

北京邮电大学数据库系统实验报告

实验名称: 数据库实验平台的安装 数据库创建与维护实验

计算机科学与技术系 2015211306班

魏晓

学号:2015211301

实验目的

- 通过对MySQL数据库的安装和简单使用；
 - 了解安装MySQL数据库的软硬件环境和安装方法；
 - 熟悉MySQL数据库相关使用；
 - 熟悉MySQL数据库的构成和相关工具；
 - 通过MySQL数据库的使用来理解数据库系统的基本概念
 - 通过创建数据库、并进行相应的维护，了解并掌握MySQL数据库数据库的创建和维护的不同方法和途径，并通过这一具体的数据库理解实际数据库所包含的各要素。
-

实验平台及环境

实验平台:MySQL 14.14

运行环境:Mac OS High Sierra 10.13(17A405)

交互式界面:MySQL WorkBench 6.3.10

```
Xiaos-MacBook-Pro:~ weixiao$ mysql --version
```

```
mysql Ver 14.14 Distrib 5.7.20, for macos10.12 (x86_64) using EditLine wrapper
```

实验原理

- 数据库是长期储存在计算机内、有组织的、可共享的大量数据集合。使得数据按一定的数据模

型组织、描述和储存，可为各种用户共享。数据可管理系统（DBMS）是位于用户和操作系统之间的一层数据管理软件。数据库系统（DBS）是指在计算机系统中引入数据库后的系统构成。

- 数据库系统对硬件资源的要求：
 - 足够大的内存：操作系统、DBMS的核心模块、数据缓冲区、应用程序
 - 足够大的外存：磁盘、操作系统、DBMS、应用程序、数据库及其备份
 - 较高的通道能力，提高数据传送率
 - 数据库对软件资源的要求：
 - 操作系统或汉字操作系统
 - 数据库管理系统和主语言系统
 - 应用开发工具软件：报表生成器、表单生成器、查询和视图设计器等
 - 应用系统和数据库

实验步骤

MySQL 数据库下载安装

- 具体步骤
 - 从<https://dev.mysql.com/downloads/>下载MySQL

MySQL Community Downloads

MySQL Community Server (GPL)

(Current Generally Available Release: 5.7.21)

MySQL Community Server is the world's most popular open source database.

[DOWNLOAD](#)

- 然后再在<https://dev.mysql.com/downloads/workbench/>下载WorkBench

MySQL Workbench 6.3.10

Select Operating System:

[Looking for previous GA versions?](#)

macOS

macOS (x86, 64-bit), DMG Archive

6.3.10

97.5M

[Download](#)

(mysql-workbench-community-6.3.10-macos-x86_64.dmg)

MD5: 4ad59ce1e00ab51fe33e23131cda29ce | [Signature](#)

- 在MacOS上安装并运行MySQL成功安装之后,再在终端中输入

```
mysql -u wx -p;  
#wx是我创建的新账户  
enter password:  
#输入密码
```

- 得到如下结果:(成功在服务器登陆此账户)

```
[Xiaos-MacBook-Pro:~ weixiao$ mysql -u wx -p  
[Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 4547  
Server version: 5.7.20 MySQL Community Server (GPL)  
  
Copyright (c) 2000, 2017, 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> █
```

- mysql 退出账户

```
[mysql> quit  
Bye  
Xiaos-MacBook-Pro:~ weixiao$ █
```

- 关闭服务器

```
mysqladmin -u root -p shutdown  
# 退出服务器
```

交互式工具连接数据库

- 交互式工具的连接配置

Setup New Connection

Connection Name:

在这里输入自定义名称

Type a name for the connection

Connection Method:

Standard (TCP/IP)

Method to use to connect to the RDBMS

Parameters

SSL

Advanced

Hostname:

127.0.0.1

Port:

3306

Name or IP address of the server host - and TCP/IP port.

Username:

root

Name of the user to connect with.

Password:

Store in Keychain ...

Clear

The user's password. Will be requested later if it's not set.

Default Schema:

The schema to use as default schema. Leave blank to select it later.

最后点击OK

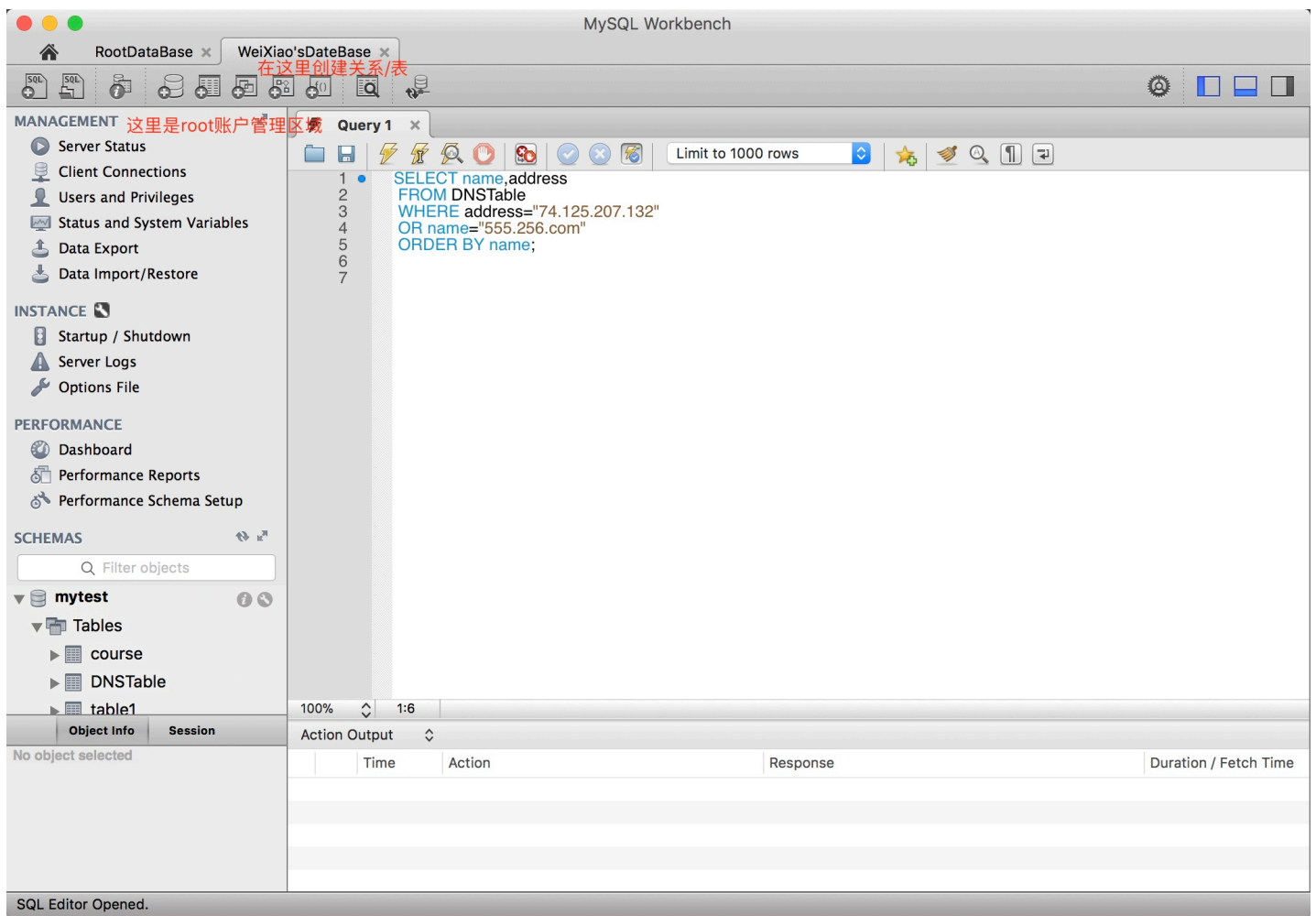
Configure Server Management...

Test Connection

Cancel

OK

- 各项功能



- 创建学生选课数据库

```
create database student-choice;  
#创建数据库  
LOAD DATA LOCAL INFILE '/path/pet.txt' INTO TABLE pet;  
#从特定文件夹导入文件  
source /test.sql  
#导入sql  
mysqldump -h localhost -u root -p test > /test.sql  
#导出sql
```

- 对数据库属性和参数进行查询、相应的修改和维护

```
mysqladmin -u root -p variables  
#查看所有属性
```

- 最大连接数

```
set GLOBAL max_connections=200
#session改为global
```

- 服务器端口

```
show global variables like 'port';
#查看
vi my.cnf
#修改
```

- 共享缓存数

```
select @@query_cache_type;
#查询缓存状态
set session query_cache_type=off;
# 关闭缓存
select @@global.query_cache_size;
#查询缓存大小
set @@global.query_cache_size=100000;
#设置最大缓存数
```

- 为DBA保留的连接数

```
show variables like 'max_connections';
#查看当前最大连接数
set global max_connections=1000;
#设置最大连接数为1000
```

- 死锁检测时间

```
show session variables like '%wait_timeout%';
#想在所有会话都有效可以设置全局配置，参数把session改为global就可以了
set global wait_timeout=90;
#设置检测时间为90
show global variables like '%wait_timeout%';
#重新查看
```

- 日志缓冲数

```
show variables like 'log_bin';
```

#查看日志是否开启

- 修改列名

```
alter table student change column bdate birthdate datetime;
```

#选择要修改的行,改编为另一个名字

#后面要有名称

- 修改数据库名

```
alter table customers rename custs;
```

#把数据库名字改编为custs

- 设置系统同步写 > 修改主MySQL配置

```
#/etc/my.cnf
```

```
log-bin = mysql-bin
```

主数据库端ID号

```
server-id = 1
```

#服务器标志号, 注意在配置文件中不能出现多个这样的标识, 如果出现多个的话mysql以第一个为准, 一组主从中此标识号不能重复。

```
log-bin=/home/mysql/logs/binlog/bin-log
```

#开启bin-log, 并指定文件目录和文件名前缀。

```
max_binlog_size = 500M
```

#每个bin-log最大大小, 当此大小等于500M时会自动生成一个新的日志文件。一条记录不会写在2个日志文件中, 所以有时日志文件会超过此大小。

```
binlog_cache_size = 128K
```

#日志缓存大小

```
binlog-do-db = adb
```

#需要同步的数据库名字, 如果是多个, 就以此格式在写一行即可。

```
binlog-ignore-db = mysql
```

#不需要同步的数据库名字, 如果是多个, 就以此格式在写一行即可。

```
log-slave-updates
```

#当Slave从Master数据库读取日志时更新新写入日志中, 如果只启动log-bin 而没有启动log-slave-updates则Slave只记录针对自己数据库操作的更新。

```
expire_logs_day=2
```

#设置bin-log日志文件保存的天数, 此参数mysql5.0以下版本不支持。

```
binlog_format="MIXED"
```

#设置bin-log日志文件格式为: MIXED, 可以防止主键重复。

2.从数据库的修改

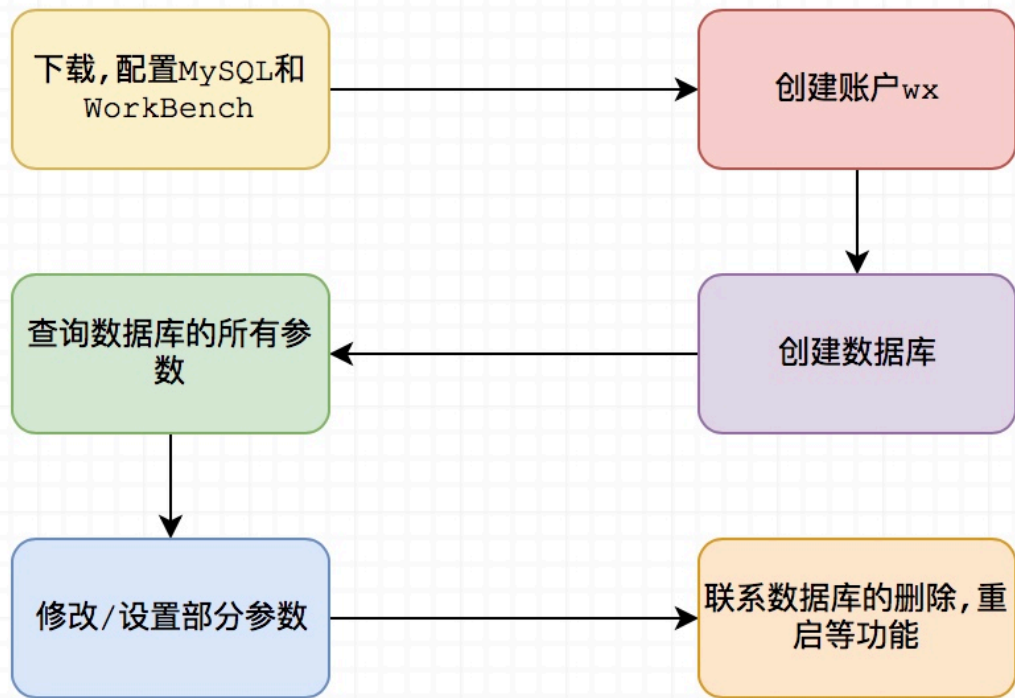
```
# 执行同步命令, 设置主数据库ip, 同步帐号密码, 同步位置
mysql>change master to master_host='192.168.1.2',
master_user='slave',master_password='111111',
master_log_file='mysql-bin.000009',master_log_pos=196;
# 开启同步功能
mysql>start slave;
```

至此就完成了数据库的设置同步写

- 练习数据库的删除等维护

```
SHOW DATABASES;
# 显示所有数据库
DROP DATABASE IF EXISTS tutorial_database;
# 删除数据库
```

实验步骤



实验结果及分析

- 结果1:成功在Mac OS上部署了MySQL服务器,并安装了WorkBench
- 结果2:创建了学生选课数据库
- 结果3:对数据库的相关属性进行了设置和优化,适应学期实验课程
- 结果4:联系了数据库的删除
- 结果5:练习和掌握了相关SQL语句,并且熟悉了在交互式界面创建和管理数据库

实验小结

- 图形化工具可简单方便的建立一个实用的数据库,使用“create database ”语句或者向导均可建立一个名为“student”的数据库,并可以通过多种方式配置该数据库的信息以及参数。
- 用“drop database ”语句以及右键提示均可删除该数据库。当使用“drop database ”语句删除数据库时,显示“无法删除数据库,因为该数据库当前正在使用。”可以将数据库设置为单用户模式,并马上中断其他用户连接,使用语句“ALTER DATABASE student SET SINGLE_USER

WITH ROLLBACK | IMMEDIATE; GO DROP DATABASE student”。

- 在这次实验中，分别尝试了在不同的操作系统环境中安装并操作SQL Server，并在过程中借助网络查询方式解决了很多关于系统不兼容方面的问题，最后搭建了较为稳定的实验平台。
- 通过进行安装，配置，创建、维护、删除等操作后，我对SQL Server数据库工具的了解程度有了一些提升，也对老师课上讲授的知识有了切实的体验，同时相关的数据库操作也让我熟悉了基本的Transact_SQL语句，为以后的实验打下基础。