

数据库系统及应用

数据库实验环境准备

2023.4.9

1 实验要求

- 完成 MySQL 的安装和使用，为后续实验提供支持。
- 可选 Windows 或 Linux 等系统安装。
- 成功运行提供的 [DB.sql](#) 文件则实验通过。
- 可选使用 python 连接数据库。
- 可选使用 Visual Studio Code 集成开发环境连接数据库。
- 安装 PowerDesigner。

2 WINDOWS 安装使用 MYSQL

2.1 安装 MySQL

2.1.1 下载安装

1. [睿客网盘下载](#)
2. 或者[官网下载](#)

2.1.2 双击 *mysql* 文件安装，可选 *Developer Default* 或者 *Custom* 模式，见图 [2.1](#)。

- 选择 Developer Default，将安装 mysql 开发所需的所有组件
- 选择 Custom，则需要手动选择应安装的组件，我们必选 MySQL Servers 和 Applications 下的 server、workbench、shell，然后从 MySQL Connectors 下选择一个或多个连接器用于从代码访问 mysql 数据库 (python、C++ 等，至少一个，用于后期实验的使用)。没列出的组件可随开发需求安装，后续也可以通过 MySQL Installer 进行组件的增减。

注意：安装时不满足要求的组件是灰色的，鼠标放置在上面也会出现相应的提示，我们应勾选黑色的可安装组件，并且尽量选择版本号相符的组件

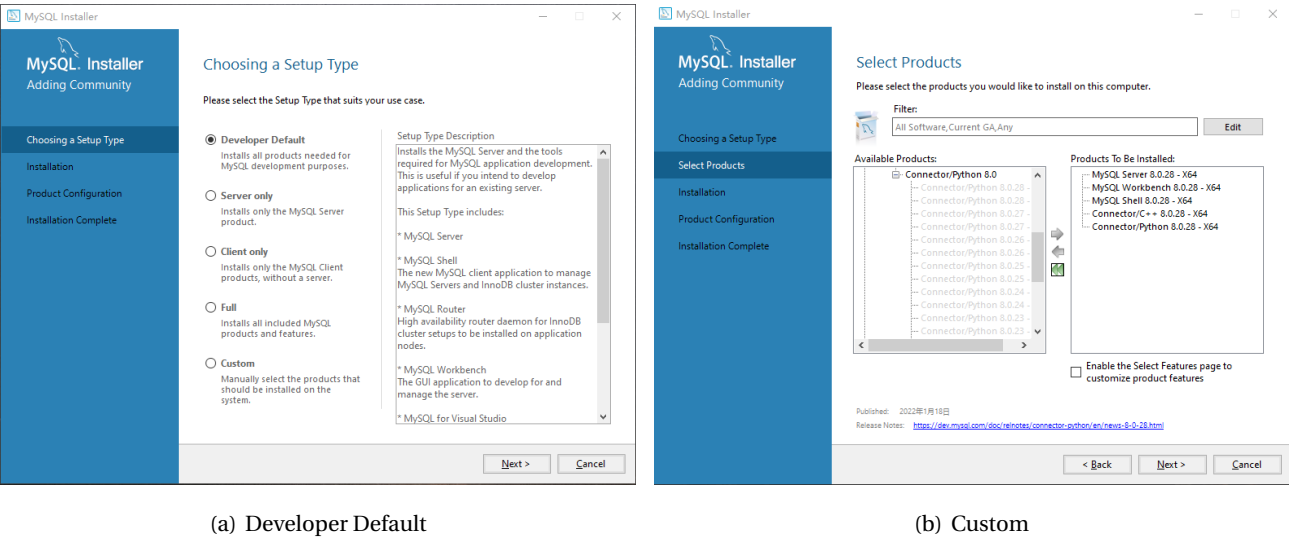


Figure 2.1: 模式选择

2.1.3 检查依赖

- 选择好安装组件后，会进行依赖检查，如果所需的库和软件都满足，则不会出现该界面
- 如果所需的依赖不满足，则会出现图 2.2 所示界面，根据提示安装相应依赖即可。比如图中因为勾选了安装 MySQL for Visual Studio 组件，但系统未安装 Visual Studio，故会出现提示，如果不使用 Visual Studio，直接进入下一步即可，将不会安装 MySQL for Visual Studio 组件；也可前往下载安装 Visual Studio。

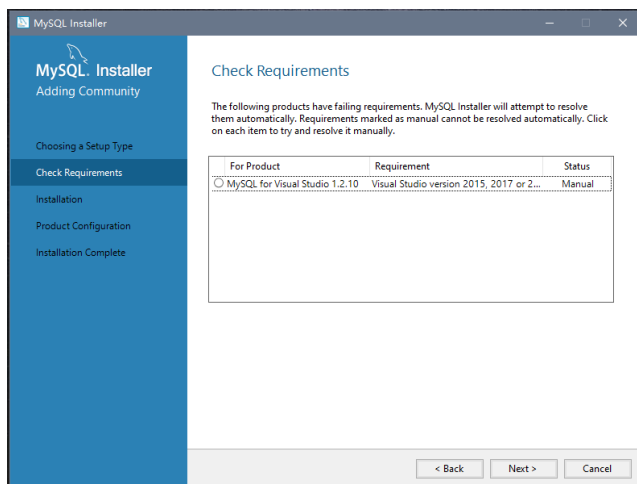
2.1.4 安装

- 确定好需要安装的产品后，点击 Execute 进行安装。这时会进入安装程序进度界面，安装需要一些时间，请保持网络连接。见图 2.3。

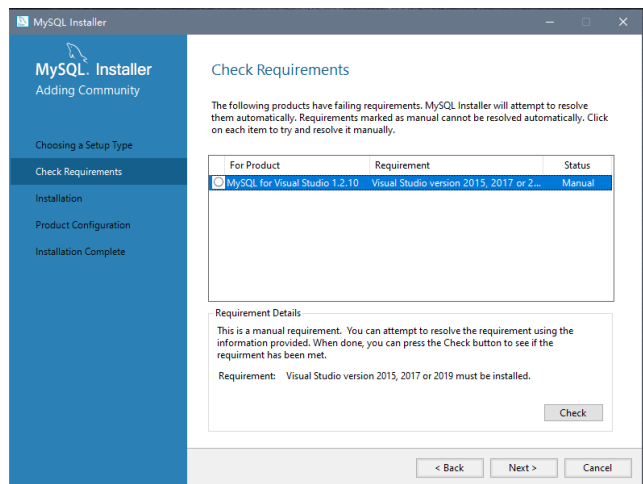
2.1.5 配置

进入产品配置环境，大部分产品配置按默认方式点击 NEXT 即可。见图 2.4。

- Type and Networking: 默认设置
- Authentication Method: 默认设置
- Accounts and Roles: 设置一个 **root** 用户密码，添加一个管理员，设置用户名和密码，后期可以在 workbench 和 shell 中登录这两个用户。

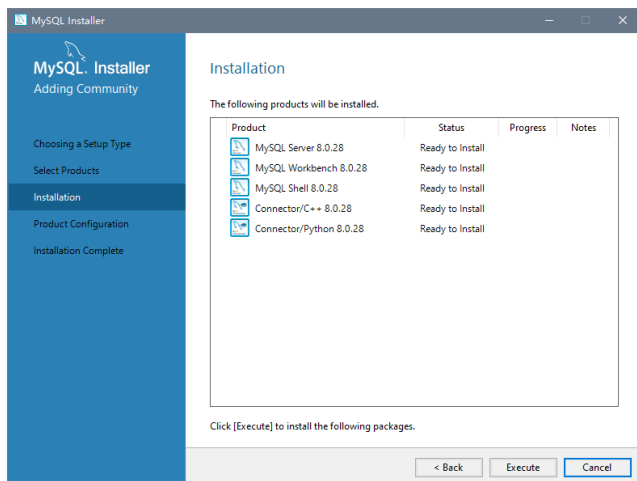


(a) 检查依赖

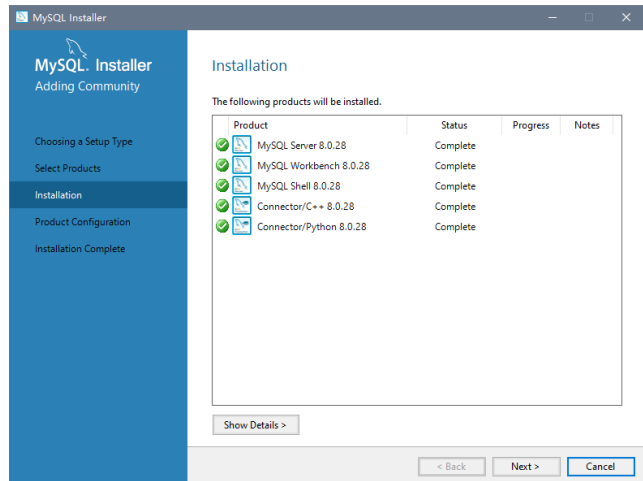


(b) 依赖信息

Figure 2.2: 依赖



(a) 点击 Execute

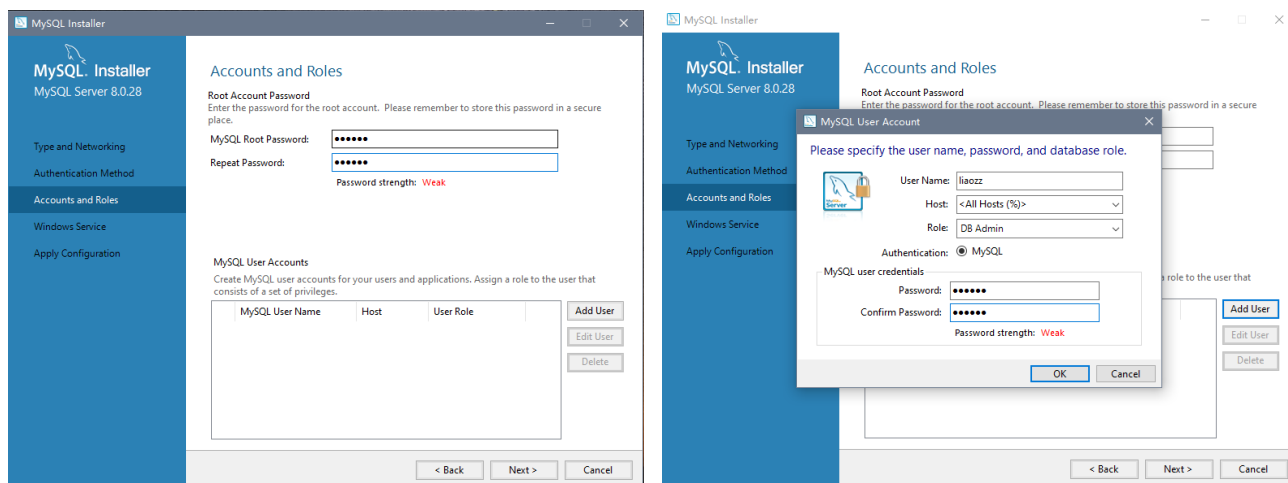


(b) 组件安装完毕

Figure 2.3: 安装

- **Windows Service:** 建议使用默认设置，或仅修改设置为不开机自启动。
- **Apply Configurations:** 点击 Execute，再点击 finish。

如果安装了其他组件需要配置，一般默认设置即可。



(a) 设置 root 密码

(b) 增加用户

Figure 2.4: 配置

2.1.6 安装完毕，点击 *finish*。后续可以使用 *MySQL Installer* 增减组件。

2.2 使用 MySQL

2.2.1 添加环境变量

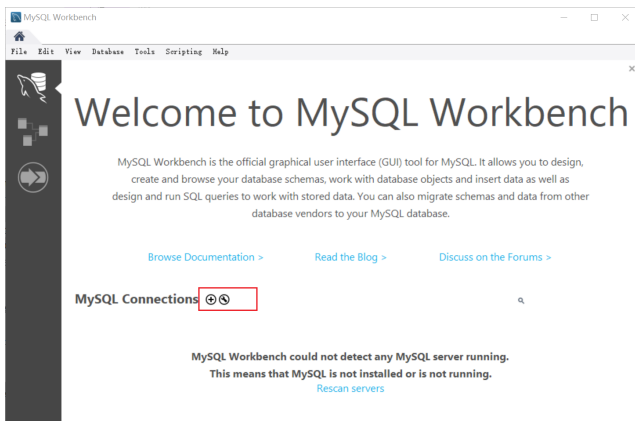
- 将 MySQL 默认安装位置"C:\Program Files\MySQL\MySQL Server 8.0\bin"添加到环境变量，方便今后再 cmd 中使用 mysql。

2.2.2 使用 *workbench* 连接 *mysql*，见图 2.5

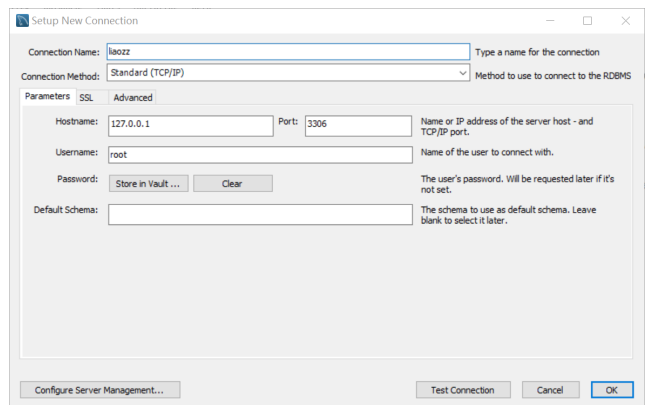
1. 打开 *workbench*，在开始界面点击 MySQL Connections 旁的加号新建一个连接。
2. 输入一个 Connection Name 后，点击 OK。
3. 点击新建的连接。
4. 输入对应 User 的密码。

2.2.3 *workbench* 运行 *mysql* 语句

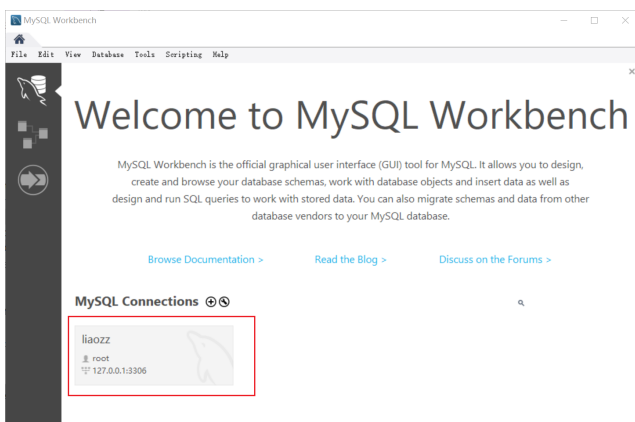
1. 打开DB.sql文件或在空白界面输入文件中的 mysql 语句，出现图 2.6结果即为安装测试成功。
2. 注：如果打开 DB.sql 文件时闪电箭头未亮，说明未连接至数据库，点击 Database, 再点击下方的 Connect to Database 完成连接。



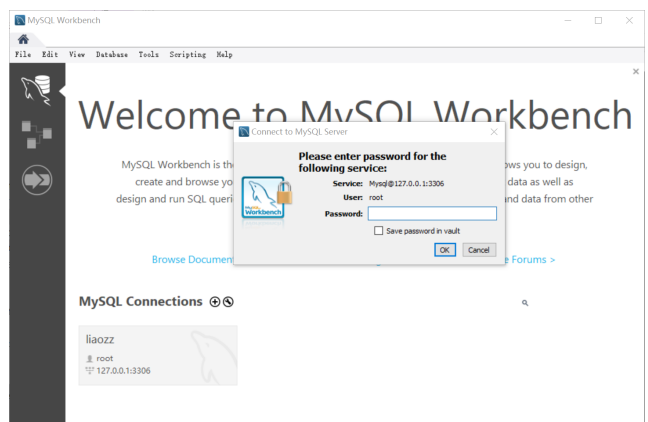
(a) 1



(b) 2

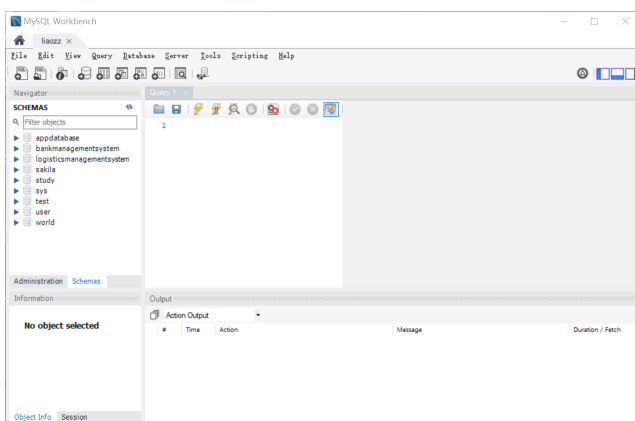


(c) 3

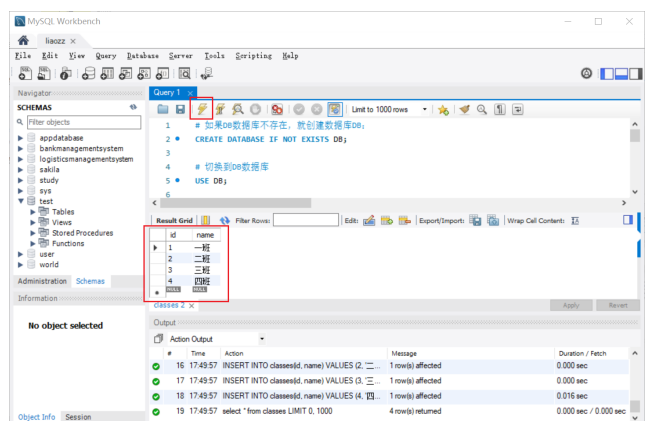


(d) 4

Figure 2.5: workbench



(a) 键入



(b) 点击闪电运行

Figure 2.6: 运行结果

2.2.4 使用命令行连接 *mysql*

1. 打开一个命令行终端，输入"`mysql -u <your user name> -p`" 命令，然后输入对应的密码。
2. 进入 *mysql* 环境，可以在终端中执行 *mysql* 语句。见图 2.7

```
C:\Users\liaozi>mysql -u root -p
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
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> use DB;
```

(a) 进入 *mysql*

```
mysql> use DB;
Database changed
mysql> show tables;
+-----+
| Tables_in_db |
+-----+
| classes      |
+-----+
1 row in set (0.01 sec)

mysql> select * from classes;
+----+-----+
| id | name |
+----+-----+
| 1  | 一班 |
| 2  | 二班 |
| 3  | 三班 |
| 4  | 四班 |
+----+-----+
4 rows in set (0.00 sec)
```

(b) 运行 *mysql* 语句

Figure 2.7: 运行结果

2.3 更多教程。注：如果菜鸟教程链接点击无法进入，可自行搜索。

- 菜鸟教程 [MySQL 安装](#)。
- [更详细安装教程](#)
- [MySQL 基本命令](#)

3 LINUX 上安装 MySQL

3.1 Ubuntu

- 执行"`sudo apt-get install mysql-server`" 命令。
- 具体请参阅[Ubuntu 安装 MySQL 教程](#)。

3.2 CentOS

- 具体参阅[菜鸟教程 MySQL 安装](#)。

4 MAC 版安装

- 具体参阅[MySQL 安装 \(Mac 版\)](#)。

5 安装 POWERDESIGNER

1. 下载PowerDesigner 安装包。
2. 基本一路 Next 即可安装成功，插件可全部勾选。

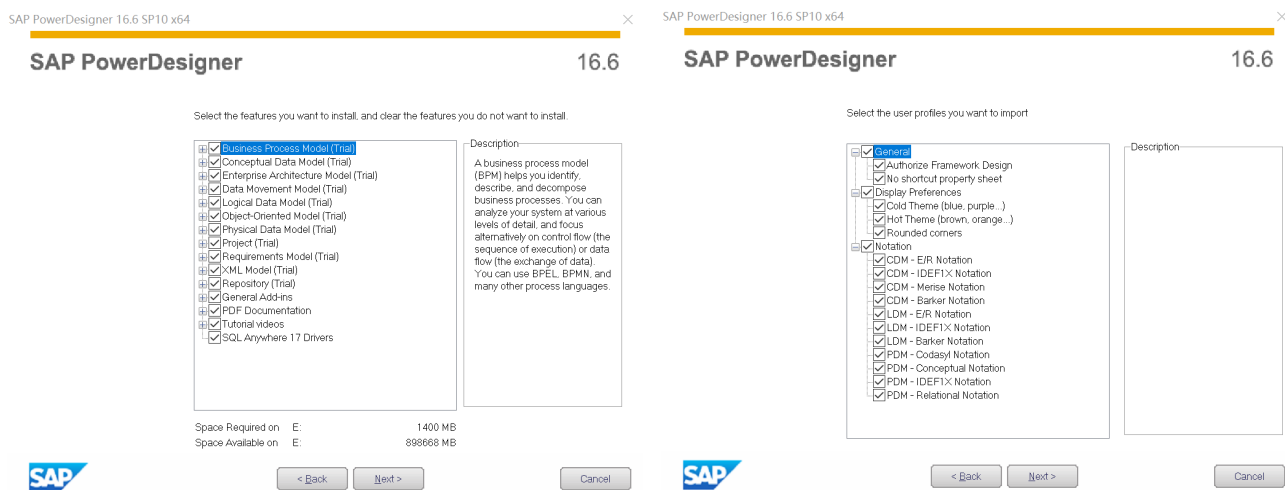


Figure 5.1: 勾选了全部插件

6 选做：使用 PYTHON 连接数据库

1. MySQL 安装了 Connector/Python。
2. 使用"pip install mysqlclient" 安装 MySQLdb 库。（前提是安装了 pip）
3. 运行DB.py文件，结果见图 6.1。

7 选做：使用 VISUAL STUDIO CODE 连接数据库

1. 安装 Visual Studio Code。
2. 安装插件 MySQL、MySQL Syntax、VSCode-Database。
3. 连接数据库，使用快捷键 Ctrl+Shift+P，选择：SQL:Connect to MySQL/PostgreSQL Server，填写账号等信息。
4. 测试连接，打开 DB.sql 文件，右键执行"Run MySQL Query"。



```
DB.py x
DB.py > ...
1  #encoding=utf-8
2  import MySQLdb
3
4  # 打开数据库连接 (url,username,password,database)
5  db = MySQLdb.connect(("localhost","root","1234","db",charset='utf8'))
6
7  # 获得操作游标
8  cursor = db.cursor()
9
10 # 执行SQL语句
11 cursor.execute("select * from classes;")
12
13 # 提取数据
14 data = cursor.fetchall()
15
16 for d in data:
17     print(d)
18
19 # 关闭数据库连接
20 db.close()

终端 问题 输出 调试控制台
PS C:\Users\lab1> python .\DB.py
(1, '一班')
(2, '二班')
(3, '三班')
(4, '四班')
PS C:\Users\lab1>
```

Figure 6.1: python 连接数据库

8 代码

Listing 1: DB.sql

```
# 如果DB数据库不存在，就创建数据库DB：
CREATE DATABASE IF NOT EXISTS DB;

# 切换到DB数据库
USE DB;

# 删除classes表（如果存在）：
DROP TABLE IF EXISTS classes;

# 创建classes表：
CREATE TABLE classes (
  id BIGINT NOT NULL AUTO_INCREMENT,
  name VARCHAR(100) NOT NULL,
  PRIMARY KEY (id)
) DEFAULT CHARSET=utf8;

# 插入classes记录：
INSERT INTO classes(id, name) VALUES (1, '一班');
INSERT INTO classes(id, name) VALUES (2, '二班');
INSERT INTO classes(id, name) VALUES (3, '三班');
INSERT INTO classes(id, name) VALUES (4, '四班');

select * from classes;
```

Listing 2: DB.py

```
#encoding=utf-8
import MySQLdb

# 打开数据库连接 (url, username, password, database)
db = MySQLdb.connect("localhost", "root", "1234", "db", charset='utf8')

# 获得操作游标
cursor = db.cursor()

# 执行SQL语句
cursor.execute("select * from classes;")
```

```
# 提取数据
data = cursor.fetchall()

for d in data:
    print(d)

# 关闭数据库连接
db.close()
```