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（介绍实验目的）

|  |
| --- |
| 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（介绍实验使用的硬件设备、软件系统、开发工具等）

|  |
| --- |
| 1、硬件设备：  i7-9750H CPU@2.60GHz 2.59GHz; 1.8GHz; 16G RAM;  2、软件系统  Ubuntu 20.04.4 LTS (GNU/Linux 5.10.102.1-microsoft-standard-WSL2 x86\_64)  3、开发工具  MySQL |

三、实验过程（介绍实验过程、设计方案、实现方法、实验结果等）

|  |
| --- |
| **1、连接数据库**  输入命令“mysql -u root -p”并按照提示输入密码    **2、MySql常用命令**  2.1 显示数据库列表  输入命令“show databases;”查看当前服务器上的数据库。    2.2 切换数据库  输入命令“use 数据库名；”切换当前使用的数据库。    2.3 查看表  输入命令“show tables；”可以显示当前数据库的所有表名。    2.4 查看关系  输入命令“desc 表名；”可以查看对应关系    2.5 导入数据库  命令：source sql文件路径    2.6 查看参数值  命令： show variables like "参数名";    2.7退出MySql  命令：exit;    **3、实用SQL命令**  3.1 创建数据库  输入命令“create database 数据库名;”来创建数据库。    3.2 删除数据库  输入命令“drop database 数据库名;”删除对应数据库。    3.3 修改表名称    **4、创建数据库**  4.1新建数据库并使用该数据库    4.2 创建关系          4.3修改关系      4.4 添加元组      4.5 删除元组      4.6 修改元组      **五、在mysql中验证本课程第3章例子中给出的College数据库上的SQL语句的正确性。**  （SQL数据定义和SQL数据更新的例子在上述实验过程中均有验证。）  5.1 单关系查询  投影查询：  查询学生的学号和姓名    查询所有的系名    查询全部学生信息    扩展的投影查询：  查询学生的学号和姓名（姓名全大写）    查询学生的姓名和出生年份    选择查询：  查询计算机系（CS）全体学生的学号和姓名    查询计算机系（CS）全体男同学的学号和姓名    查询计算机系（CS）和数学系（Math）全体学生的学号和姓名    字符串匹配：  查询首字母为E的学生的学号和姓名    查询姓名为四个字母且首字母为E的学生的学号和姓名    正则表达式：  查询姓名首字母为E或F的学生的学号和姓名    空值判断：  查询选了课但还未取得成绩的同学    集合操作：  查询选修了1002号或3006号课的学生的选课信息    查询选修了1002号或3006号课的学生的学号    结果排序  查询计算机系（CS）全体学生的学号和姓名，并按照学号升序排列    查询全体学生的信息，结果按所在系升序排列，同一个系得学生按年龄降序排列    限制查询结果数量：  查询3006号课得分最高的前2名学生的学号和成绩    聚集查询：  查询计算机系全体学生的数量    查询计算机系学生的最大年龄    分组查询：  统计每门课的选课人数和平均成绩    统计每个系的男生人数和女生人数    查询选修了两门以上课程的学生的学号和选课数    查询两门以上课程得分超过80的学生的学号以及这些课程的平均分    5.2连接查询  笛卡尔积：  查询学生及其选课情况，列出学号、姓名、课号、得分    内连接：  查询学生及其选课情况，列出学号、姓名、课号、得分      自然连接：  查询学生及其选课情况，列出学号、姓名、课号、得分    自连接：  查询和Elsa在同一个系学习的学生的学号和姓名    外连接：  查询没有选课的学生的姓名和学号：    5.3嵌套查询  子查询的类型：  查询和Elsa在同一个系学习的学生得学号和姓名（含Elsa）  不相关子查询    相关子查询    **六、 将Product.sql文件下载到当前工作目录，创建Product数据库，然后用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 Product.type = 'laptop' AND Product.maker != ALL (SELECT 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 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 P1 WHERE P1.type = 'laptop' AND NOT EXISTS(SELECT \* FROM Product AS P2 WHERE P1.type = 'laptop' AND P2.type = 'pc' AND P1.maker = P2.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 (P2.model = 3002 AND P1.price < p2.price);    (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 P1.price < P2.price AND P2.model = 3002);    (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.price 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 P1.speed < P2.speed);    (l) Find the manufacturers of PC’s with at least three different speeds. (使用内连接查询)  SELECT maker FROM (SELECT DISTINCT maker, speed FROM PC JOIN Product ON (PC.model = Product.model)) AS P GROUP BY maker HAVING COUNT(\*) >= 3;    (m) Find the manufacturers of PC’s with at least three different speeds. (使用分组查询)  SELECT maker FROM (SELECT DISTINCT maker, speed FROM PC JOIN Product ON (PC.model = Product.model)) AS P group BY maker HAVING COUNT(\*) >= 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(\*) FROM (SELECT DISTINCT speed, maker FROM PC JOIN Product ON (PC.model = Product.model)) AS TEMP GROUP BY maker) AS P(maker, num) WHERE num >= 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 model = ANY (SELECT model FROM Product WHERE maker = 'A');      (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 SET price = price \* 0.9 WHERE EXISTS (SELECT model FROM Product WHERE maker = 'A' AND PC.model = Product.model);    (r) 题目(g)–(k)用不同方法编写相同的查询。请你从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。在做效率分析时，我们假定每个关系上只有主索引，而没有其他索引(请自学第6章中索引的概念和功能)。  易读性：我认为（h）（i）（j）的可读性最高，因为其语句简短，并且使用了in、比较符号这样的值观简单的数学关系符号，更加直观。而（g）语句是按照大小关系将所有的元素进行外连接，然后找到右侧为空的元素，（k）是采用了判断是否存在比当前元素更大的元素来寻找最大值，这两种方式都不是人们平时在寻找最大值时的常规思路。  执行效率：（g）仅需要执行一次外连接，效率较高，而（h）（i）（j）（k）需要执行两次连接，效率较低。 |

四、实验结论（总结实验发现及结论）

|  |
| --- |
| 数据库管理系统是十分高效、便捷的数据管理方式，它可以通过连接、嵌套、单关系查询等的相互组合来实现各种各样的查询操作，而对于同一种操作的不同实现方式，其优缺点也是不尽相同，因此拥有清晰的逻辑对于使用DBMS是十分重要的。 |