实验题目	关系数据库管理系统与 SQL			实验日期	2022.04.03
班级	1903103	学号	1190200523	姓名	石翔宇

CS33503 数据库系统实验

实验检查记录

实验结果的正确性(60%)		表达能力(10%)	
实验过程的规范性(10%)		实验报告(20%)	
加分(5%)		总成绩(100%)	

实验报告

一、实验目的

1. 掌握一种关系数据库管理系统(RDBMS)的使用方法。
2. 学会使用 SQL 创建、修改、查询和控制关系数据库。

二、实验环境

硬件设备： Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz 2.59 GHz
软件系统： Windows 11 22H2、Ubuntu 20.04.4 LTS、MySQL Server version: 8.0.28-0ubuntu0.20.04.3
开发工具： Visual Studio Code 1.65.2

三、实验过程

实验过程及实现方法：

1. 练习 SQL 命令。
   a) 显示数据库列表：SHOW DATABASES;
   b) 选择创建的数据库 Product：USE Product;
   c) 显示所有的表名字：SHOW TABLES;
   d) 显示数据表 PC 的结构：DESCRIBE PC;
   e) 清空 PC 表中记录：DELETE FROM PC;
   f) 创建数据库 college：CREATE DATABASE college;
   g) 删除数据库 college：DROP DATABASE college;

2. 用 SQL 编写本课程第 3 章习题 11 中的全部数据库查询和更新语句。
   a) Find the manufacturers that sell laptops but not PC's. (使用集合差运算)：
   (SELECT DISTINCT maker FROM Product WHERE type = 'laptop') MINUS
   (SELECT DISTINCT maker FROM Product WHERE type = 'pc');
   b) Find the manufacturers that sell laptops but not PC's. (使用含有 IN 的嵌套查询)：SELECT DISTINCT maker FROM Product WHERE type = 'laptop'

```
    AND maker NOT IN (SELECT DISTINCT maker FROM Product WHERE type =
    'pc');
c) Find the manufacturers that sell laptops but not PC's. (使用含有 EXISTS
    的嵌套查询): SELECT DISTINCT maker FROM Product AS S WHERE type =
    'laptop' AND NOT EXISTS (SELECT * FROM Product AS T WHERE T.type
    = 'pc' AND S.maker = T.maker);
d) Find the model numbers of all printers that are cheaper than the printer
    model 3002. (使用内连接查询): SELECT P1.model FROM Printer AS P1
    JOIN Printer AS P2 on (P1.price < P2.price) AND P2.model =
    '3002';
e) Find the model numbers of all printers that are cheaper than the printer
    model 3002. (使用含有比较运算符的嵌套查询): SELECT model FROM Printer
    WHERE price < (SELECT price FROM Printer WHERE model = '3002');
f) Find the model numbers of all printers that are cheaper than the printer
    model 3002. (使用含有 EXISTS 的嵌套查询): SELECT model FROM Printer AS
    P1 WHERE EXISTS (SELECT * FROM Printer AS P2 WHERE P2.model =
    '3002' AND P1.price < P2.price);
g) Find the PC model with the highest available speed. (使用外连接查询):
    SELECT DISTINCT P1.model FROM PC AS P1 LEFT JOIN PC AS P2 ON
    P1.speed < P2.speed WHERE P2.model is NULL;
h) Find the PC model with the highest available speed. (使用含有 IN 的嵌套查
    询): SELECT model FROM PC WHERE speed IN (SELECT MAX(speed) FROM
    PC);
i) Find the PC model with the highest available speed. (使用含有=的嵌套查
    询): SELECT model FROM PC WHERE speed = (SELECT MAX(speed) FROM
    PC);
j) Find the PC model with the highest available speed. (使用含有>=的嵌套查
    询): SELECT model FROM PC WHERE speed >= ALL (SELECT speed FROM
    PC);
k) Find the PC model with the highest available speed. (使用含有 EXISTS 的嵌
    套查询): SELECT model FROM PC AS P1 WHERE NOT EXISTS (SELECT *
    FROM PC AS P2 WHERE P2.speed > P1.speed);
l) Find the manufacturers of PC's with at least three different speeds.
    (使用内连接查询): SELECT DISTINCT P1.maker FROM ((Product AS P1
    NATURAL JOIN PC AS PC1) JOIN (Product AS P2 NATURAL JOIN PC AS
    PC2) ON (P1.maker = P2.maker) JOIN (Product AS P3 NATURAL JOIN PC
    AS PC3) ON (P1.maker = P3.maker)) WHERE PC1.speed != PC2.speed
    AND PC1.speed != PC3.speed AND PC2.speed != PC3.speed;
m) Find the manufacturers of PC's with at least three different speeds.
    (使用分组查询): SELECT maker FROM Product NATURAL JOIN PC GROUP BY
    maker HAVING COUNT(DISTINCT speed) >= 3;
n) Find the manufacturers of PC's with at least three different speeds.
    (使用派生关系): SELECT maker FROM (SELECT maker, COUNT(DISTINCT
```

实验题目	关系数据库管理系统与 SQL			实验日期	2022.04.03
班级	1903103	学号	1190200523	姓名	石翔宇

speed) AS c FROM Product NATURAL JOIN PC GROUP BY maker) AS P WHERE P.c >= 3;

o) Decrease the price of all PC's made by maker A by 10%. (使用含有=的更新条件): UPDATE PC SET price = price * 0.9 WHERE 'A' = (SELECT maker FROM Product WHERE PC.model = Product.model);

p) Decrease the price of all PC's made by maker A by 10%. (使用含有 IN 的更新条件): UPDATE PC SET price = price * 0.9 WHERE model IN (SELECT model FROM Product WHERE maker = 'A');

q) Decrease the price of all PC's made by maker A by 10%. (使用含有 EXISTS 的更新条件): UPDATE PC AS P1 SET price = price * 0.9 WHERE EXISTS (SELECT * FROM Product AS P2 WHERE P1.model = P2.model AND P2.maker = 'A');

r) 从 SQL 语句的易读性和执行效率两方面对题目(g) - (k)的 SQL 语句进行分析和比较。(分析结果在"实验结果"部分)

实验结果:

1. 练习 SQL 命令。

a) 显示数据库列表:

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| Product            |
| college            |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0.00 sec)
```

b) 选择数据库 Product:

```
mysql> USE Product;
Database changed
```

c) 显示所有的表名字:

```
mysql> SHOW TABLES;
+------------------+
| Tables_in_Product |
+------------------+
| Laptop           |
| PC               |
| Printer          |
| Product          |
+------------------+
4 rows in set (0.00 sec)
```

d) 显示数据表 PC 的结构:

实验题目	关系数据库管理系统与 SQL			实验日期	2022.04.03
班级	1903103	学号	1190200523	姓名	石翔宇

```
mysql> DESCRIBE PC;
+-------+-------+------+-----+---------+-------+
| Field | Type  | Null | Key | Default | Extra |
+-------+-------+------+-----+---------+-------+
| model | int   | YES  |     | NULL    |       |
| speed | float | YES  |     | NULL    |       |
| ram   | int   | YES  |     | NULL    |       |
| hd    | int   | YES  |     | NULL    |       |
| price | int   | YES  |     | NULL    |       |
+-------+-------+------+-----+---------+-------+
5 rows in set (0.07 sec)
```

e) 清空 PC 表中记录：

```
mysql> DELETE FROM PC;
Query OK, 13 rows affected (0.07 sec)
```

f) 创建数据库 college：

```
mysql> CREATE DATABASE college;
Query OK, 1 row affected (0.07 sec)
```

g) 删除数据库 college：

```
mysql> DROP DATABASE college;
Query OK, 0 rows affected (0.08 sec)
```

2. 本课程第 3 章习题 11 中的全部数据库查询和更新语句执行结果。

a)-c) Find the manufacturers that sell laptops but not PC's. （由于 MySQL 限制，无法展示使用 MINUS 的语句）

```
mysql> SELECT DISTINCT maker FROM Product WHERE t
ype = 'laptop' AND maker NOT IN (SELECT DISTINCT
maker FROM Product WHERE type = 'pc');
+-------+
| maker |
+-------+
| F     |
| G     |
+-------+
2 rows in set (0.07 sec)

mysql> SELECT DISTINCT maker FROM Product AS S WH
ERE type = 'laptop' AND NOT EXISTS (SELECT * FROM
 Product AS T WHERE T.type = 'pc' AND S.maker = T
.maker);
+-------+
| maker |
+-------+
| F     |
| G     |
+-------+
2 rows in set (0.05 sec)
```

d)-f) Find the model numbers of all printers that are cheaper than the printer model 3002.

```
mysql> SELECT P1.model FROM Printer AS P1 JOIN Pr
inter AS P2 on (P1.price < P2.price) AND P2.model
 = '3002';
+-------+
| model |
+-------+
|  3001 |
|  3004 |
|  3005 |
|  3006 |
|  3007 |
+-------+
5 rows in set (0.04 sec)

mysql> SELECT model FROM Printer WHERE price < (S
ELECT price FROM Printer WHERE model = '3002');
+-------+
| model |
+-------+
|  3001 |
|  3004 |
|  3005 |
|  3006 |
|  3007 |
+-------+
5 rows in set (0.00 sec)

mysql> SELECT model FROM Printer AS P1 WHERE EXIS
TS (SELECT * FROM Printer AS P2 WHERE P2.model =
'3002' AND P1.price < P2.price);
+-------+
| model |
+-------+
|  3001 |
|  3004 |
|  3005 |
|  3006 |
|  3007 |
+-------+
5 rows in set (0.00 sec)
```

g)-k) Find the PC model with the highest available speed.

```
mysql> SELECT DISTINCT P1.model FROM PC AS P1 LEF
T JOIN PC AS P2 ON P1.speed < P2.speed WHERE P2.m
odel is NULL;
+-------+
| model |
+-------+
|  1005 |
|  1006 |
+-------+
2 rows in set (0.00 sec)

mysql> SELECT model FROM PC WHERE speed IN (SELEC
T MAX(speed) FROM PC);
+-------+
| model |
+-------+
|  1005 |
|  1006 |
+-------+
2 rows in set (0.04 sec)
```

```
mysql> SELECT model FROM PC WHERE speed = (SELECT
 MAX(speed) FROM PC);
+-------+
| model |
+-------+
|  1005 |
|  1006 |
+-------+
2 rows in set (0.00 sec)

mysql> SELECT model FROM PC WHERE speed >= ALL (S
ELECT speed FROM PC);
+-------+
| model |
+-------+
|  1005 |
|  1006 |
+-------+
2 rows in set (0.00 sec)

mysql> SELECT model FROM PC AS P1 WHERE NOT EXIST
S (SELECT * FROM PC AS P2 WHERE P2.speed > P1.spe
ed);
+-------+
| model |
+-------+
|  1005 |
|  1006 |
+-------+
2 rows in set (0.00 sec)
```

l)-n) Find the manufacturers of PC’s with at least three different speeds.

```
mysql> SELECT DISTINCT P1.maker FROM ((Product AS
 P1 NATURAL JOIN PC AS PC1) JOIN (Product AS P2 N
ATURAL JOIN PC AS PC2) ON (P1.maker = P2.maker) J
OIN (Product AS P3 NATURAL JOIN PC AS PC3) ON (P1
.maker = P3.maker)) WHERE PC1.speed != PC2.speed
AND PC1.speed != PC3.speed AND PC2.speed != PC3.s
peed;
+-------+
| maker |
+-------+
| A     |
| D     |
| E     |
+-------+
3 rows in set (0.00 sec)

mysql> SELECT maker FROM Product NATURAL JOIN PC
GROUP BY maker HAVING COUNT(DISTINCT speed) >= 3;

+-------+
| maker |
+-------+
| A     |
| D     |
| E     |
+-------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT maker FROM (SELECT maker, COUNT(DIS
TINCT speed) AS c FROM Product NATURAL JOIN PC GR
OUP BY maker) AS P WHERE P.c >= 3;
+-------+
| maker |
+-------+
| A     |
| D     |
| E     |
+-------+
3 rows in set (0.00 sec)
```

o)-q) Decrease the price of all PC's made by maker A by 10%. （每个命令执行完之后会将价格复原）

```
mysql> UPDATE PC SET price = price * 0.9 WHERE 'A
' = (SELECT maker FROM Product WHERE PC.model = P
roduct.model);
Query OK, 3 rows affected (0.06 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> SELECT price FROM PC NATURAL JOIN Product
WHERE maker = 'A';
+-------+
| price |
+-------+
|  1903 |
|   896 |
|   430 |
+-------+
3 rows in set (0.00 sec)
```

```
mysql> UPDATE PC SET price = price * 0.9 WHERE mo
del in (SELECT model FROM Product WHERE maker = '
A');
Query OK, 3 rows affected (0.04 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> SELECT price FROM PC NATURAL JOIN Product
WHERE maker = 'A';
+-------+
| price |
+-------+
|  1903 |
|   896 |
|   430 |
+-------+
3 rows in set (0.00 sec)
```

```
mysql> UPDATE PC AS P1 SET price = price * 0.9 WH
ERE EXISTS (SELECT * FROM Product AS P2 WHERE P1.
model = P2.model AND P2.maker = 'A');
Query OK, 3 rows affected (0.03 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> SELECT price FROM PC NATURAL JOIN Product
WHERE maker = 'A';
+-------+
| price |
+-------+
|  1903 |
|   896 |
|   430 |
+-------+
3 rows in set (0.01 sec)
```

实验题目	关系数据库管理系统与 SQL			实验日期	2022.04.03
班级	1903103	学号	1190200523	姓名	石翔宇

r)对(g) - (k)的 SQL 语句进行分析和比较：

易读性：(i)的易读性最好，(h)和(j)的易读性相对较差，(g)和(k)的易读性在 5 个 SQL 语句中最差。

执行效率：由于(h)和(i)只需要对 PC 表扫描 2 次，与其他的语句效率相比较高；而因为在 speed 上没有索引，所以(g)、(j)和(k)的效率较差。

四、实验结论

通过本次实验，我对于关系数据库管理系统 MySQL 的使用方法有了更加深刻的掌握，也学会了通过不同的语句来创建、修改、查询和控制关系数据库，体会到了数据库的方便与魅力。