

数据库实验报告 实验六 数据更新

姓名	学号	班级	课室
熊明	20305055	计科5班	D503

一、实验目的

熟悉数据库的数据更新操作，能够使用SQL语句对数据库进行数据的插入、更新、删除操作。

二、实验环境

数据库：Mysql

图形化工具：Navicat Premium 16

三、实验内容

在本次实验中，主要的内容是如何使用SQL语句对数据进行更新。

本节实验的主要内容包括：

- 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。
- 使用SELECT INTO语句，产生一个新表并插入数据。
- 使用UPDATE 语句可以修改指定表中满足WHERE子句条件的元组，有三种修改的方式：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。
- 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。

注："SELECT INTO"是SQL查询语句的一部分，用于将查询的结果插入到新表中。它的语法如下：

```
1 SELECT column1, column2, ...
2 INTO new_table
3 FROM existing_table
4 WHERE condition;
```

其中：

- column1, column2, ... 是要选择的列。
- new_table 是要将结果插入的新表。
- existing_table 是要从中选择数据的现有表。
- condition 是可选的筛选条件。

这个语句执行时，它会从现有表中选择指定列的数据，并将结果插入到新表中。

四、课内实验

1. 使用SQL语句向STUDENTS表中插入元组(编号：700045678；名字：LiMing；EMAIL：LX@cdeg.com；年级：1992)

```
1 INSERT INTO STUDENTS VALUES (700045678,"LiMing","LX@cdemg.com",1992);
```

结果如下:

sid	sname	email	grade
700045678	LiMing	LX@cdemg.com	1992
800001216	gfxrgs	hhce4@qhldj.gov	1992
800002933	vnbqz%svv	pvhxd4l@zqur.org	2002
800005753	waqcj	hlhq0h8@jdba.gov	1992
800006682	fiiluommh	ihzd6_k@kzvft.gov	1992
800006941	ogvmu	62sfbdlrt.gov	1995
800007595	uxqqbkjn	cr8g@zrvgt.edu	1997
800008565	ehlycg	nach10@uic.com	1999
800009026	rcxaihj	4ul4kqb@hko.edu	2002
800009099	zapyv	jmqn8@iwaiu.org	1992
800009249	zyuoh	8enjrcu@upfw.org	1991
800010666	uwphrw	emb7k@ipp.com	1992

2. 对每个课程, 求学生的选课人数和学生的平均成绩, 并把结果存入数据库。使用SELECT INTO和INSERT INTO两种方法实现。(提示: 可先创建一个新表再插入数据)

首先创建表 ans2 来存储数据

```
1 CREATE TABLE ans2(  
2   cid INT,  
3   nums INT,  
4   avg_socre double)
```

1. select into方法

```
1 SELECT cid,COUNT(sid) as nums,AVG(score) as avg_socre  
2 INTO ans2  
3 FROM choices  
4 GROUP BY cid
```

利用上述表达一直报错, 1327 - Undeclared variable:ans2。上网找资料发现mysql不支持select into语法。故用别的方法代替:

```
1 CREATE TABLE ans2(  
2   SELECT cid,COUNT(sid) as nums,AVG(score) as avg_socre  
3   FROM choices  
4   GROUP BY cid)
```

部分结果如下:

cid	nums	avg_socre
▶ 10001	5898	75.9724
10002	6013	75.8944
10003	5975	75.9430
10004	6110	76.1287
10005	6031	76.0171
10006	6090	76.0700
10007	5965	75.9119
10008	5985	75.6591
10009	5965	76.2120
10010	6027	75.8274
10011	6086	76.3341
10012	5819	76.1281
10013	5982	75.6476
10014	5866	76.1030
10015	5990	75.8497
10016	5867	76.0415
10017	5992	76.0847

2. INSERT INTO方法

```

1  INSERT INTO ans2(
2  SELECT cid,COUNT(sid) as nums,AVG(score) as avg_socre
3  FROM choices
4  GROUP BY cid)

```

部分结果如下：

	cid	nums	avg_socre
▶	10001	5898	75.972401103
	10002	6013	75.894402035
	10003	5975	75.943018387
	10004	6110	76.128746231
	10005	6031	76.017144919
	10006	6090	76.06996252
	10007	5965	75.911903895
	10008	5985	75.659107402
	10009	5965	76.211955531
	10010	6027	75.827411167
	10011	6086	76.334104251
	10012	5819	76.128085265
	10013	5982	75.647606868
	10014	5866	76.10301461
	10015	5990	75.849673202
	10016	5867	76.041512915
	10017	5992	76.084659913

3. 在STUDENTS表中使SQL语句将姓名为“LiMing”的学生的年级改为“2002”

```
1 | UPDATE students SET grade = 2002 WHERE sname = "LiMing";
```

结果如下：

	sid	sname	email	grade
▶	700045678	LiMing	LX@cdemg.com	2002
	800001216	gfxrgs	hhce4@qhldj.gov	1992

4. 在TEACHERS表中使SQL语句将所有教师的工资多加500元

```
1 | UPDATE teachers SET salary = salary + 500;
```

原先：

tid	tname	email	salary
▶ 200003125	fqmmmyi	wcjcg@glq.net	3928
200005322	udjom	gd5l8@ppmlf.gov	873
200009423	jhoulvtr	12cj@pzc.gov	3286
200010493	xzgkflko	zcf_m@def.com	3332
200011899	lwujzim	s4zrck@pew.net	(Null)
200016731	nfgrod	dprcy@qxqi.com	3819
200017039	qibsy	b8mtpe@enbba.edu	4179
200023671	zfuxiahq	lr_32t@cfo.net	3677
200025513	rnupx	c7ce@ayt.org	2524
200026895	nuyoq	xibl@izd.edu	4830
200033112	awboicq	hjct6g@izgm.edu	3632
200047819	lmogygb	if095@dxdm.org	3968
200050683	bmwdtty	uw_6@ozpe.edu	955
200058185	xcoybrz	dmis@peg.gov	3727

修改后:

tid	tname	email	salary
▶ 200003125	fqmmmyi	wcjcg@glq.net	4428
200005322	udjom	gd5l8@ppmlf.gov	1373
200009423	jhoulvtr	12cj@pzc.gov	3786
200010493	xzgkflko	zcf_m@def.com	3832
200011899	lwujzim	s4zrck@pew.net	(Null)
200016731	nfgrod	dprcy@qxqi.com	4319
200017039	qibsy	b8mtpe@enbba.edu	4679
200023671	zfuxiahq	lr_32t@cfo.net	4177
200025513	rnupx	c7ce@ayt.org	3024
200026895	nuyoq	xibl@izd.edu	5330
200033112	awboicq	hjct6g@izgm.edu	4132
200047819	lmogygb	if095@dxdm.org	4468
200050683	bmwdtty	uw_6@ozpe.edu	1455
200058185	xcoybrz	dmis@peg.gov	4227

5. 将姓名为zapvyv的学生的课程“C”的成绩加上5分

首先利用下面语句，查询原始分数是多少：

```

1 SELECT *
2 FROM choices
3 WHERE sid = (SELECT sid
4 FROM students
5 WHERE sname = "zapyv")
6 AND cid = (
7 SELECT cid
8 FROM courses
9 WHERE cname = "C")

```

结果如下：

no	sid	tid	cid	score
541282386	800009099	264496827	10042	80

然后更新数据：

```

1 UPDATE choices SET score = score + 5 WHERE sid =
2 (SELECT sid
3 FROM students
4 WHERE sname = "zapyv") AND cid =
5 (SELECT cid
6 FROM courses
7 WHERE cname = "C");

```

最后更新完数据结果如下：

no	sid	tid	cid	score
541282386	800009099	264496827	10042	85

6. 在STUDENTS表中使用SQL语句删除姓名为“LiMing”的学生信息

原始状态为：

sid	sname	email	grade
700045678	LiMing	LX@cdemg.com	2002
800001216	gfxrgs	hhce4@qhldj.gov	1992

执行下列删除语句：

```

1 DELETE FROM students WHERE sname = "LiMing";

```

最终状态如下：

sid	sname	email	grade
800001216	gfxrgs	hhce4@qhldj.gov	1992
800002933	vnbqz%svv	pvhxd4l@zqur.org	2002

7. 删除所有选修课程“Java”的选课记录

利用面语句原本状态：

```

1 SELECT *
2 FROM choices
3 WHERE cid = (
4 SELECT cid
5 FROM courses
6 WHERE cname = "Java")

```

no	sid	tid	cid	score
500007637	833961570	209860626	10004	80
500017515	856259316	292690214	10004	93
500025137	840205068	214988004	10004	76
500045819	870755799	247332115	10004	65
500079832	811353704	220627919	10004	60
500133744	881617726	266625150	10004	52
500147952	806836750	271311516	10004	69
500155554	898186633	213071784	10004	64

然后执行下述语句：

```

1 DELETE FROM choices WHERE cid = (
2 SELECT cid
3 FROM courses
4 WHERE cname = "Java");

```

最后查询，结果为空

no	sid	tid	cid	score
(N/A)	(N/A)	(N/A)	(N/A)	(N/A)

8. 对COURSES表做删去时间<48的元组的操作，并讨论该删除操作所受到的约束

查询courses表时间<48的元素：

```

1 SELECT *
2 FROM courses
3 WHERE `hour` <48

```

部分结果如下：

cid	cname	hour
▶ 10007	uml	30
10009	cryptology	36
10011	distributed computing	36
10012	erp	40
10013	artificial intelligence	46
10016	data mining	40
10018	unix/linux	40
10019	jsp	28
10020	j2ee	46
10021	j2me	40

执行下述删除操作：

```
1 | DELETE FROM courses WHERE `hour` < 48
```

报错：

```
1451 - Cannot delete or update a parent row: a foreign key constraint fails
(school.choices, CONSTRAINT FK_CHOICES_COURSES FOREIGN KEY (cid) REFERENCES
courses (cid))`
```

原因是被删除的元组中，有的cid作为choices表的外键存在，该元组不能被删除。

若要删除该元组，方法一可以同时将choices对应外键的元素删除。方法二解除外键约束检查，删除后再恢复外键检查，但是可能会造成一些错误。

五、自我实践

1. 向STUDENTS表插入编号是“800022222”且姓名是“WangLan”的元组

```
1 | INSERT INTO students VALUES (800022222, "WangLan")
```

该语句会报错 1136 - Column count doesn't match value count at row 1

```
1 | INSERT INTO students VALUES (800022222, "WangLan", NULL, NULL)
```

该语句正常执行，结果如下：

800022091	kdthgvooo	ejxo@ooaw.gov	1994
▶ 800022222	WangLan	(Null)	(Null)

2. 向TEACHERS表插入元组("200001000", "LXL", "s4zrck@pew.net", "3024")

```
1 | INSERT INTO teachers VALUES (200001000, "LXL", "s4zrck@pew.net", 3024)
```

结果如下：

tid	tname	email	salary
200001000	LXL	s4zrck@pew.net	3024
200003125	fqmmyi	wcjc@glq.net	4428

3. 将TEACHERS表中编号为“200010493”的老师工资改为4000

```
1 | SELECT * FROM teachers WHERE tid = 200010493
```

先查询原始工资：

tid	tname	email	salary
200010493	xzgkflko	zcf_m@def.com	3832

然后更改工资：

```
1 | UPDATE teachers SET salary = 4000 WHERE tid = 200010493;
```

结果如下：

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

4. 将TEACHERS表中所有工资小于2500的老师工资改为2500

先查询原始小于2500工资的老师：

```
1 | SELECT * FROM teachers WHERE salary < 2500;
```

部分结果如下：

tid	tname	email	salary
200005322	udjom	gd5l8@ppmlf.gov	1373
200050683	bmwdtty	uw_6@ozpe.edu	1455
200066763	btiagkw	dxd09@vxumi.net	1436
200120595	pejyw	zu9ee@zrjls.org	1486
200132776	rdhzntcl	kug3q@fyot.net	1421
200144197	uvcoff	gni8t7@zag.org	2064
200164470	pclrbeuwx	dn5b@czr.org	1459
200180180	nibggfks	tr9ej6_@ifmdd.org	2020
200207128	utydr	1zt3@tkhna.com	2429
200213467	mhlmtbix	yau6go@ngt.org	1953

然后更改他们的工资：

```
1 | UPDATE teachers SET salary = 2500 WHERE salary < 2500
```

然后查询工资小于2500的老师信息，查询结果为空，说明更改成功：

tid	tname	email	salary
▶ (N/A)	(N/A)	(N/A)	(N/A)

5. 将由编号为“200016731”的老师讲授的课程全部改成由姓名为“rnupx”的老师讲授

首先查询编号为200016731老师讲授的课程：

```
1 SELECT *
2 FROM choices
3 WHERE tid = 200016731
```

部分结果如下：

no	sid	tid	cid	score
500248263	817673762	200016731	10046	(Null)
▶ 500977362	891471910	200016731	10021	55
501751499	838617230	200016731	10048	62
502838711	851290158	200016731	10049	67
530103400	813737671	200016731	10019	95
533668489	860975663	200016731	10008	90
534235172	854985820	200016731	10026	74
534705108	889135469	200016731	10029	74
541652845	807064377	200016731	10027	84
547771169	873617908	200016731	10037	77
553313919	807673623	200016731	10005	81
553782058	867178237	200016731	10043	97

然后修改所有课程信息，将其改为rnupx老师教授

```
1 UPDATE choices SET tid = (SELECT tid FROM teachers WHERE tname = "rnupx")
  WHERE tid = 200016731
```

再查询由tid为200016731的老师教授的课程信息时，查询结果为空：

no	sid	tid	cid	score
▶ (N/A)	(N/A)	(N/A)	(N/A)	(N/A)

6. 更新编号“800071780”的学生年级为“2001”

原始状态为：

▶ 800071780	iscmwink	htoactp@aof.gov	2003
-------------	----------	-----------------	------

修改如下：

```
1 UPDATE students SET grade = 2001 WHERE sid = '800071780'
```

结果如下：

800071412	yiigoavp	tui1d7@jbadq.gov	1994
▶ 800071780	iscmwink	htoactp@aof.gov	2001

7. 删除没有学生选修的课程

查询没有学生选修的课程号：

```

1 SELECT cid
2 FROM(
3 SELECT cid,COUNT(sid) as COUNT
4 FROM choices
5 GROUP BY cid) as c
6 WHERE COUNT<=0

```

得到结果为空值，说明所有课程都有学生选。

cid
▶ (N/A)

删除语句为：

```

1 DELETE FROM courses WHERE cid in
2 (SELECT cid
3 FROM(
4 SELECT cid,COUNT(sid) as COUNT
5 FROM choices
6 GROUP BY cid) as c
7 WHERE COUNT<=0)

```

受影响的行数为0

8. 删除年级高于1998的学生信息

查询年级高于1998的学生信息：

```

1 SELECT *
2 FROM students
3 WHERE grade > 1998

```

执行删除语句：

```

1 DELETE FROM students WHERE grade>1998

```

报错，因为有外键约束，不能直接删除。

将含有外键约束的数据一同删除解决该问题：

```

1 DELETE FROM choices WHERE sid = ANY(SELECT sid
2 FROM students
3 WHERE grade > 1998)
4 DELETE FROM students WHERE grade>1998

```

最后查询年级>1998的学生数据时，返回为空值。

sid	sname	email	grade
(N/A)	(N/A)	(N/A)	(N/A)

9. 删除没有选修课程的学生信息

查询没有选修课程的学生信息：

```

1 SELECT *
2 FROM students
3 WHERE sid NOT in (SELECT sid
4 FROM choices)

```

结果如下：

sid	sname	email	grade
800022222	WangLan	(Null)	(Null)
AAA	user1	aa	1

删除如下：

```

1 DELETE FROM students WHERE sid not in (SELECT sid FROM choices)

```

再次查询没有选修课程的学生信息时，结果为空。

sid	sname	email	grade
(N/A)	(N/A)	(N/A)	(N/A)

10. 删除成绩不及格的选课记录

查询成绩不及格的选课记录如下：

```

1 SELECT *
2 FROM choices
3 WHERE score<60

```

部分结果如下：

no	sid	tid	cid	score
500015294	888484354	276307538	10050	54
500018789	832640245	239055611	10018	50
500021593	819911256	253207637	10041	50
500038935	841789759	223939189	10017	50
500040843	860579754	286542488	10020	53
500042536	829207334	255658245	10025	54
500056030	819331388	274499985	10022	51
500056347	844280874	292124905	10017	52

删除语句如下：

```
1 | DELETE FROM choices WHERE score < 60
```

删除后，再次查询成绩不及格的选课记录，结果为空：

no	sid	tid	cid	score
▶	(N/A)	(N/A)	(N/A)	(N/A)