

中山大学数据科学与计算机学院 移动信息工程专业-数据库系统 本科生实验报告

(2017-2018 学年秋季学期)

课程名称:数据库系统实验

教学班级	15M1	专业 (方向)	移动互联网
学号	15352408	姓名	张镓伟

一、 实验目的

- 1. 熟悉 SQL 的数据更新操作。
- 2. 能够使用 SQL 语句对数据库进行数据插入、更新和删除操作。

二、 实验内容

本次实验中,主要的内容是如何使用 SQL 语句对数据进行更新。

- a.使用 INSERT INTO 语句插入数据,包括插入一个元组或将子查询的结果插入到数据库中两种方式。
- b.使用 SELECT INTO 语句,产生一个新表并插入数据。
- c.使用 **UPDATE 语句**可以修改指定表中满足 WHERE 子句条件的元组,有三种修改方式:修改某一个元组的值,修改多个元组的值,带子查询的修改语句。
- **d.**使用 **DELETE 语句**删除数据:删除某一个元组的值,删除多个元组的值,带子查询的删除语句。

三、 实验过程及结果

(1)向 STUDENTS 表插入编号是"800022222"且姓名是"WangLan"的元组。 Sql 语句:

```
INSERT
INTO STUDENTS
-VALUES ('8000222222', 'WangLan', NULL, NULL)
-SELECT * FROM STUDENTS WHERE STUDENTS.sid = '800022222'
```

查询结果:



可以看到已成功向数据库插入这条信息

(2)向 TEACHERS 表插入元组("200001000", "LXL", "s4zrck@pew,net", "3024")。 Sql 语句:



```
INSERT
INTO TEACHERS
-VALUES ('200001000', 'LXL','s4zrck@pew,net','3024')
-SELECT * FROM TEACHERS WHERE TEACHERS.tid = '200001000'
```

查询结果如图:

	tid	tname	email	salary
1	200001000	LXL	s4zrck@pew,net	3024

可以看到已成功向数据库插入这条信息

(3)将 TEACHERS 表中编号为"200010493"的老师工资改为 4000。

Sql 语句:

```
DUPDATE TEACHERS

SET salary=4000

WHERE tid='200010493'

SELECT * FROM TEACHERS WHERE TEACHERS.tid = '200010493'
```

查询结果如图:

	tid	tname	email	salary
1	200010493	xzgkflko	zcf_m@def.com	4000

可以看到已成功向数据库插入这条信息

(4)将 TEACHERS 表中所有工资小于 2500 的老师工资改为 2500。

Sql: 语句:

```
UPDATE TEACHERS

SET salary=2500

WHERE salary< 2500

SELECT * FROM TEACHERS WHERE salary<2500
```

查询结果如下:



修改了之后我查询了工资小于 **2500** 的老师,结果是一个都没有,也就是说修改成功了

(5)将编号为"200016731"的老师讲授的课程全部改成由姓名为"rnupx"的老师讲授。 Sql 语句:



```
UPDATE CHOICES

SET tid=(

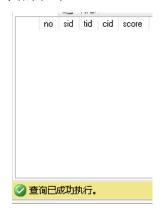
SELECT tid

FROM TEACHERS

WHERE tname='rnupx'
)
-WHERE tid='200016731'

-select *from CHOICES WHERE tid='200016731'
```

查询结果如下:



更新了之后查询了原来编号为 2000167313 的老师还有没有课,结果是没有,说明修改成功。

(6)更新编号"800071780"的学生年级为"2001"。 Sql 语句:

```
J UPDATE STUDENTS
| SET grade='2001'
|-WHERE sid='800071780'
| SELECT * FROM STUDENTS WHERE sid='800071780'
```

查询结果如下:

	sid	sname	email	grade
1	800071780	iscmwink	htoactp@aof.gov	2001

(7)删除没有学生选修的课程。

Sql 语句:

```
FROM COURSES
WHERE cid NOT IN(
select cid
from CHOICES
```

查询结果如下:





从第一次实验就知道所有课都有人选修,所以没有课被删除

(8)删除年级高于 1998 的学生信息。

Sql 语句:由于 CHOICES 中有外键的存在,不能够直接删除 STUDENTS 表的数据,需先删除 CHOICES 表的数据。

```
| DELETE
| FROM CHOICES
| WHERE sid in (
| SELECT sid
| FROM STUDENTS
| WHERE grade<1998
|-)
| DELETE |
| FROM STUDENTS
| WHERE grade<1998
```

查询结果如下:

```
inle (139750 行受影响)
(46600 行受影响)

Select * from STUDENTS
- WHERE grade<1998
```

查询 grade 比 1998 高的学生,发现已经没有记录了

(9)删除没有选修课程的学生信息。

Sql 语句如下:

```
| DELETE
| FROM STUDENTS
| WHERE sid NOT IN(
| SELECT sid
| FROM CHOICES
|-)
```

查询结果如下,只有1个人没有选课,这个人是我们之前新插入的:

```
(1 行受影响)
|select * FROM STUDENTS
WHERE sid NOT IN(
SELECT sid
FROM CHOICES
)
| sid |sname | email | grade
```

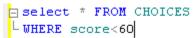
重新查询没有选课的学生,发现没有记录



(10)删除成绩不及格的选课记录。

Sql 语句及结果如下:

首先查询不及格的记录:



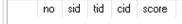
	no	sid	tid	cid	score
1	500000984	850955252	234145610	10021	54
2	500015294	888484354	276307538	10050	54
3	500017594	893241017	258724747	10023	51
4	500019253	826800279	254743764	10042	52
5	500030331	848113583	243330265	10034	51
6	500032049	822809365	299638722	10012	50
7	500040843	860579754	286542488	10020	53
8	500041694	817917365	215345279	10022	53
9	500044694	866606290	235091038	10006	52
10	500056030	819331388	274499985	10022	51

然后删除:

```
□ DELETE
| FROM CHOICES
| WHERE score<60
```



重新查询不及格记录,发现已经没有记录了:



四、 实验感想

这次实验是熟悉 sql 语句中的更新语句。这次的练习涉及到 SQL 更新语句的 UPDATE 和 DELETE,这次实验比较简单,不过写语句的时候逻辑要清晰,不然删除语句写错就会删错记录,这个时候适时地备份数据库就显得尤为重要。