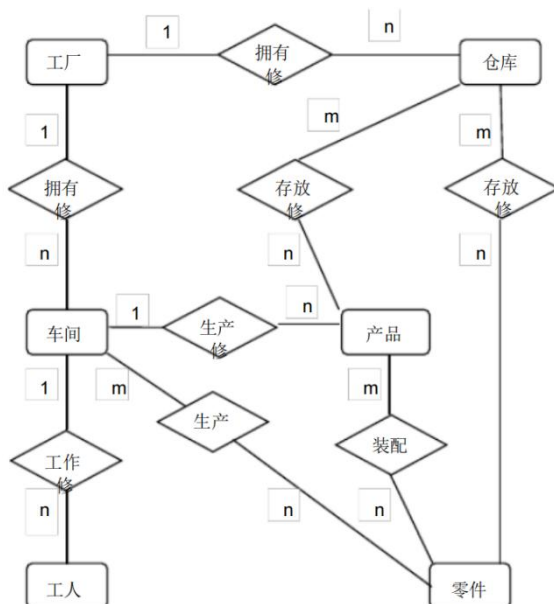
```
[mysqld]
Basedir=c:/mysql5.7
Datadir=c:/mysql5.7/data
Port=3306
Skip-grant-tables（用来跳过密码登录）
```
 - (5) 初始化数据库（创建 my.ini 配置文件后，数据库文件目录 C:\mysql5.7\data 还没创建，就需要这一步来完成 MySQL 的初始化功能）

```
Mysqld -- initialize-insecure
```
 - (6) 其他命令
启动/停止 MySQL 服务： net start mysql/net stop mysql
登录 MySQL： mysql -u root -p 密码 [数据库名]
退出数据库： exit/quit
2. 数据库管理工具软件 Navicat
 - (1) 安装软件，正常安装即可。
 - (2) 启动好 MySQL 数据库，保持 cmd 界面不关闭；
 - (3) 打开 Navicat 软件，创建连接，按照提示创建即可。如果 MySQL 数据库 root 身份登录时没有密码，这里创建的时候，密码框为空即可。创建连接，即完成。
3. 使用 workbench 软件，完成下面数据库的创建以及 ER 图。

工厂（包括厂名和厂长名）需要建立数据库，用来管理存储以下信息：

 - ① 一个厂内有多个车间，每个车间有车间号、车间主任姓名、地址和电话；
 - ② 一个车间内有多个工人，每个工人有职工号、姓名、年龄、性别和工种；

- ③ 一个车间生产多种产品，产品有产品号 and 价格；
- ④ 一个车间生产多种零件，一个零件也可能由多个车间制造。零件有零件号、重量和价格，
- ⑤ 一个产品由多种零件组成，一种零件也可装配在多种产品内，
- ⑥ 产品与零件均存入仓库中，厂内有多个仓库，仓库有仓库号、仓库保管员姓名和电话。



四、实验过程

1. 下载 mysql server 官方软件，版本号为 5.7.31 版本。官网下载地址为：
<https://downloads.mysql.com/archives/installer/>
2. 下载之后双击安装就会出现下图 1 所示的选项界面，选择 Choosing a Setup Type 里面的 Custom 模块，选择结束后点击 Next。

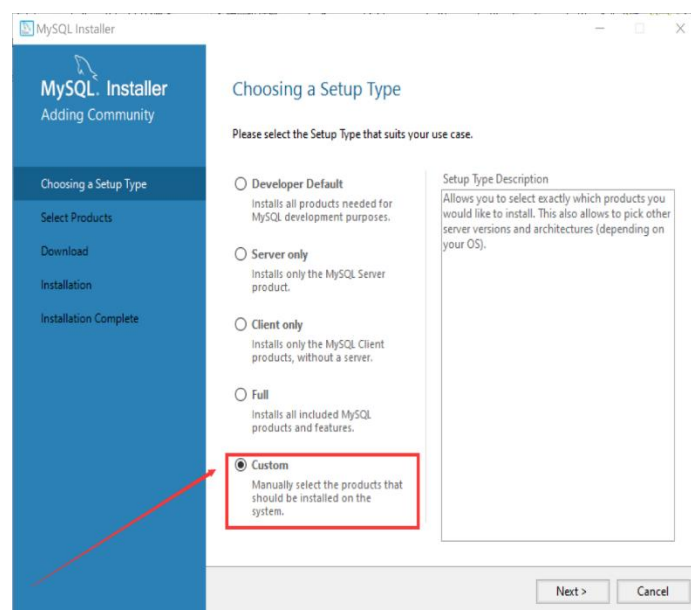


图 1 选择数据库安装类型

3. 出现下面的页面，按下图顺序依次点击，这里是根据自己电脑的配置来安装适配自己电脑的 MySQL Server。点击之后可以选择修改路径，在本人电脑中，就把该应用安装到

D:\APP\MySql 目录下，以便日后寻找和使用。

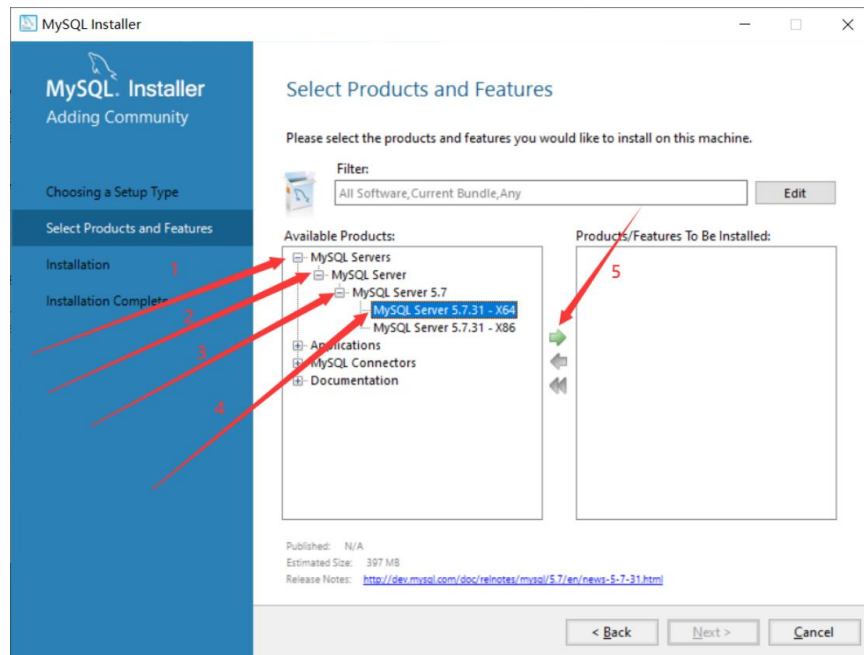


图 2 选择安装适配机型版本的数据库

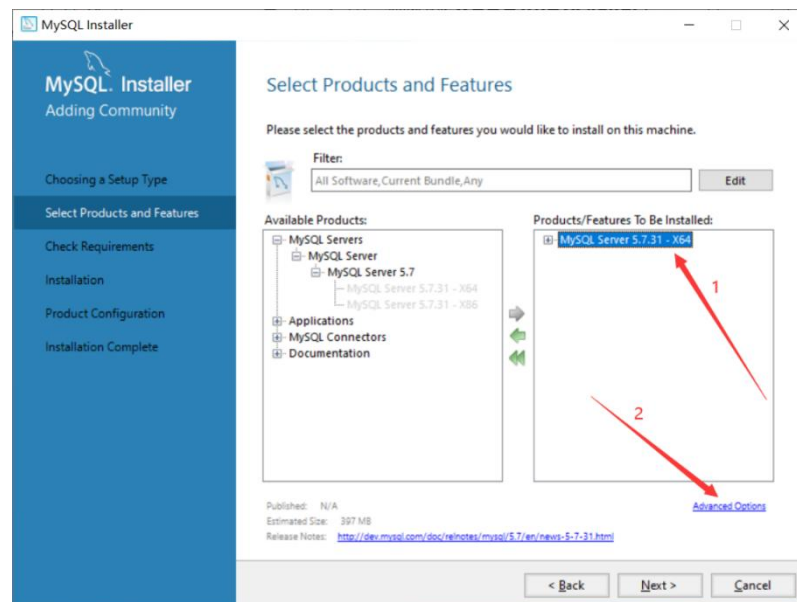


图 3 选择安装数据库的位置

4. 3 完成后会出现图 4 界面，继续点击 Next，选择成功之后单击 Next 完成下载请求。在此之后一直点击 Next，这里就不再一一列出图标了。

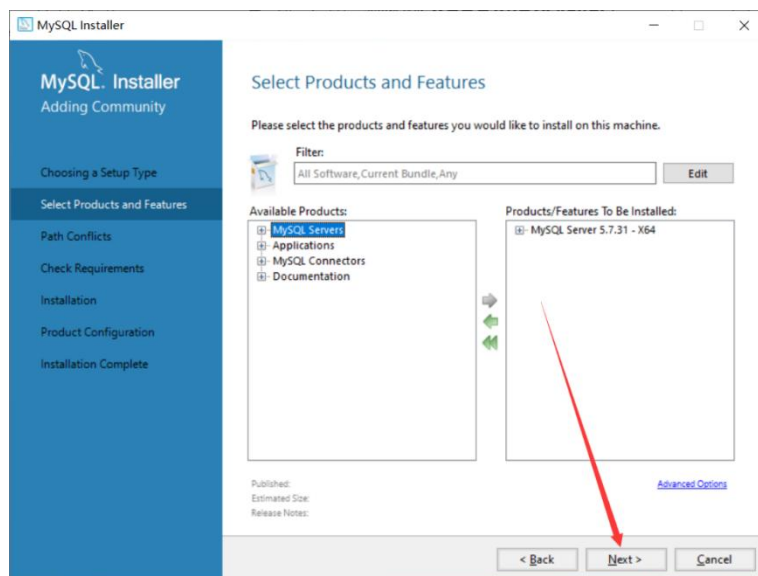


图 4 数据库下载安装

5. 当安装进行到以下页面时，需要为数据库设置密码。

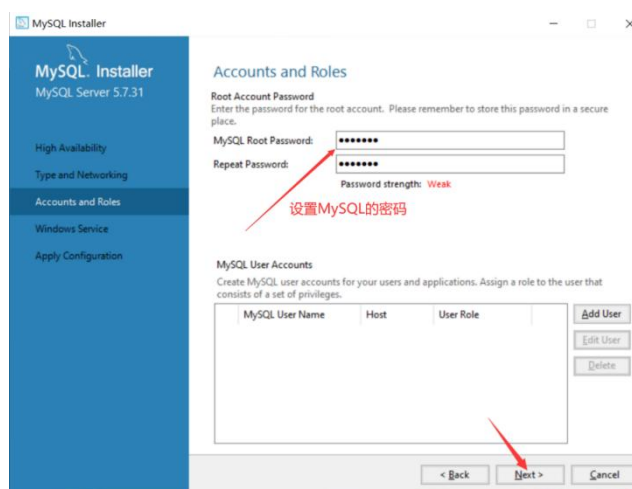


图 5 为数据库设置密码

6. 设置密码后一直点击 next，直到出现图 6 说明数据库安装成功，单击 Finish。

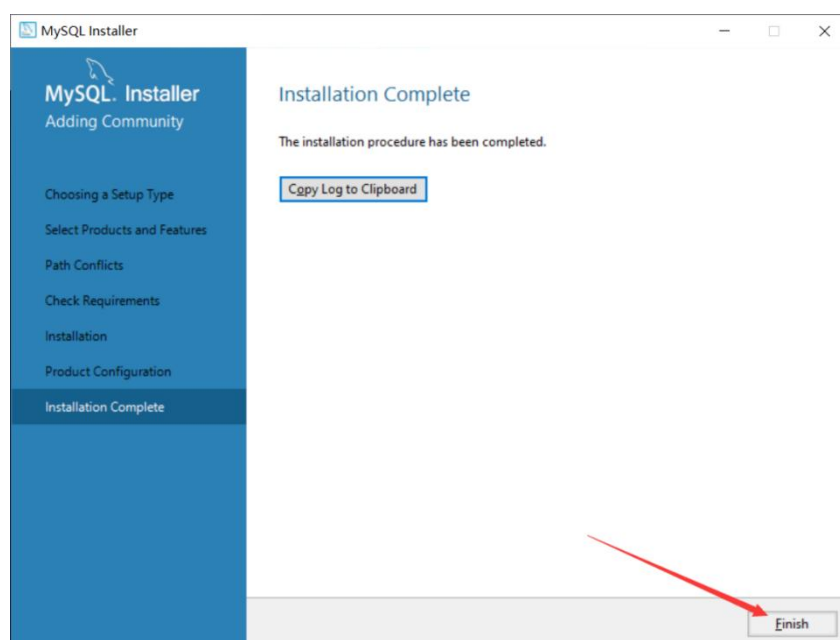


图 6 数据库安装完成界面

4.2 安装 Navicat 和 MySQL Workbench

从 Navicat 官方下载 15.0 版本（稳定版）的 .exe 文件，下载后双击安装。从 MySQLBench 官方下载 8.0 版本的 .exe 安装包文件，下载后双击安装。

4.3 在 Dos 命令行里开启数据库

1. 首先用管理员身份打开 command。键入 net start mysql，之后登录数据库，键入 mysql -u root -p+密码。
2. 打开 Navicat 软件，然后连接本机数据库就能正常进行建立模型等一系列操作了。

4.4 用 MySQLWorkbench 生成 ER 图

步骤 1：打开 MySQLWorkbench 软件，然后点击左上角 File 之后创建一个新的模型，点击 New Model，如下图所示。

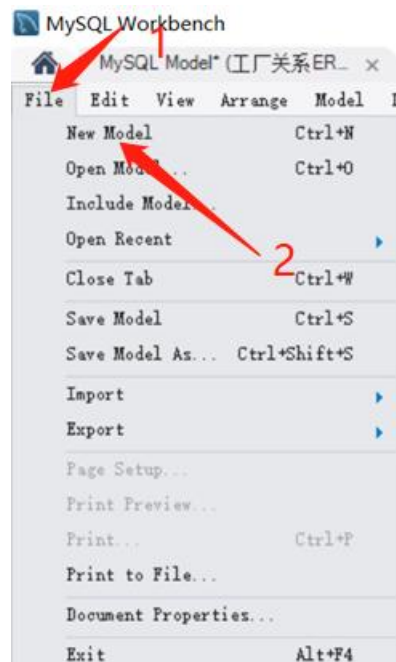


图 7 创建 ER 图步骤 1

步骤 2：完成步骤 1 之后会出现如图 8 所示的页面，在这个页面里点击 Add Diagram 就能创建画布，之后再在画布上绘制 ER 图即可。

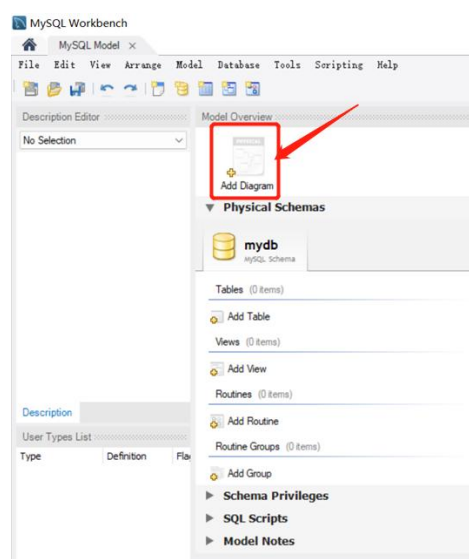


图 8 创建 ER 图步骤 2

步骤 3: 打开画布页面之后, 如下图 9 所示, 就能创建了. 在下图 9 中, 创建实体可以点击第一个箭头所指的快捷键, 点击之后立即创建一个表格, 之后可以根据该表格的各个属性进行修改, 修改成需要的若干实体以及它们的属性。当所有实体画完之后, 再通过第二个箭头定义不同实体之间的关系。值得注意的是, 在定义关系的时候一定要看清两个实体之间到底是一对多还是多对一的关系, 反映在 ER 图上是否和实际上的不相符。



图 9 创建 ER 图步骤 3

4.4 实验结果

用 MySQL Workbench 画出的 ER 图如下图 10 所示。

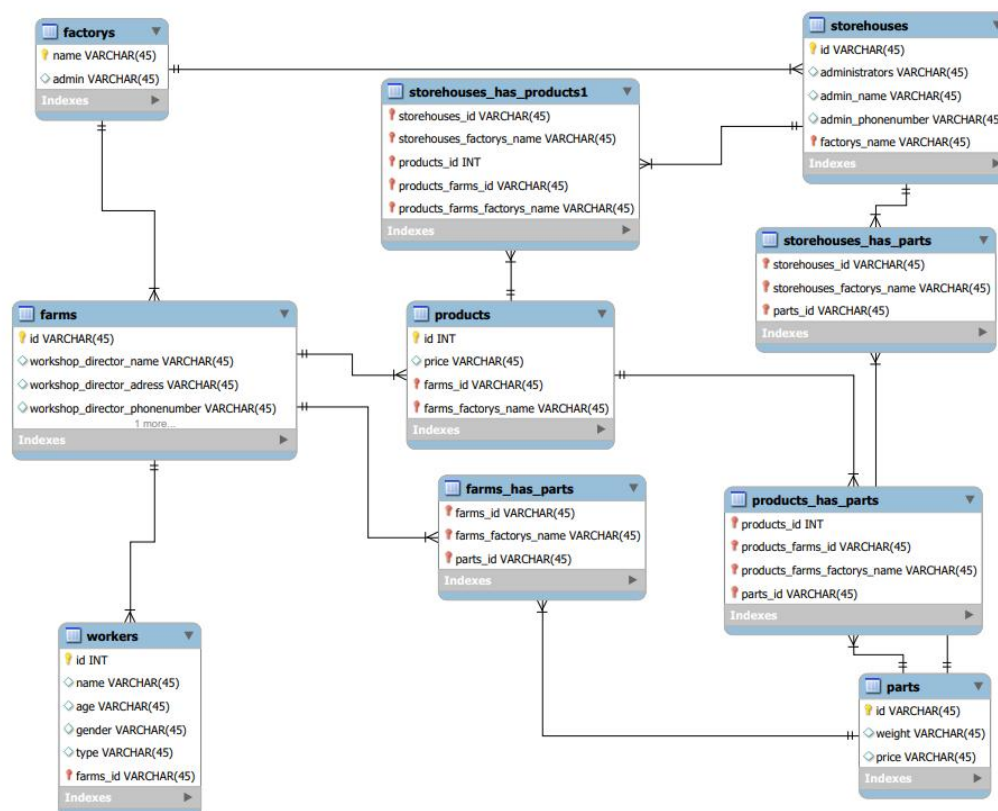


图 10 MySQL Workbench 生成的 ER 图

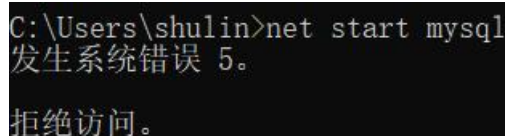
结果如上图所示, 其中不含下划线的方框表示实体, 其中的每个方框的第一行 (蓝色部分) 是表头, 代表该表的名称。每张表的第二个行表示的属性为主键, 在表中用黄色的标识符表示。不同实体之间连接的线表示关系。在上图中, 一个工厂可以有多个车间, 因此工厂和车间是一对多的关系; 一个车间可以有多个员工工作, 因此车间和员工之间的关系为一对多的关系; 一个工厂可以同时拥有多个仓库, 因此工厂和仓库是一对

多的关系。一个车间能生产多种零件，同时一种零件也能有多个车间生产，因此车间和零件之间的关系为多对多的关系；一个车间能生产出多种产品，因此车间和产品之间的关系为一对多的关系；一个仓库能存放多种零件，一种零件也能存放在多个仓库里，因此零件和仓库之间的关系为多对多的关系。在工厂表里面，工厂名字是唯一识别该工厂的标签（国家规定注册公司不得重名），因此工厂名字可以作为工厂表的主键。同理，车间的序号是唯一区别车间的标签，因此车间序号可以作为车间的主键。工人的工号是唯一区别工人的属性，因此工号可以作为工人的主键。仓库号是区别仓库和其他仓库的唯一标识，因此仓库号属性是仓库表的主键。产品号是产品与其他产品的唯一标识，因此产品号是产品表格的主键。零件号是零件区别于其他零件的唯一属性，因此将零件号作为零件表的主键。

五、实验总结

本次实验应该是本学期最简单的实验，但是由于我们是刚接触数据库，一些知识还是不够牢固，费了九牛二虎之力才搭建好运行环境，结果又不知哪里的原因，数据库在第二次运行时就启动失败，系统报错。所谓万事开头难，也正是如此。下面总结一下在本次实验中遇到的几个问题，以供老师遇到同样问题时参考。

问题 1：输入 `net start mysql` 系统报错如下图 11 所示。



```
C:\Users\shulin>net start mysql
发生系统错误 5。

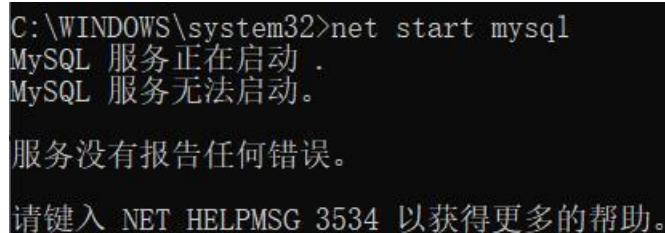
拒绝访问。
```

图 11 问题 1 报错

问题解析：这个是我安装好 mysql server 之后打开 cmd，结果报给系统错误。

问题解决：经过多方查证和询问，我终于明白是因为我没有以管理员身份打开 cmd，普通用户没有开启系统数据库的权限，需要管理员身份。因此我总结：有关系统报错拒绝访问的内容，一般都需要给管理员权限，否则不能访问。

问题 2：解决 1 问题时，输入 `net start mysql` 系统报错如下图 12 所示。



```
C:\WINDOWS\system32>net start mysql
MySQL 服务正在启动。
MySQL 服务无法启动。

服务没有报告任何错误。

请键入 NET HELPMSG 3534 以获得更多的帮助。
```

图 12 问题 2 报错

问题解析：此时已经给予了管理员权限了，但是仍旧无法正常启动 Mysql。

问题解决：经过多方查询和搜索，我在 CSDN 上看到了相应的解决方案。首先需要访问电脑各个端口，查看数据库的端口是否已经被其他应用占用。在 4.1 中配置的数据库的端口默认为 3306，当我查询到 3306 端口被 mysql.exe 占用时，说明端口是正常的，那么就是文件数据的问题了，这种情况下一般是 DATA 文件夹的问题，可以先将其删除掉，之后重新对数据库初始化，命令为：`mysqld -initialize`，此时会在原来删除的地方重新生成一个 DATA 文件。此时再重新以管理员身份打开 command 输入 `net start mysql`，问题解决。结果如下图 13 所示：

```
C:\WINDOWS\system32>net start mysql
MySQL 服务正在启动 .
MySQL 服务已经启动成功。
```

图 13 问题 2 解决后截图

问题 3: 键入 `mysql -u root -p` 需要输入密码，但是在此之前并没有设密码或者之前设置的密码不能使用。如下图 14 所示：

```
C:\WINDOWS\system32>mysql -u root -p
Enter password: *****
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)

C:\WINDOWS\system32>mysql -u root -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: NO)
```

图 14 问题 3 报错

问题分析：可能是在安装数据库的时候，随即设置了密码，因此打不开。

问题解决：经过多方询问，最终找到了答案。在 windows 环境下，需要在任务管理器里面关掉 `mysql.exe` 服务，之后用管理员身份打开 DOS 窗口，输入 `mysqld --skip-grant-tables`，这段命令的作用是启动 MySQL 服务的时候跳过权限表认证。之后当前 DOS 页面不能再使用了，需要再打开一个窗口，输入 `mysql`，成功后会返回 “`mysql>`”，在这之后需要连接权限数据库，命令为 `use mysql`；此时就可以更改密码了，命令为 `update user set password=password("密码") where user="root"`；最后一步刷新权限，命令为 `flush privileges`；之后就能正常使用数据库了。解决成功后的图片如图 15 所示。

```
C:\WINDOWS\system32>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.31-log

Copyright (c) 2000, 2020, 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>
```

图 15 问题 3 解决截图

总结一下此次实验，总体来说难度还是有的，但难度并不集中在于对概念模型或物理模型的理解上，而是在于对环境的配置上，和问题的解决上。实验过程中遇到的问题远远不止以上三个问题，但是以上三个问题是最普遍的问题，其解决问题的方式是经得起验证的，但是我还是坚持熬了个大夜配完了环境，也能重新运行好了。主要花的时间在于对 Navicat 的破解，毕竟官方正版价格高不可攀。通过本次实验，我深刻意识到数据库概念模型实际上是现实世界到机器世界的一个中间层次。数据库概念模型用于信息世界的建模，是现实世界到信息世界的第一层抽象，是数据库设计人员进行数据库设计的有力工具，也是数据库设计人员和用户之间进行交流的语言。具体体现在，我先在 `powerdesigner` 上画出了概念模型，然后通过软件的编译自动能生成对应的物理模型。在概念模型上，我仅仅在上面定义了实体以及它们之间的关系，就能生成对应的物理模型图，从一个抽象的概念变成了一个句体实际的图，不得不佩服工程师的伟大。之后又在 `MySQLWorkbench` 上面直接做出了 ER 图，和前者生成的差别不大。