**武汉大学国家网络安全学院**

**《数据库系统》实验指导书**

二○二二年三月

目 录

[实验一 MySQL环境搭建及SQL使用 1](#_Toc97020360)

[一、实验目的 1](#_Toc97020361)

[二、实验内容 1](#_Toc97020362)

[三、实验步骤 2](#_Toc97020363)

# 实验一 MySQL环境搭建及SQL使用

## 一、实验目的

1. 掌握数据库环境（LAMP，Linux+Apache+MySQL+PHP）的搭建过程。

2. 熟悉MySQL的基本操作（创建用户、登录、执行SQL、创建数据库、导入数据库、数据库完整性检查、数据库备份）。

3. 学会创建DDL脚本，包括创建、删除表，指定主键、候选键和外键；学会修改并展示表结构，学会创建、删除索引。

4. 掌握数据的增、删、查、改等操作，学会分析在操作过程中出现失败的原因；掌握视图的创建、删除和更新；理解视图消解机制，学会更新视图的操作并分析更新过程中出现失败的原因。

5. 掌握order by、group by、having等子句的使用；掌握各种谓词的使用；掌握集合函数的使用；掌握在时间字段上的查询。

6. 掌握嵌套子查询；掌握多表的连接查询，包括左连接、右连接、外连接、自连接。

7. 掌握MySQL中存储过程、函数、触发器的定义和使用。

8. 安装和使用phpmyadmin。

## 二、实验内容

1. 在Windows系统中安装VMWare虚拟机，在VMWare中安装Ubuntu系统，在Ubuntu中搭建LAMP实验环境。

2. 使用MySQL进行一些基本操作：

（1）登录MySQL，在MySQL中创建用户，并对新建的用户赋予权限。

（2）创建并跳转到新的数据库，显示所有数据库和当前数据库。

（3）显示所有的表和查看表的属性。

（4）导入employees和sakila两个样例数据库，对这两个数据库进行完整性检查，对这两个数据库进行备份、导入与导出。

3. SQL DDL使用

（1） 创建DDL脚本，包括创建、删除表，指定主键、候选键和外键；

（2）修改并展示表结构

（3）创建、删除索引。

4. SQL DDL & DML使用

（1）数据的增、删、查、改等操作

（2）掌握视图的创建、删除和更新

5. 简单查询

（1）使用order by、group by、having等子句；

（2）使用各种谓词；

（3）使用集合函数；

（4）在时间字段上查询。

6. 复杂查询

（1）嵌套子查询；

（2）多表连接查询，包括左连接、右连接、外连接、自连接。

7. 创建存储过程、函数和触发器，调用创建的存储过程、函数，触动触发器。

8. 安装并使用phpmyadmin。

## 三、实验步骤

1. 请自行在 Windows系统中安装VMWare虚拟机，在VMWare中安装Ubuntu系统，在Ubuntu中安装LAMP步骤如下（也可使用安装菜单）：

sudo apt-get install update

sudo apt-get install apache2

sudo apt-get install mysql-server

sudo apt-get install php

2. 熟悉MySQL的基本操作步骤如下：

（1）进入MySQL：mysql -u root -p

（2）创建新用户：CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'password';

（3）赋予权限：GRANT ALL PRIVILEGES ON \* . \* TO 'newuser'@'localhost';

（4）创建新的数据库：CREATE DATABASE newdbname;

（5）跳转到新创建的数据库：USE newdbname;

（6）显示所有数据库和当前数据库：SHOW DATABASES;

SELECT DATABASE();

（7）显示所有的表并查看表的属性：SHOW TABLES;

DESCRIBE tabname;

（8）导入两个样例数据库

① 解压数据包：unzip test\_db\_master.zip

② 进入数据包目录：cd test\_db\_master

③ 导入employee数据库：sudo sudo mysql -t < employees.sql

④ 解压数据包：unzip sakila-db.zip

⑤ 进入数据包目录：cd sakila-db

⑥ 进入MySQL，导入sakila schema数据：SOURCE /path/to/sakila-schema.sql

⑦ 导入sakila数据：SOURCE /path/to/sakila-data.sql

（9）对导入的两个数据库进行完整性检查

① 查看外键检查状态：show variables like ‘%foreign\_key\_checks%’;

② 查看主键检查状态：show variables like ‘%unique\_checks%’;

③ 查看外键检查结果：select @@foreign\_key\_checks;

④ 查看主键检查结果：select @@unique\_checks;

（10）数据库备份（以employee为例）

① 不导出任何数据，只导出数据库表结构：

mysqldump -u test\_02 -p --no-data employees> employees\_bak1.sql

② 只导出数据，而不添加CREATE TABLE 语句：

mysqldump -u test\_02 -p --no-create-info employees> employees\_bak2.sql

③ 导出全部数据库：

mysqldump -u test\_02 -p employees> employees\_bak3.sql

（11）导出和导入（以sakila为例）

① 导出为TXT文件：

select \* from country into outfile '/your/file/path/country';

② 导入TXT文件：

load data infile '/your/file/path/country' into table country;

③ 导出为CSV文件：

select \* from country into outfile '/your/file/path/country.csv';

④ 导入CSV文件：

load data infile '/your/file/path/country.csv' into table country;

⑤ 导出为xml文件：

mysql -u test\_02 -p --xml -e 'select \* from sakila.country' > /your/file/path/country.xml

3. SQL DDL使用

（1）使用create语句创建项目所有的表；

（2）使用alter命令修改表结构，删除某个表中的“xxx”字段；修改某个表中的“xxx”字段的类型为char(2)，该字段不能为空，默认值为“m”；

（3）添加类型为char（2）的‘xxx’字段数据，添加是否成功，如果失败分析失败的原因并进行必要的操作使字段添加成功；

（4）修改某个表，添加类型为varchar2(18)的字段，并添加check约束，要求该字段的长度为18，并且只能由数字组成，并且指定该字段为候选键；

（5）设置某个表中某个字段的check约束为大于0；设置某个字段默认值为“未审核”，设置某个”字段的check约束为“未审核”、“审核已通过”、“审核不通过”；

（6）在某个表中创建索引；

（7）使用describe命令展示表结构。

4. SQL DDL & DML使用

1. 使用insert语句将数据插入到相应的表中；
2. 使用delete语句删除表中有关联表外建对应的记录，能否成功删除，如果不能请分析原因；
3. 使用update语句更新外键数据，能否成功修改，如果不能请分析原因。；
4. 创建视图；
5. 对视图进行查询操作；
6. 对表进行联合查询操作；
7. 更新视图，分析更新操作可以执行成功或失败的原因。

5. 简单查询

1. 查询性别为“男”的所有学生的学号、姓名和班级号；（单表简单查询）
2. 查询xxx表，获得性别为“女”的记录，结果按照班级ID降序排列；（order by）
3. 查询xxx表，按照年龄从小到大排序；（获取子串函数、order by）
4. 查询学号以“2002”开头的学生信息，字段包括学号、姓名、班级号；（使用子串函数）
5. 查询学号中包含“01”的学生信息，字段包括学号、姓名、班级号；（like）
6. 查询状态为“未审核”，且申请时间在2013年9月4日之后的请假申请单的信息，包含申请时间为9月4日的申请单；（单表多条件查询）
7. 查询审核状态为“未审核”和“审批已通过”两种类型的申请单ID；（in）
8. 查询时间在2013年8月31日和2013年9月2日之间所提交的申请单ID，请假原因；(between、时间)
9. 查询XXX老师所教课程的的选课人次（一人选两门课程，算两人次）；
10. 查询XXX老师所教课程的的选课人数（一人选两门课程，算一人）；（distinct）
11. 查询XX老师对请假单审核不通过的请假原因与学生姓名；
12. 统计每门课的学生的个数（集合函数）；
13. 查询选课人数超过3人的课程号，并按课程号降序排列（group by, having, order by …desc）。

6. 复杂查询

（1）查询审批通过人数最多的课程名称和教师姓名；

（2）查询选课人数最多和第二多的课程名称和任课老师姓名；(选做)

（3）查询只选了课程《算法设计》的学生姓名；

（4）查询选修了全部课程的学生姓名；

（5）查询选修了课程3的学生学号、姓名、身份证号；

（6）据学生学号将xxx表与xxx表做自然连接查询；

（7）据学号将xxx表与xxx表进行左连接查询，并解释这样做所具有的业务含义；

（8）据学号将xxx表与xxx表进行右连接查询，并解释这样做所具有的业务含义；

（9）总结自然连接、左连接、右连接查询在产生的结果上面有什么区别。

//7. 创建并调用存储过程、函数和触发器

（1）创建并调用存储查询过程（以employee为例）

① 创建一个查询存储过程：

> delimiter##

> create procedure select\_manager(in fname varchar(20), in lname varchar(20))

-> begin

->select \* from dept\_manager natural join employees where first\_name = fname and last\_name = lname;

->end

> ##

② 调用这个存储查询过程：call select\_manager('Xiaobin','Spinelli');

（2）创建并调用函数（以employee为例）

① 创建函数：

> delimiter ##

> create function title\_num(title\_name varchar(50)) returns int

> begin

> declare num int;

> select count(title) from titles where to\_date > curdate() and title = title\_name group by title into num;

> return num;

> end

> ##

set global log\_bin\_trust\_function\_creators=1;

② 调用函数：select title\_num("manager");

（3）创建并调用触发器（以employee为例）

① 查看表dept\_manager：select \* from dept\_manager;

② 创建新表quit：create table quit (emp\_no int, depr\_no char(4), quit\_date date);

③ 创建触发器：

mysql> delimiter ##

mysql> create trigger after\_ insert\_ quit after insert on quit for each row

-> begin

-> update dept\_ manager set to\_ date = curdate( ) where emp\_ no = 110039;

-> end

-> ##

④ 触发触发器：insert into quit values (110039, 'd001', curdate());

⑤ 查看新表quit：select \* from quit;

⑥ 再次查看表dept\_manager：select \* from dept\_manager;

8. 安装phpmyadmin步骤如下：

（1）安装phpmyadmin：sudo apt-get install phpmyadmin

（2）安装php-mbstring：sudo apt-get install php-mbstring

（3）安装php-gettext：sudo apt-get install php-gettext

（4）编辑php.ini: vim /path/to/php.ini 修改；extension=php-mbstring.so

（5）重启Apache2：sudo /path/to/apache2 restart

（6）查看IP地址：ifconfig

（7）登录phpmyadmin：从浏览器地址 your.ip.4.address/phpmyadmin