**每日作业卷**

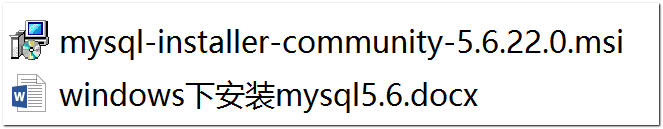
**javaweb第19天mysql**

# 关卡1

## 训练案例1

### 训练描述

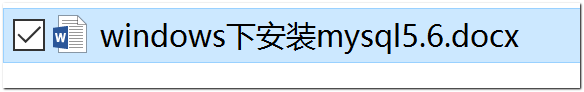
能够独立完成mysql数据库的安装,启动mysql数据库服务和停止数据库服务



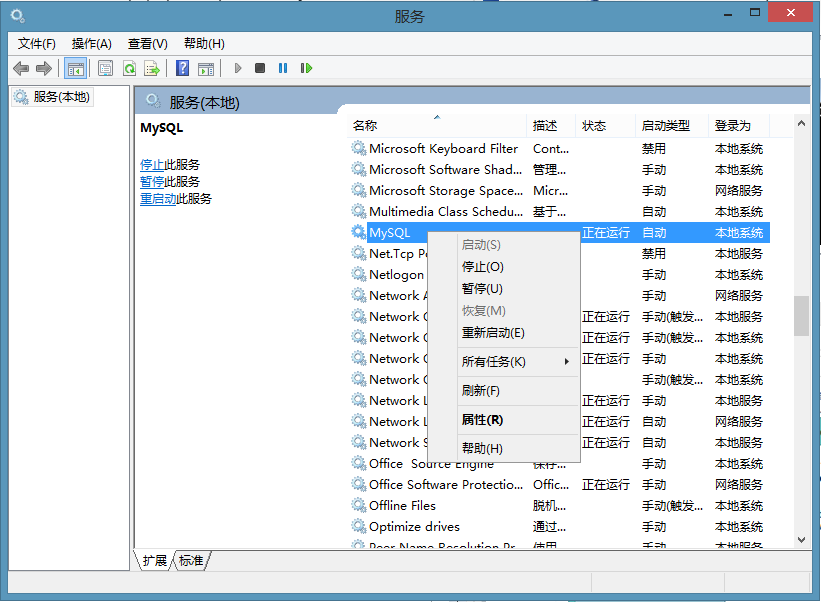
### 操作步骤答案

1. 下载好Mysql数据库应用程序windows版本5.6.22.0.msi. 双击安装

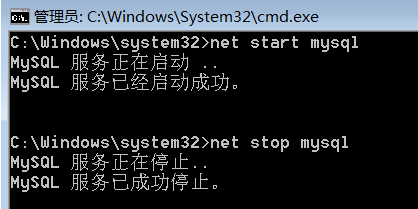
安装步骤参照下发的文档



1. 安装后，MySQL会以windows服务的方式为我们提供数据存储功能。开启和关闭服务的操作：右键点击我的电脑→管理→服务→可以找到MySQL服务开启或停止。



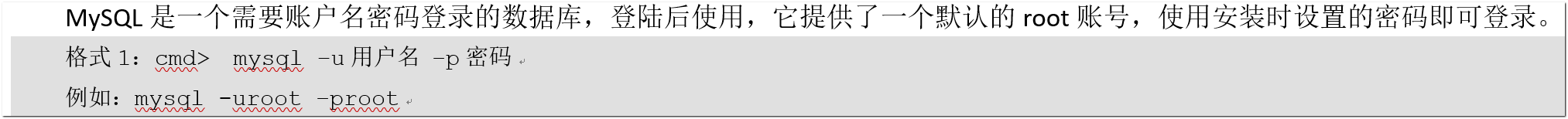
1. 也可以在DOS窗口，通过命令完成MySQL服务的启动和停止（必须以管理员身份运行cmd命令窗口）



## 训练案例2

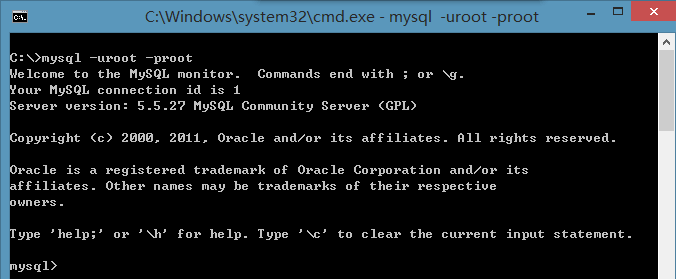
### 训练描述

登录mysql数据库,使用root登录,采用cmd窗口命令登录

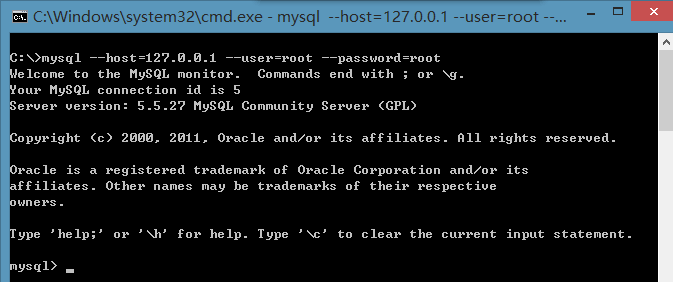


### 操作步骤答案

1. 打开cmd窗口,使用mysql命令,使用root用户登录 mysql -uroot -proot



1. 使用root用户,以ip地址形式登录数据库 mysql --host=127.0.0.1 --user=root --password=root



## 训练案例3

### 训练描述

MySQL图形化开发工具安装,使用工具完成mysql数据库的连接

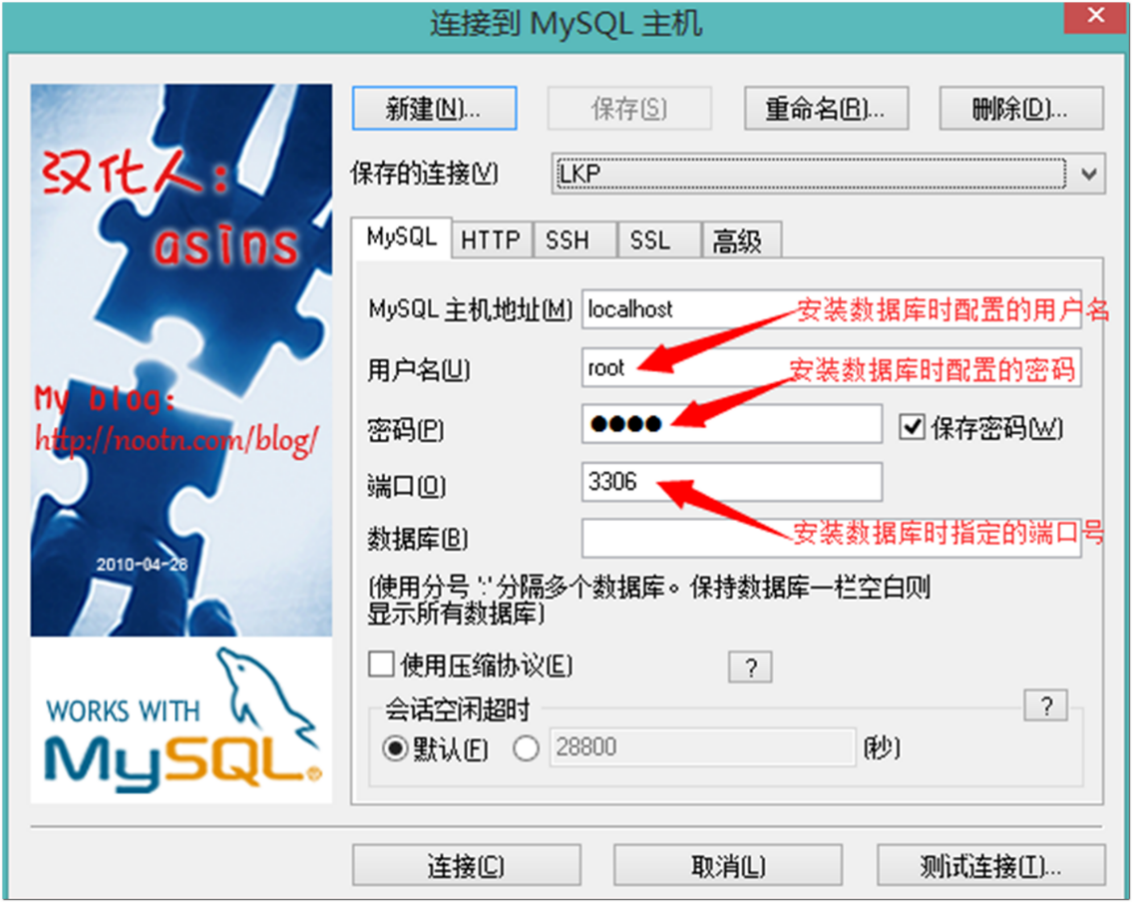


### 操作步骤答案

1. 安装SQLyog(免费版) windows版本 双击安装即可.

下载SQLyog直接安装即可.

1. 使用SQLyog连接mysql数据库配置. 具体如下:



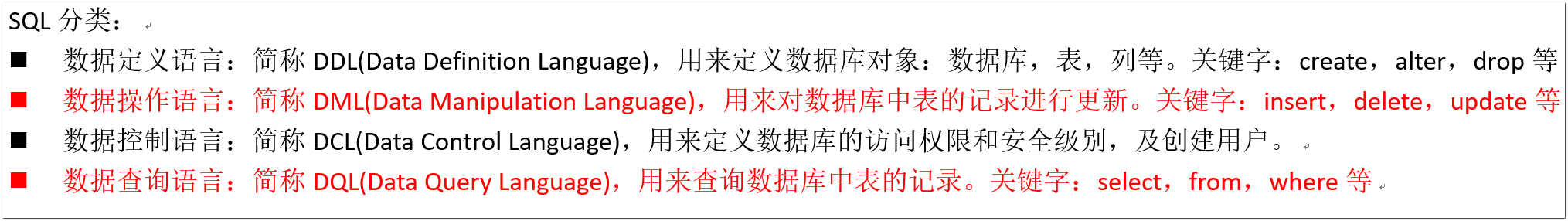
## 训练案例4

### 训练描述

SQL语句分类 DDL, DML ,DQL, DCL表示的含义

### 操作步骤答案

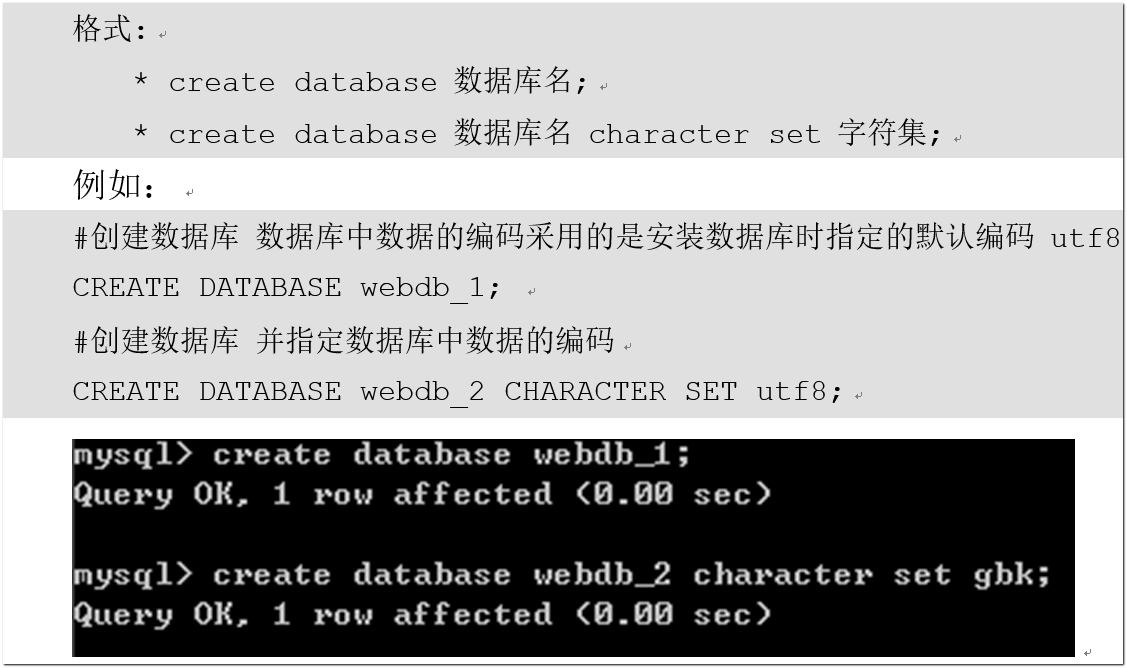
1.说出DDL,DML,DQL,DCL分别表示的语句种类.



## 训练案例5

### 训练描述

创建数据库并设置utf8编码,完成数据库的添加,删除,查询,使用.



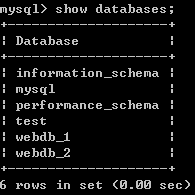
### 操作步骤答案

1. 使用root用户登录数据库,创建数据库并设置编码utf8.

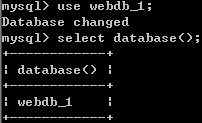
create database webdb\_1 character set utf8;

2.使用root用户登录数据库,查询数据库,删除数据库,使用数据库。

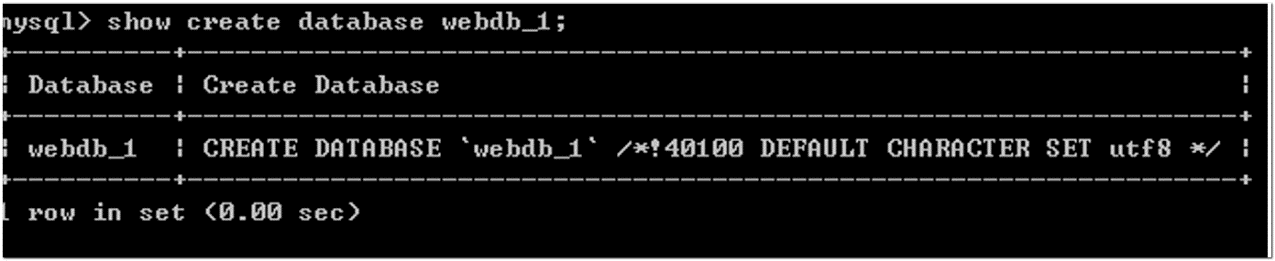
查询root下的所有数据库信息 : show databases;



使用数据库 use webdb\_1;

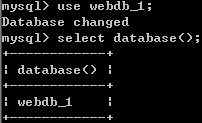


查看某个数据库的定义的信息: show create database webdb\_1;



删除数据库 drop database webdb\_1;

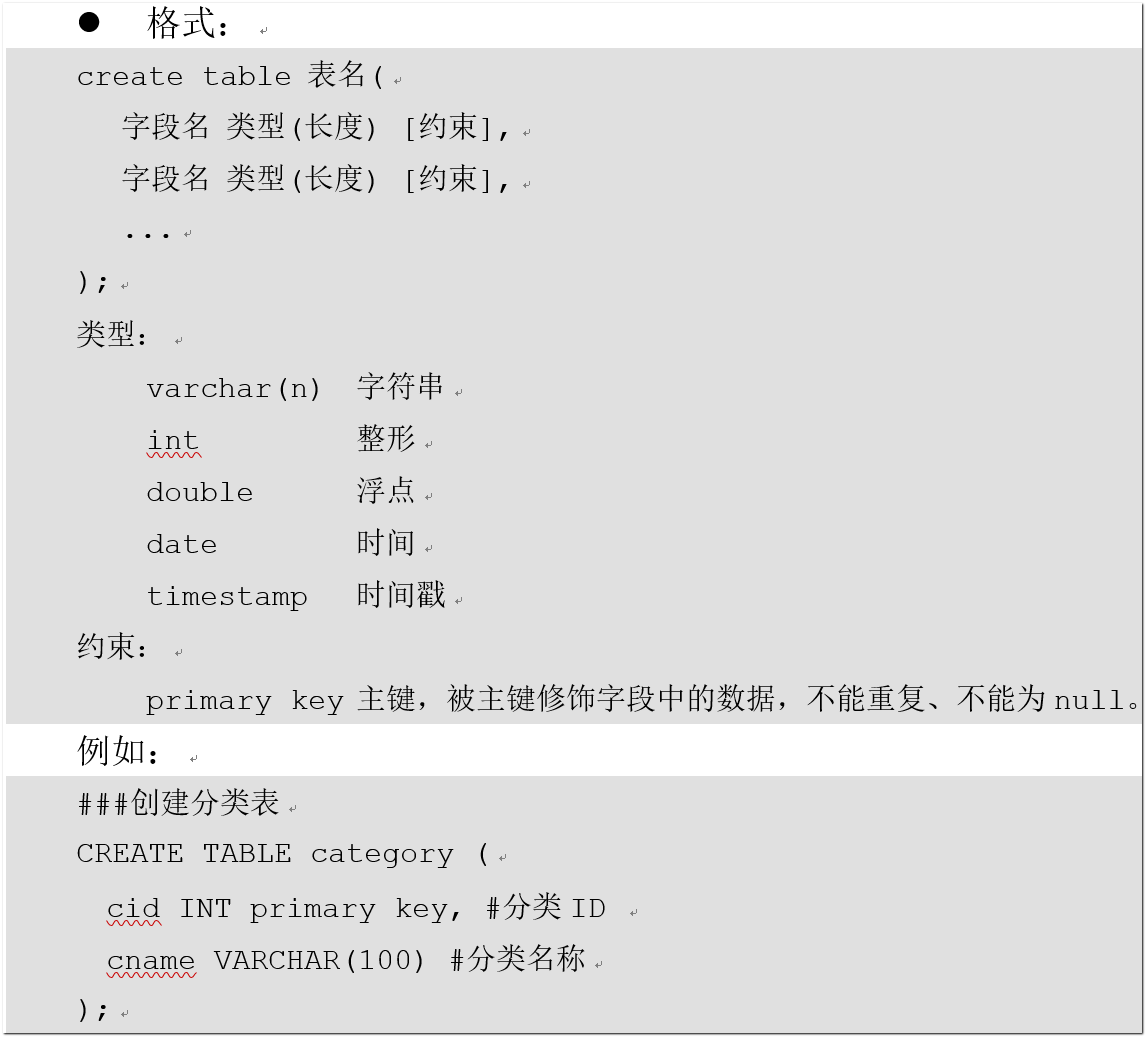
查询当前使用的数据库 select database();



## 训练案例6

### 训练描述

使用数据库webdb\_1,在该数据库下创建表: category.



### 操作步骤答案

1. 切换webdb\_1数据库

use webdb\_1;

1. 创建表 category 字段cid varchar(32)类型 主键 , cname 字符类型 varchar(100)

create table if not exists category(cid varchar(32) primary key,cname varchar(100));

## 训练案例7

### 训练描述

使用数据库webdb\_1,在该数据库针对表: category. 表结构查询,修改表结构,删除表.









### 操作步骤答案

1. 使用命令desc查询表结构

desc categroy;

1. 使用命令drop删除表

drop table category;

1. 使用命令alert完成表结构修改.



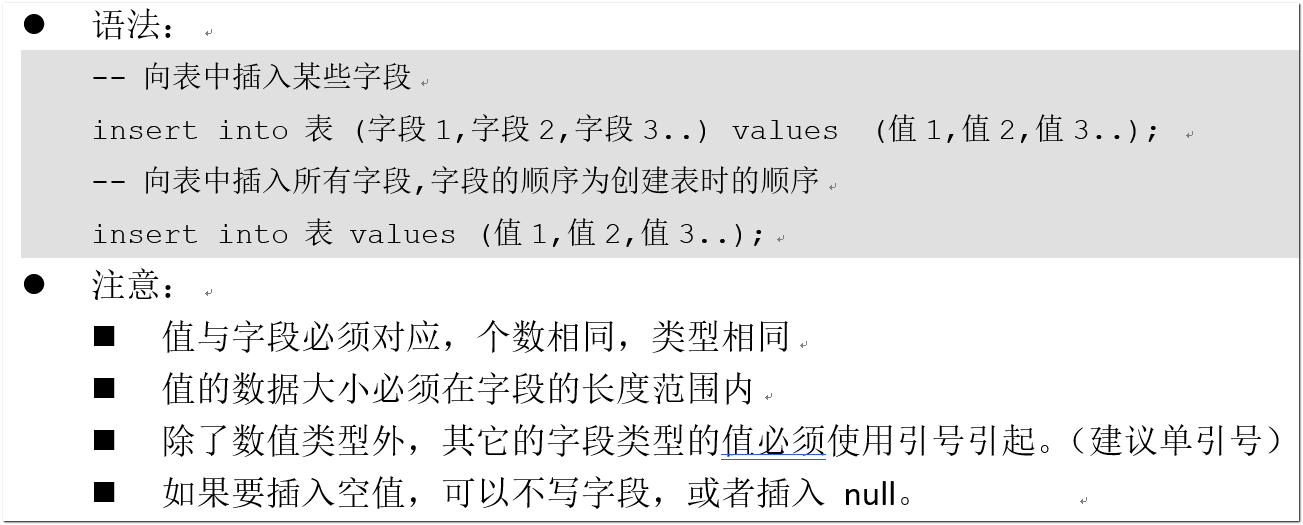


## 训练案例8

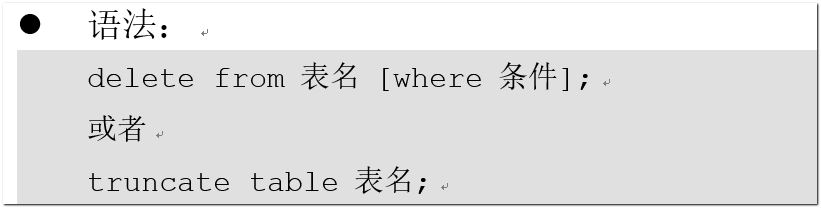
### 训练描述

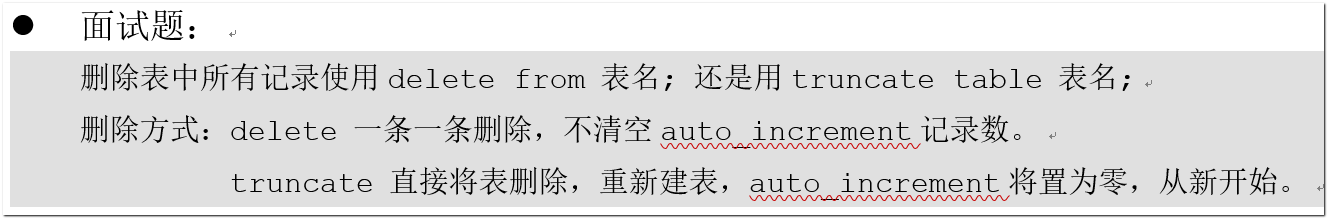
DML之insert update delete表记录操作使用 完成表记录的添加,修改,删除操作.

insert使用语法

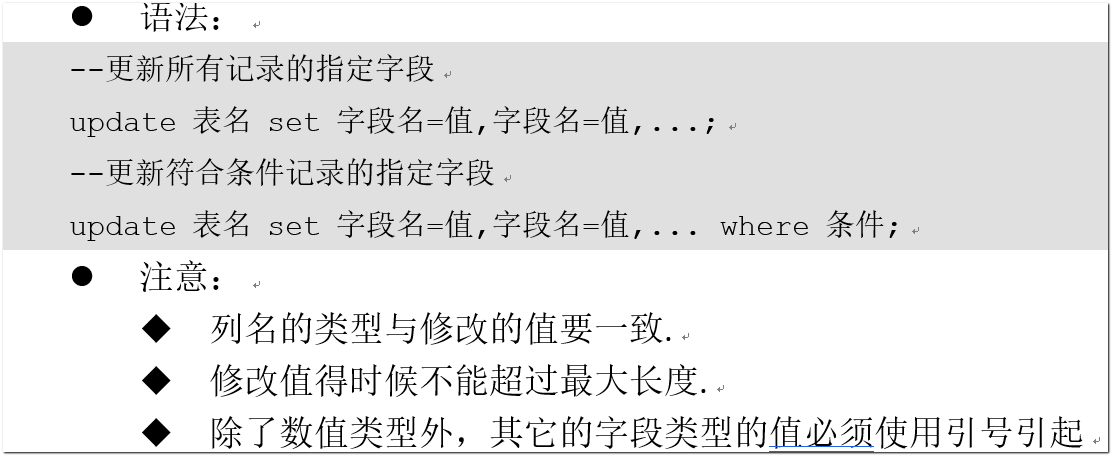


delete使用语法





update使用语法



### 操作步骤答案

1. 使用insert命令完成表category数据的插入

INSERT INTO category(cid,cname) VALUES('c001','电器');

INSERT INTO category(cid,cname) VALUES('c002','服饰');

INSERT INTO category(cid,cname) VALUES('c003','化妆品');

INSERT INTO category(cid,cname) VALUES('c004','书籍');

INSERT INTO category(cid) VALUES('c005');

INSERT INTO category(cname,cid) VALUES('耗材','c006');

1. 使用delete命令删除category一条记录.

delete from category where cid =’c001’;

1. 使用update命令修改category表中某些记录的字段值

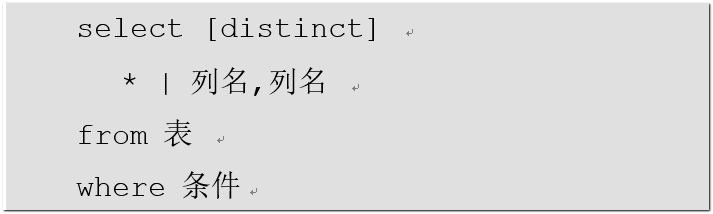
update category set cname=’化妆’ where cid = ‘c005’;

## 训练案例9

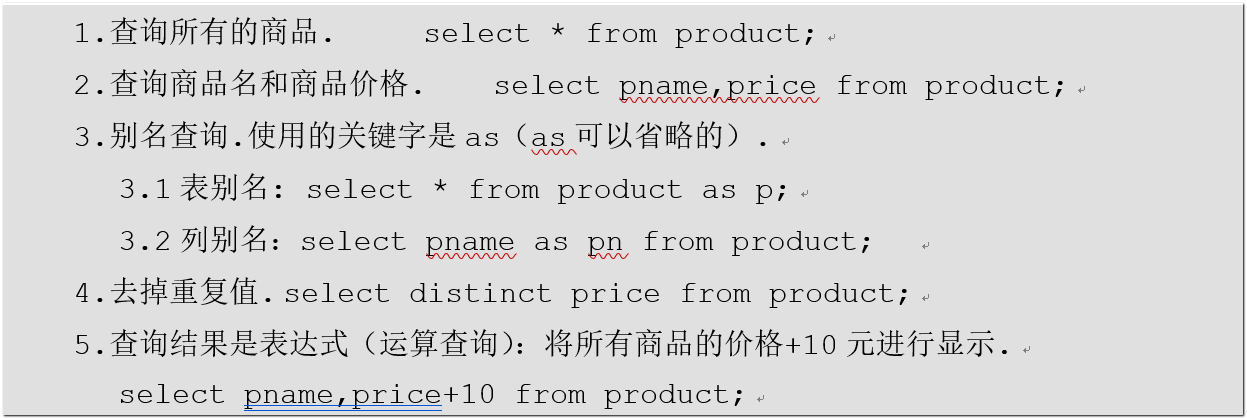
### 训练描述

DQL查询语句使用,创建一张新表product 录入初始化数据,完成商品的查询以及条件查询.

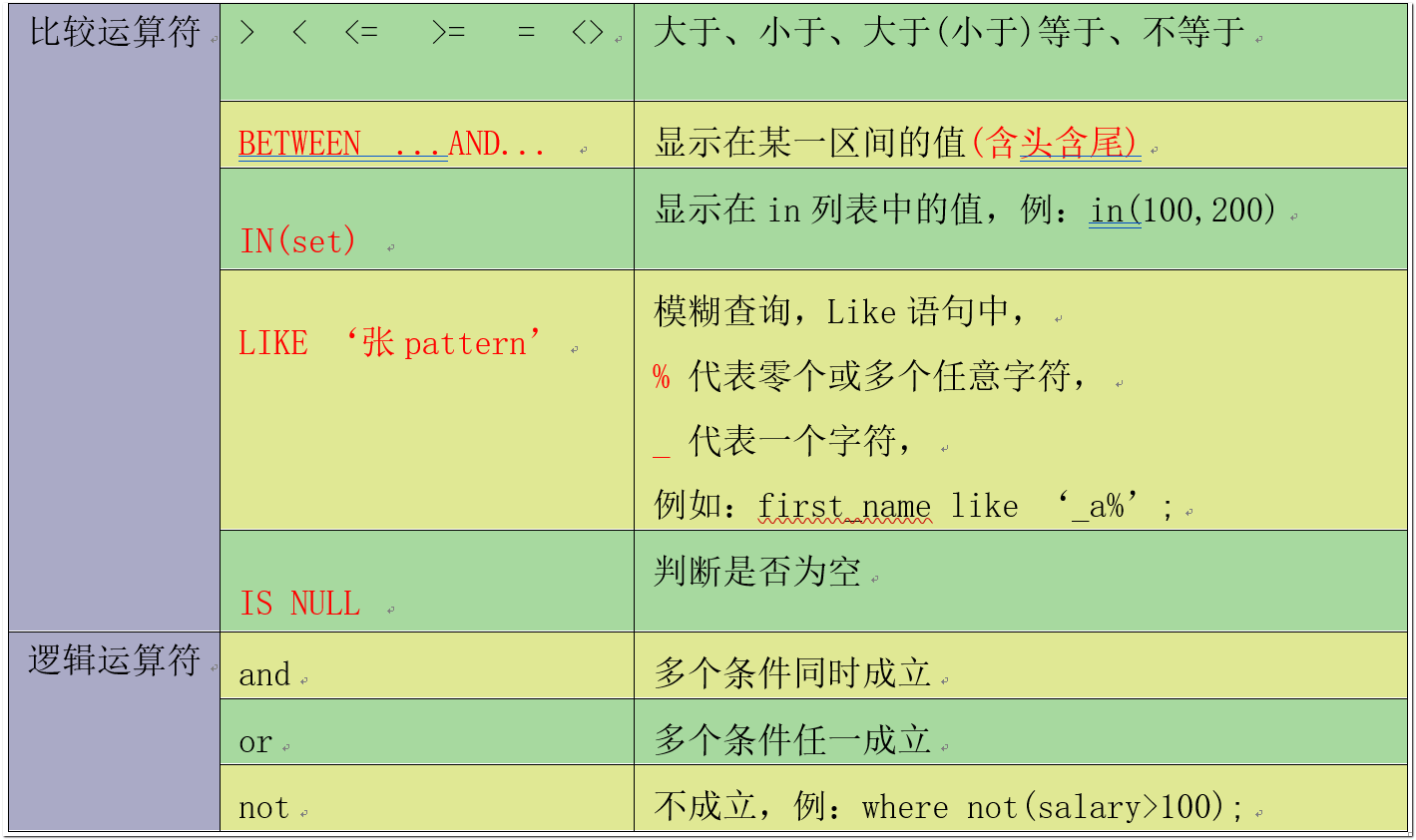
基本查询语法



简单查询说明



条件查询说明



准备工作如下:

创建表product以及准备数据sql脚本.

#创建商品表：

create table product(

pid int primary key,

pname varchar(20),

price double,

category\_id varchar(32)

);

INSERT INTO product(pid,pname,price,category\_id) VALUES(1,'联想',5000,'c001');

INSERT INTO product(pid,pname,price,category\_id) VALUES(2,'海尔',3000,'c001');

INSERT INTO product(pid,pname,price,category\_id) VALUES(3,'雷神',5000,'c001');

INSERT INTO product(pid,pname,price,category\_id) VALUES(4,'JACK JONES',800,'c002');

INSERT INTO product(pid,pname,price,category\_id) VALUES(5,'真维斯',200,'c002');

INSERT INTO product(pid,pname,price,category\_id) VALUES(6,'花花公子',440,'c002');

INSERT INTO product(pid,pname,price,category\_id) VALUES(7,'劲霸',2000,'c002');

INSERT INTO product(pid,pname,price,category\_id) VALUES(8,'香奈儿',800,'c003');

INSERT INTO product(pid,pname,price,category\_id) VALUES(9,'相宜本草',200,'c003');

INSERT INTO product(pid,pname,price,category\_id) VALUES(10,'面霸',5,'c003');

INSERT INTO product(pid,pname,price,category\_id) VALUES(11,'好想你枣',56,'c004');

INSERT INTO product(pid,pname,price,category\_id) VALUES(12,'香飘飘奶茶',1,'c005');

INSERT INTO product(pid,pname,price,category\_id) VALUES(13,'果9',1,NULL);

### 操作步骤答案

#查询商品名称为“花花公子”的商品所有信息：

SELECT \* FROM product WHERE pname = '花花公子'

#查询价格为800商品

SELECT \* FROM product WHERE price = 800

#查询价格不是800的所有商品

SELECT \* FROM product WHERE price != 800

SELECT \* FROM product WHERE price <> 800

SELECT \* FROM product WHERE NOT(price = 800)

#查询商品价格大于60元的所有商品信息

SELECT \* FROM product WHERE price > 60;

#查询商品价格在200到1000之间所有商品

SELECT \* FROM product WHERE price >= 200 AND price <=1000;

SELECT \* FROM product WHERE price BETWEEN 200 AND 1000;

#查询商品价格是200或800的所有商品

SELECT \* FROM product WHERE price = 200 OR price = 800;

SELECT \* FROM product WHERE price IN (200,800);

#查询含有'霸'字的所有商品

SELECT \* FROM product WHERE pname LIKE '%霸%';

#查询以'香'开头的所有商品

SELECT \* FROM product WHERE pname LIKE '香%';

#查询第二个字为'想'的所有商品

SELECT \* FROM product WHERE pname LIKE '\_想%';

#商品没有分类的商品

SELECT \* FROM product WHERE category\_id IS NULL

#查询有分类的商品

SELECT \* FROM product WHERE category\_id IS NOT NULL

## 训练案例10

### 训练描述

解决DOS窗口操作数据中文乱码.

我们在dos命令行操作中文时，会报错

insert into category(cid,cname) values(‘c010’,’中文’);

ERROR 1366 (HY000): Incorrect string value: '\xB7\xFE\xD7\xB0' for column 'cname' at row 1

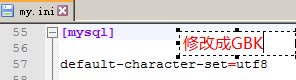
### 操作步骤答案

1: 中文乱码错误原因:因为mysql的客户端设置编码是utf8,而系统的cmd窗口编码是gbk

解决方案1：在cmd命令窗口中输入命令，此操作当前窗口有效，为临时方案。

set names gbk;

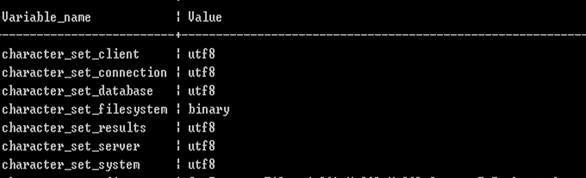
解决方案2：mysql数据库的安装目录下修改my.ini文件，将安装文件中的utf8修改为gbk,需要重启服务所有地方生效。[不建议]



解决方案3: 使用命令单独设置编码为gbk

3.1查看MySQL内部设置的编码

show variables like 'character%'; 查看所有mysql的编码



3.2使用命令修改client、connection、results的编码一致（GBK编码）client connetion result 和客户端保持一致，都为GBK.

#方式1：单独设置

set character\_set\_client=gbk;

set character\_set\_connection=gbk;

set character\_set\_results=gbk;

#方式2：快捷设置

set names gbk;

# 关卡2

## 训练案例1

### 训练描述

操作题1:

在cmd窗口,使用mysql命令,创建编码utf8的数据库webdb2,创建表users,完成相关查询

要求: users表的字段 id name age birthday salary

1:字段要求: id int 主键 , age int , name 字符类型, birthday 日期类型 salary 数值类型 要求小数点保留2位.

2:录入初始化数据,薪资可以录入,也可以不录入数据.

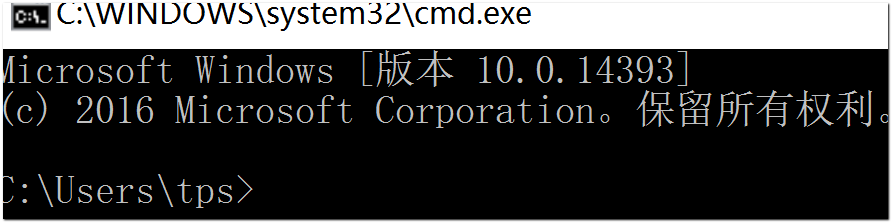
3:查询所有员工的薪资,年薪,以及姓名,采用列别名形式显示字段

4:查询所有员工,如果薪资没有的员工,salary列值显示:没有薪资,倒序显示.

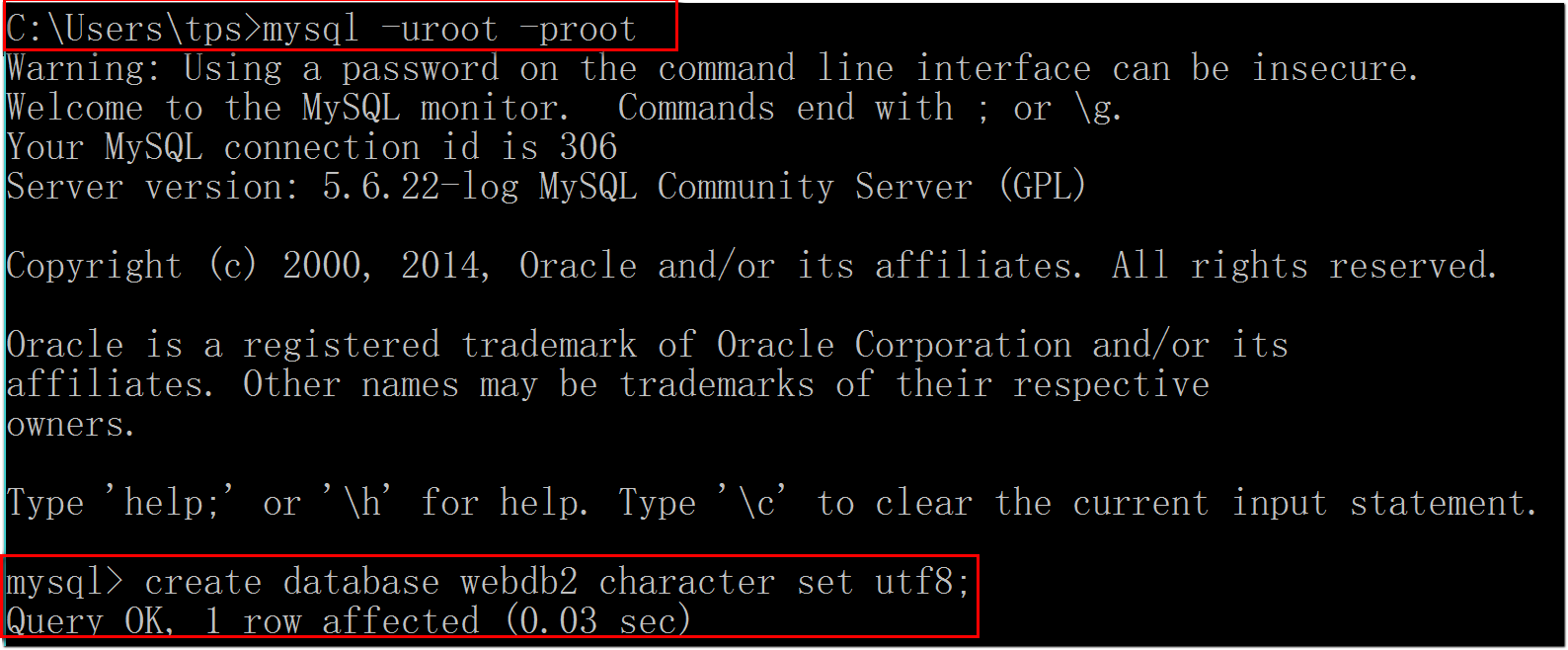
5:将salary列为null的用户薪资,修改为0.00.

### 操作步骤答案

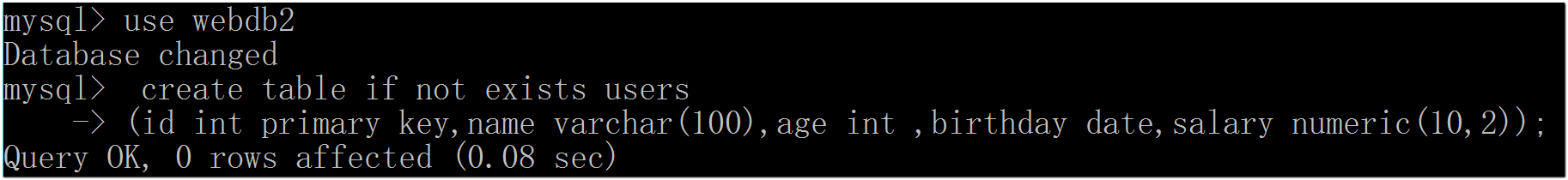
1:在windows运行窗口 输入 cmd 打开dos窗口



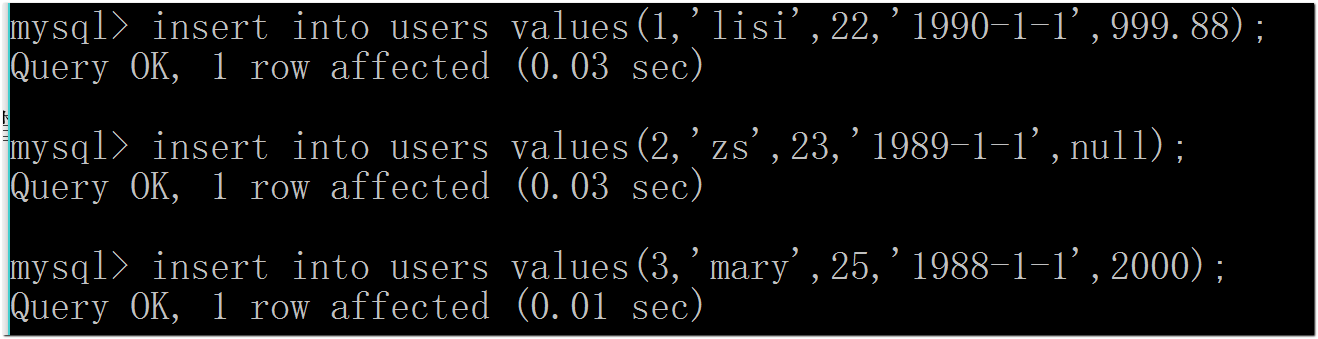
2:使用mysql命令 采用root用户登录 创建数据库webdb2 编码为utf8



3:使用webdb2数据库,创建users表 设计字段.

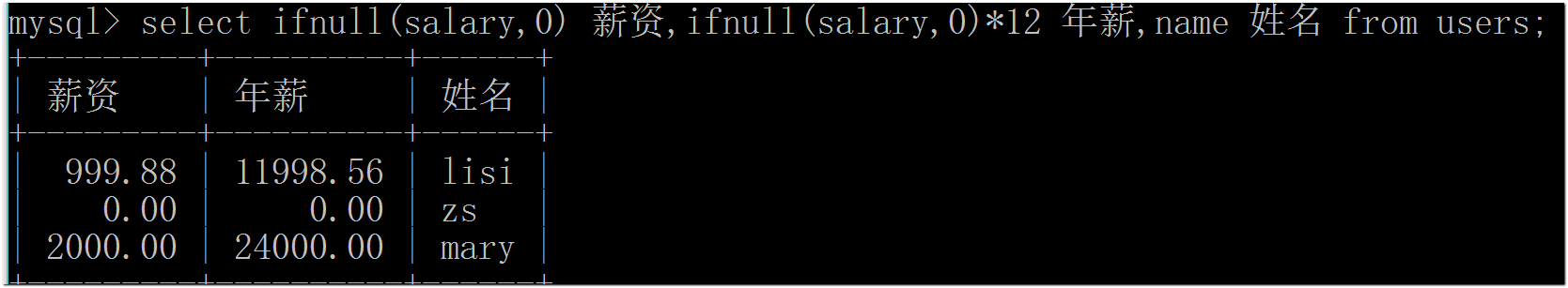


4:随机插入一些数据,注意薪资一列交替的插入数据或不插入数据

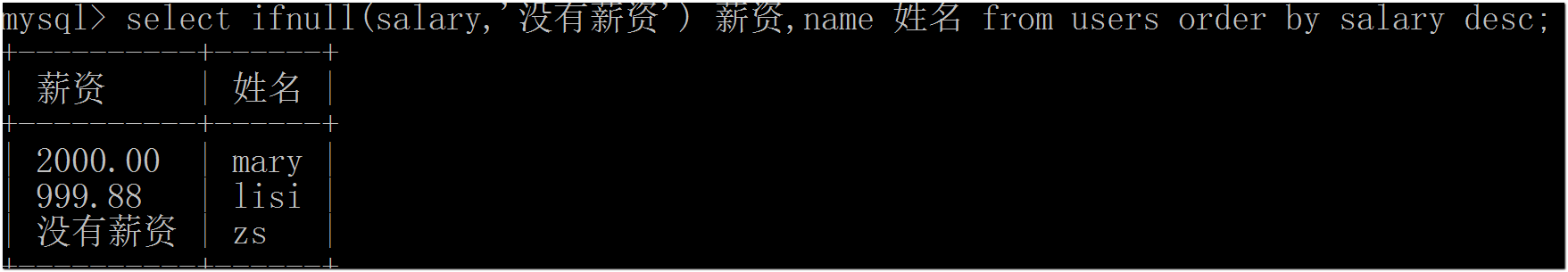


5:完成相关查询要求 使用关键字 select ,from ,where,ifnull完成数据查询.

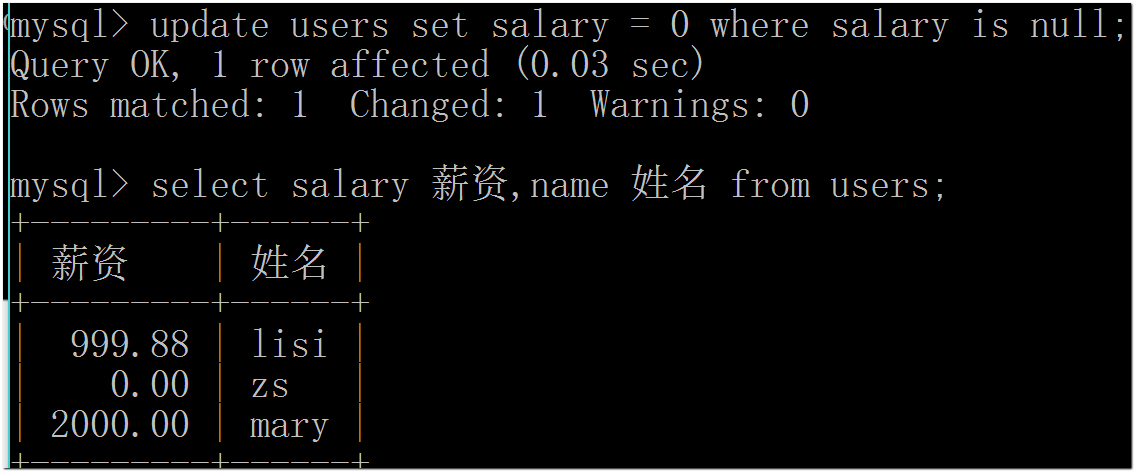
5.1:查询所有员工的薪资,年薪,以及姓名



5.2:查询所有员工,如果薪资没有的员工,salary列值显示:没有薪资,倒序显示.



5.3:将salary列为null的用户薪资,修改为0.00.



## 训练案例2

### 训练描述

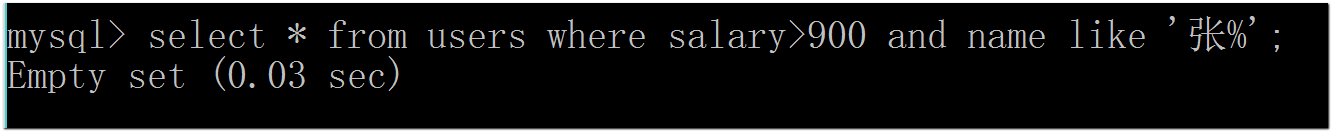
操作题2:

基于用户表users完成相关查询.

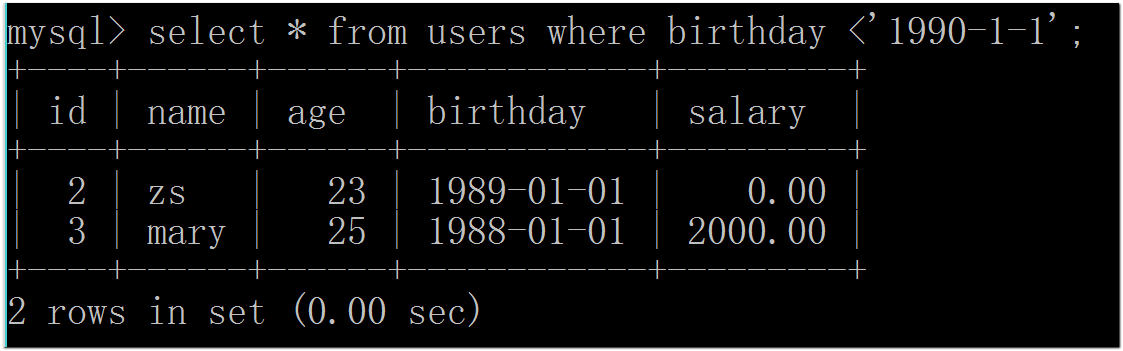
### 操作步骤答案

使用select,from ,where ,or ,and,> ,< ,like, % 完成查询.

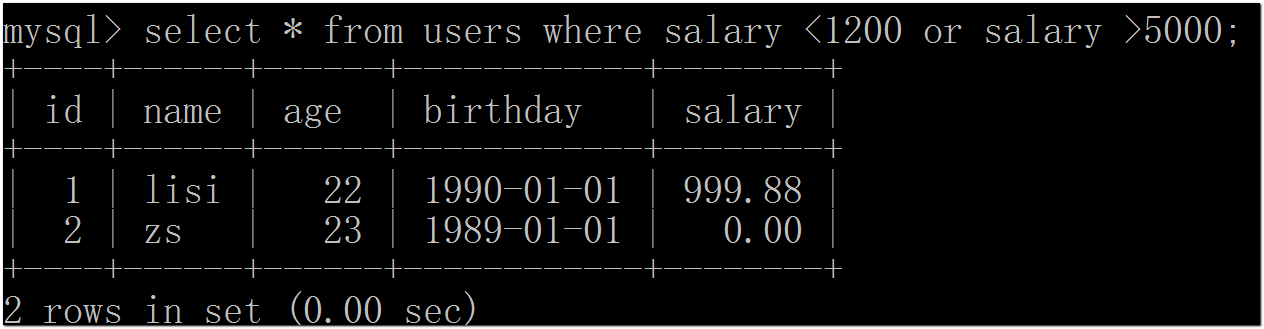
1:查询员工张姓且薪资大于900的员工.



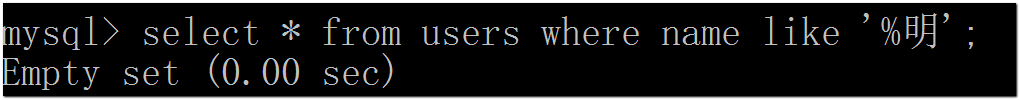
2:查询生日在1990-1-1之前的员工信息.



3:查询员工薪资小于1200 或者 大于5000的信息



4:查询姓名以明结尾的员工信息.



# 关卡3

## 训练案例1

### 训练描述

1: cmd窗口,采用root登录,创建一个新客户 rabbit,密码 123.

2:使用root创建一个全新的数据库webdb3,编码utf8.

3:给rabbit授予webdb3所有权限.

4:root用户退出,采用rabbit登录.

5:rabbit用户登录,查询所有数据库信息.

6:在webdb3下创建一张新表demo表

7:设计demo表的基本字段 id int 主键, name 姓名字符类型.

8:录入初始化数据,可以随机录入一些数据.(推荐录入3条记录以上)

9:将webdb3数据库数据进行备份.

10:将webdb3数据库进行删除.

11:对webdb3数据库数据进行恢复.

* **备份多个数据库**

　　语法：mysqldump -u username -p --databases dbname2 dbname2 > Backup.sql

　　加上了--databases选项，然后后面跟多个数据库

例如: mysqldump -u root -p --databases test mysql > D:\backup.sql

* **备份所有数据库**

　　mysqldump命令备份所有数据库的语法如下：

mysqldump -u username -p -all-databases > BackupName.sql

　　示例：

mysqldump -u -root -p -all-databases > D:\all.sql

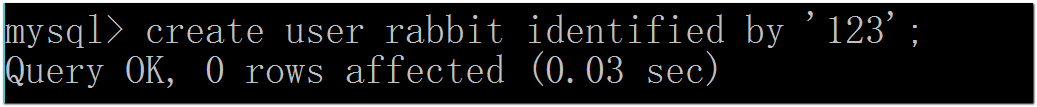
* 恢复数据库:使用mysqldump命令备份的数据库进行恢复：

　　mysql -u root -p [dbname] < backup.sql

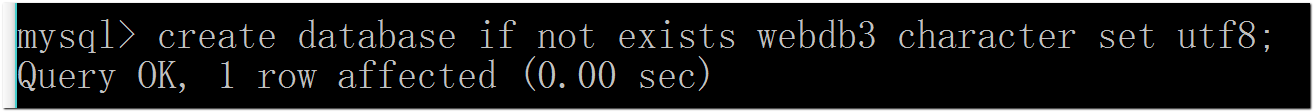
　　示例：mysql -u root -p < D:\backup.sql

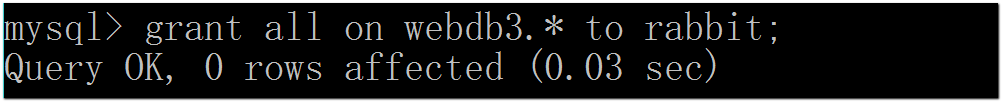
### 操作步骤答案

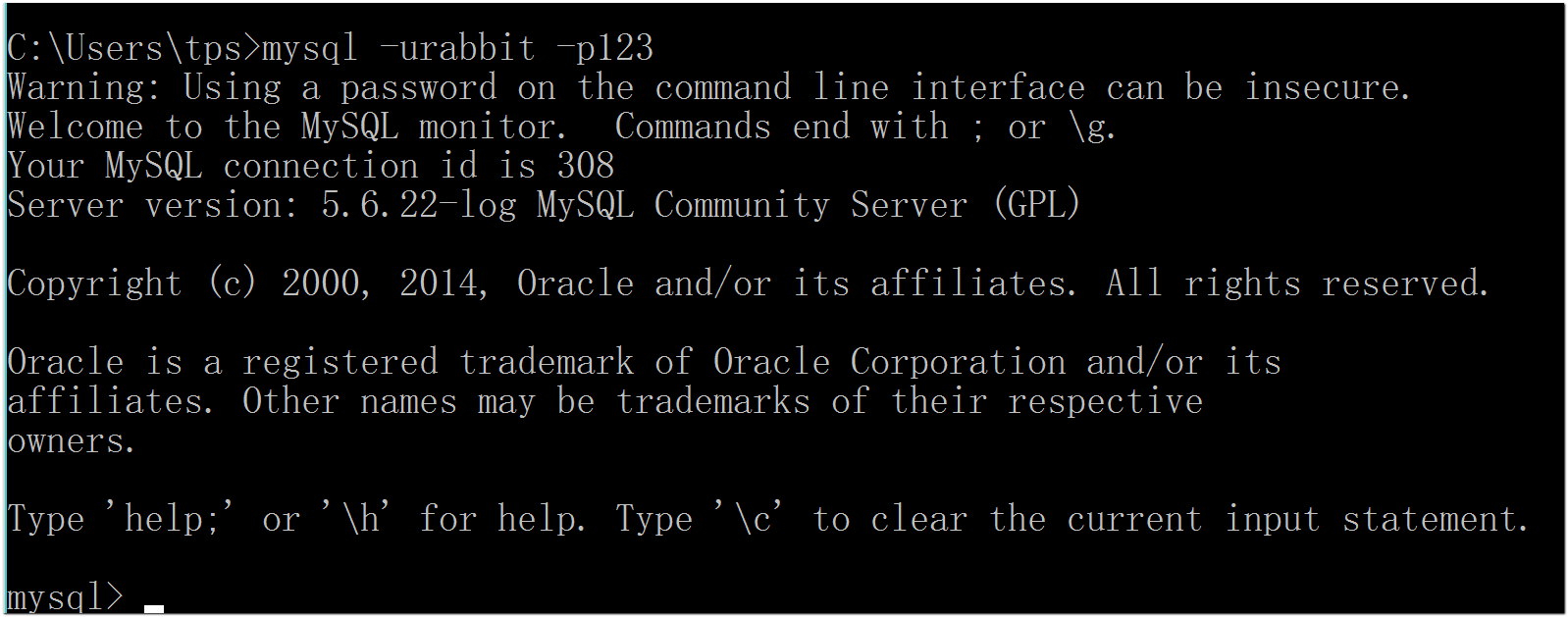
1:使用root用户登录,创建新用户rabbit 密码 123

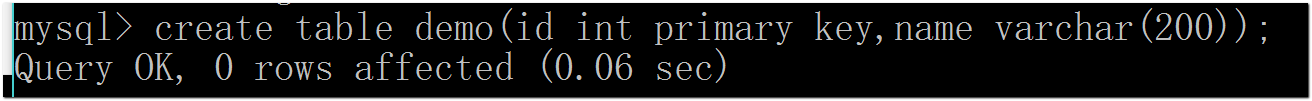


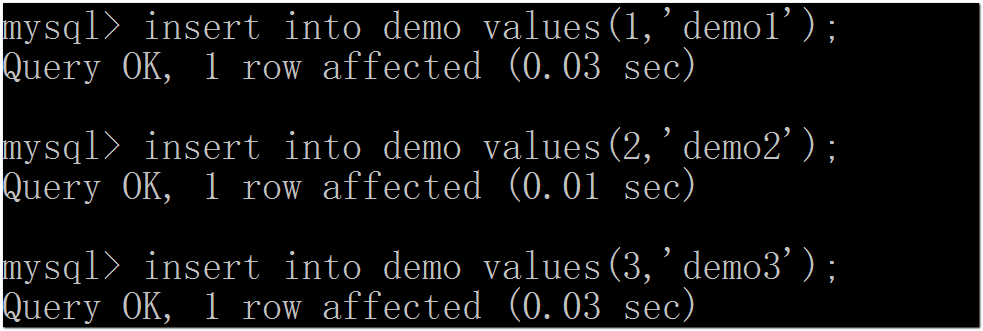
2:创建一个细腻的数据库webdb3,对rabbit用户授予该数据库的所有权限



3:使用rabbit登录 在webdb3下创建一张表(随机一张即可) 录入相关数据

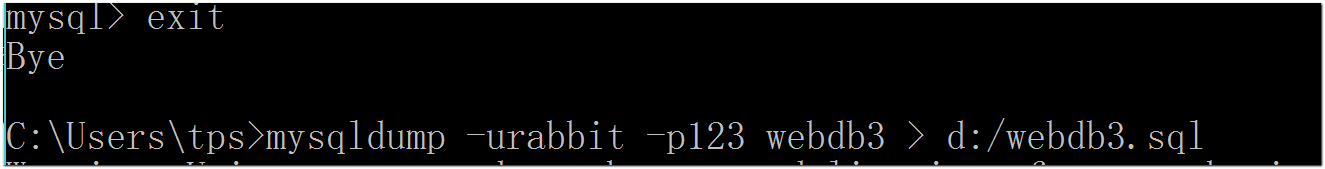


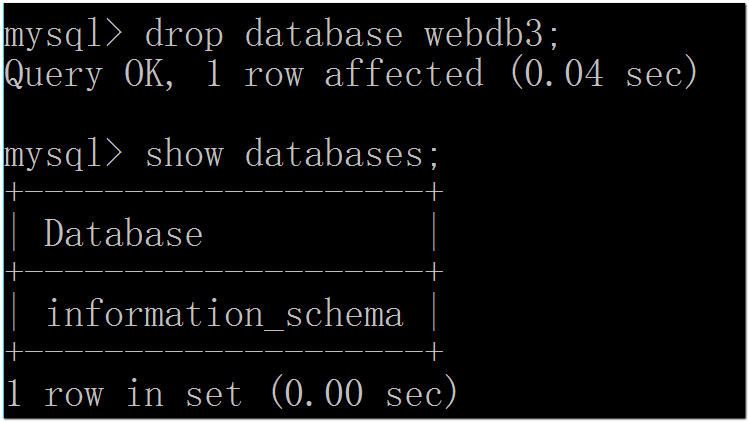




4:对webdb3进行备份和恢复操作.

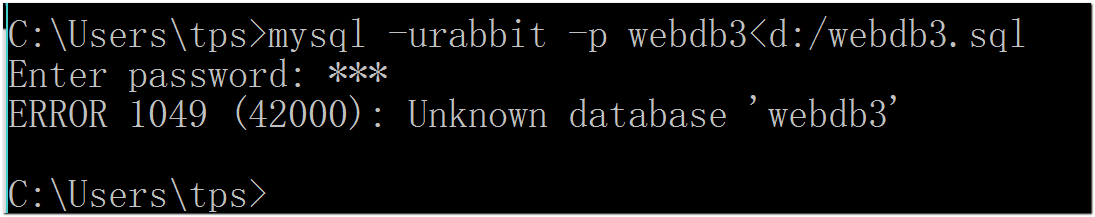
备份: mysqldump命令需要退出 在dos命令窗口直接使用.

删除数据库webdb3; 恢复刚刚删除的数据库以及表数据



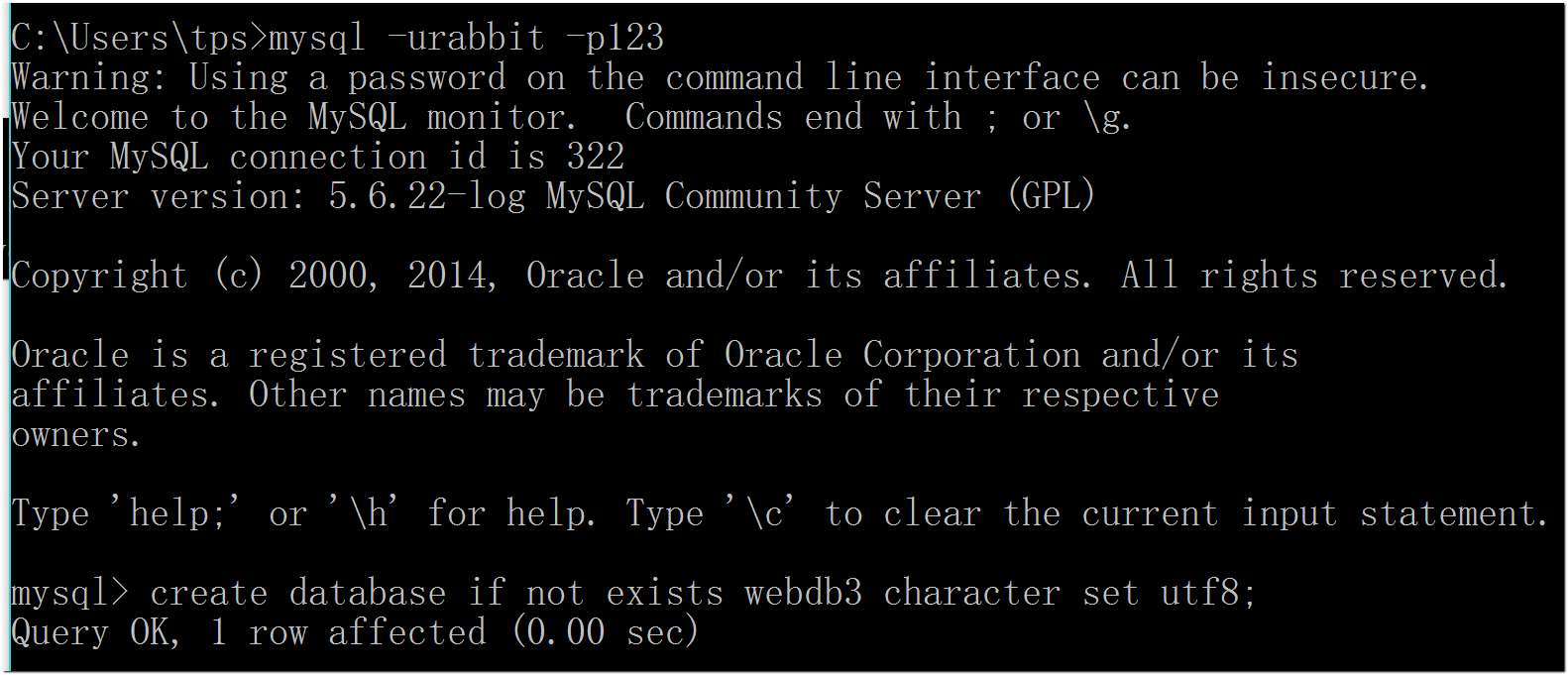
恢复: 使用rabbit用户恢复webdb3数据库数据

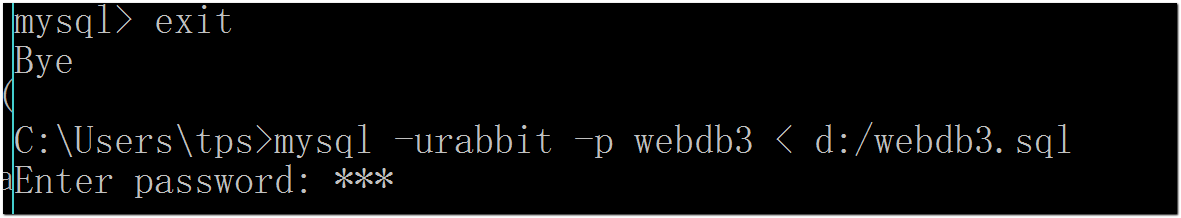
无需登录mysql,在dos下直接使用mysql恢复刚刚删除的webdb3数据库数据 如下:

需要注意,数据库备份的webdb3.sql脚本,没有创建数据库的命令,故而恢复时,执行sql脚本会报错,找不到数据库.

所以一般我们都会先创建原始的数据库名称:webdb3 然后再执行刚刚的恢复命令

登录mysql,创建数据库webdb3

当webdb3数据库存在 方可以恢复 退出mysql 恢复数据库数据

重新登录rabbit用户,发现数据库webdb3数据恢复完成

