

---

## 一、实验目的：

- 1、了解 DBMS 系统的功能、软件组成；
- 2、掌握利用 SQL 语句定义、和简单操纵数据库的方法。

## 二、实验要求：

- 1、基于实验一已经建立的表格练习典型的多表连接查询 SQL 语句
- 2、练习聚合函数
- 3、练习视图的创建查询、插入、更新、删除等操作

## 三、实验设备：

计算机、数据库管理系统如 DB2，Oracle 等软件。

## 四、实验内容

根据已经创建的数据库关系表 EMP+(学生自己的学号)和 DEPT+(学生自己的学号)完成 SQL handbook 中的老师指定的较难题目：

以下为学生实验部分，所有题目要求有语句和运行结果截屏：

### 题目 2.4. List all departments that do not have any employees.

SQL 语句：

```
SELECT DEPTNO FROM dept
WHERE DEPTNO not in (
    SELECT DEPTNO from emp)
```

运行结果：

DEPTNO
40

### 题目 2.5 For each employee whose salary exceeds his manager's salary, list the employee's name and salary and the manager's name and salary.

SQL 语句：

```
SELECT worker.ENAME,worker.SAL,manager.ENAME,manager.SAL
FROM emp worker, emp manager
WHERE worker.MGR = manager.EMPNO
AND worker.SAL > manager.SAL
```

运行结果：

ENAME	SAL	ENAME	SAL
-------	-----	-------	-----

---

**题目 2.6. List the employees who have BLAKE as their manager.**

**SQL 语句:**

```
SELECT worker.ENAME,manager.ENAME
FROM emp worker, emp manager
WHERE worker.MGR = manager.EMPNO
AND manager.ENAME = 'BLAKE'
```

**运行结果:**

ENAME	ENAME
ALLEN	BLAKE
WARD	BLAKE
MARTIN	BLAKE
TURNER	BLAKE
JAMES	BLAKE

**题目 6.3 List the name, job, and department of employees who have the same job as Jones or a salary greater than or equal to Ford.**

**SQL 语句:**

```
SELECT DISTINCT
    e1.ENAME, e1.JOB, e1.DEPTNO
FROM
    emp e1, emp e2
WHERE
    ((e2.ENAME = 'jones' AND e1.JOB = e2.JOB)
    OR
    (e2.ENAME = 'ford' AND e1.SAL >= e2.SAL))
AND
    (e1.ENAME != 'jones')
AND
    (e1.ENAME != 'ford')
```

**运行结果:**

ENAME	JOB	DEPTNO
STEVENS	MANAGER	50
BLAKE	MANAGER	30
CLARK	MANAGER	10
KING	PRESIDENT	10

**题目 6.6 Find all the employees that earn more than the average salary of employees in their department.**

**SQL 语句:**

```
SELECT
    e.ENAME
FROM
```

```
emp e, (SELECT AVG(SAL) AVG_SAL, DEPTNO FROM emp GROUP BY DEPTNO) s
WHERE
e.DEPTNO = s.DEPTNO AND e.SAL > s.AVG_SAL
```

运行结果:

ENAME
KING
JONES
SCOTT
FORD
ALLEN
WARD
BLAKE
TURNER
GREEN
STEVENS

题目 Ch5 p33: 用 SQL 实现 A/B2

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

pno
p1
p2
p4

B3

sno
s1
s2
s3
s4

A/B1

sno
s1
s4

A/B2

sno
s1

A/B3

首先根据图中的数据，在数据库创建 Table A 和 Table B2，SQL 语句如下：

```
CREATE Table A(sno char(2),pno char(2))
```

```
CREATE Table B2 (pno char(2))
```

然后在 Table B2 中插入数据，SQL 语句如下：

```
INSERT INTO B2 VALUES('p2');
```

```
INSERT INTO B2 VALUES('p4');
```

然后在 Table A 中插入数据，SQL 语句如下：

```
INSERT INTO A VALUES('s1','p1');
```

```
INSERT INTO A VALUES('s1','p2');
```

```
INSERT INTO A VALUES('s1','p3');
```

```
INSERT INTO A VALUES('s1','p4');
```

```
INSERT INTO A VALUES('s2','p1');
```

```
INSERT INTO A VALUES('s3','p2');
```

```
INSERT INTO A VALUES('s4','p2');
```

```
INSERT INTO A VALUES('s4','p4');
```

这样就完成了建表，开始进行查询：

SQL 语句:

```
SELECT DISTINCT sno FROM A X WHERE NOT EXISTS  
(SELECT pno FROM B2 WHERE NOT EXISTS  
(SELECT * FROM A WHERE sno=X.sno AND pno = B2.pno));
```

运行结果:

sno
s1
s4

解释说明:

因为本题比较重要且刚开始会理解比较困难,所以我在这里再做一次思路梳理。先对课堂讲到的“查询选修了全部课程的学生姓名”这个题目进行再次的理解。

```
SELECT Sname FROM Student WHERE NOT EXISTS  
(SELECT * FROM Course WHERE NOT EXISTS  
(SELECT * FROM SC WHERE Sno=Student.Sno AND Cno=Course.Cno));
```

这算是一个比较复杂的 sql 语句,两个 EXISTS 和三个 WHERE。其中,使用 EXISTS 后,若内层查询结果存在(为非空),则外层的 WHERE 子句返回值为真,否则返回值为假;与 EXISTS 相对的是 NOT EXISTS,使用存在量词 NOT EXISTS 后,若对应查询结果不存在(为空),则外层的 WHERE 子语句返回值为真值,否则返回假值。

这个 SQL 语句可以分为 3 层,最外层语句,最内层语句,中间层语句。结果表中的数据都是最外层的查询的表中的数据,最内层的数据包含了全部的判断语句,决定了 student 表中的哪一条记录是我们查询的记录。分析如下:

先取一条 student 记录,进入中层,再取一条 course 的记录,进入内层,此时 student 的记录和 course 的记录作为内层判断的条件(与 Enrollment 表中记录比较),相当于做了两个 for 循环进行遍历。假设取的第一条记录是 A 同学(Sno 为 123456),此时里面的 Sql 语句其实可以写成:

```
SELECT * FROM Course WHERE NOT EXISTS  
(SELECT * FROM SC WHERE Sno = '123456' AND Cno=Course.Cno));
```

此处 sno=123456 即 A 同学的学号,这条 SQL 的意思是选出没有被 A 同学选择的课程,如果不存在,则返回 false,再跟最外层的 NOT EXISTS 关联,负负得正。每一条循环的意思就是指,筛选出的每一个学生都不存在没有被他选取的那门课,即选了所有课。

有了上述题目的铺垫,本题目要查询所有供应了 p2 和 p4 的供应商,只需要先从 A 中选取 sno,从 B2 中选取 pno,组成一个组合,再和 A 表的组合进行对比,同样使用两个 NOT EXISTS 来进行判断。这里跟例题不同的地方在于,因为没有单独一个 sno 表,所以从 A 表中选取 sno 时,要用表内连接。

## 五. 实验心得

通过本次实验,我对数据库和数据表有了更深的理解。

对于第 1 道题其实使用 not in 后,就可以大大提高语句的效率。

---

对于第 4 道题，一定要记得加上 `(e1.ENAME != 'jones') AND (e1.ENAME != 'ford')` 的语句，将 `jones` 和 `ford` 从查询结果中删除，其实这个点在课堂上也有习题讲到了，需要引起细心注意。

对于最后一道题，我在课堂上就以为我理解了，但是当遇到这个新题目时，就不会运用了，说明了其实还没有理解得很深刻，所以我又重新从课堂的题目入手，通过画逻辑图，终于弄得明明白白了，其实主要是注意两个点，一个是通过两个 `for` 循环从两个表中循环取两个不同的数形成一个组合，与第三个表进行对比，看是否存在于第三个表中；第二个要理解 `exists` 和 `not exists` 的含义，从内到外进行判断，两个 `not exists` 有相当于“负负得正”的效果。以后遇到类似的题目不怕了。

通过几句简单的 SQL 语句，就可以实现复杂的需求，我不止一次被数据库的魅力所折服，觉得其十分适合用来关系型数据分析和数据管理。通过这次实验，我的数据库水平提高了一个档次！也期待之后的学习！

## 六. 诚信承诺

本人郑重承诺在 SQL 实验的实施的过程中不发生任何不诚信现象，一切不诚信所导致的后果均由本人承担。

签名（手签，不得打印）： 谢晓锋

指导教师批阅意见：

成绩评定：