

## 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 的有关视图的操作。
2. 能够使用 SQL 语句创建需要的视图，对视图进行查询和取消视图。

## 二、实验内容

1. 定义常见的视图形式，包括：
  - 行列子集视图
  - WITH CHECK OPTION 的视图
  - 基于多个基表的视图
  - 基于视图的视图
  - 带表达式的视图
  - 分组视图
2. 考察 WITH CHECK OPTION 这一语句在视图定义后产生的影响，包括对修改操作、删除操作、插入操作的影响
3. 讨论视图的数据更新情况，对子行列视图进行数据更新。
4. 使用 DROP 语句删除一个视图，由该视图导出的其他视图定义仍在数据字典中，但已不能使用，必须显式删除。同样的原因，删除基表时，由该基表导出的所有视图定义都必须显式删除。

## 三、实验过程及结果

- (1) 定义选课信息和课程名称的视图 VIEWC。

Sql 语句：

```
CREATE VIEW VIEWC
AS SELECT NO,SID,CHOICES.cid,TID,SCORE,CNAME
from CHOICES,COURSES
WHERE CHOICES.cid=COURSES.cid
SELECT * FROM VIEWC
```

查询结果：



	NO	SID	cid	TID	SCORE	CNAME
1	500000058	823063829	10037	249596497	76	software testing
2	500000253	829348273	10010	202560416	87	software engineering
3	500000984	850955252	10021	234145610	54	j2me
4	500001270	847061074	10025	292043491	92	embed system
5	500002143	860635914	10039	238811498	82	fortran
6	500002371	829785562	10028	273189968	77	architectonics
7	500002715	822137137	10011	218922066	67	distributed computing
8	500003262	826310502	10005	267846042	90	c++
9	500004144	817636568	10047	253205179	60	computer interface
10	500005519	813520169	10023	226385492	NULL	corba
11	500005935	801967882	10021	234419511	70	j2me

(2) 定义学生姓名与选课信息的视图 VIEWS。

Sql 语句:

```
CREATE VIEW VIEWS
AS SELECT STUDENTS.sname,no,CHOICES.sid,cid,tid,score
from CHOICES,STUDENTS
WHERE CHOICES.sid=STUDENTS.sid
SELECT *FROM VIEWS
```

查询结果如图:

	sname	no	sid	cid	tid	score
1	cqkjkuf	500003262	826310502	10005	267846042	90
2	wzinemrs	500004144	817636568	10047	253205179	60
3	rcypjhsnc	500005519	813520169	10023	226385492	NULL
4	bzbogqep	500007637	833961570	10004	209860626	80
5	mamqnc	500007901	894256303	10018	209831209	75
6	nrxfdwz	500011064	837501641	10029	290991539	NULL
7	mqpawdcm	500015544	893821981	10015	287021451	84
8	rkqwldddj	500017103	877343595	10009	268944598	94
9	fomgfdthz	500018971	879679311	10029	216020145	62
10	hxxvbriat	500033594	803918060	10037	261402689	88
11	bciskmgn	500036969	814238027	10024	263326045	98

(3) 定义年级低于 1998 的学生的视图 S1(SID, SNAME, GRADE)。

Sql 语句:

```
CREATE VIEW S1(sid,sname,grade)
AS SELECT sid,sname,grade
from STUDENTS
WHERE grade > 1998
WITH CHECK OPTION
SELECT *FROM S1
```

查询结果如图:

	sid	sname	grade
1	800002933	vnbqzsvv	2002
2	800008565	ehlycg	1999
3	800009026	rcxaihj	2002
4	800013889	nahhluoe	2000
5	800015244	egwwnf	1999
6	800023487	tkbzqduq	1999
7	800028044	ztozk	2001
8	800029781	kkivmiw	2000
9	800031326	froly	1999
10	800031798	oenbdg	2000
11	800032383	msnubuki	2004

REEDOM-COMPUTE(freedo... | School | 00:00:00 | 40092 行

(4) 查询学生为 “uxjof” 的学生的选课信息。

Sql: 语句:



```
SELECT *  
FROM VIEWS  
WHERE sname='uxjof'
```

查询结果如下:

	sname	no	sid	cid	tid	score
1	uxjof	506978093	800023963	10046	220667042	94
2	uxjof	541221076	800023963	10018	238341990	84
3	uxjof	567316431	800023963	10037	258375444	98

(5) 查询选修课程“UML”的学生的编号和成绩。

Sql 语句:

```
SELECT sid,score  
FROM VIEWC  
WHERE cname='UML'
```

查询结果如下:

	sid	score
1	848035070	88
2	897664264	62
3	898453203	91
4	846108663	86
5	827984677	86
6	823352185	NULL
7	882778410	69
8	884993242	74
9	806427512	96
10	848803637	85
11	839666024	92

2.0 RTM) FREEDOM-COMPUTE(freedo... School 00:00:00 5965 行

(6) 向视图 S1 插入记录(60000001,Lily,2001)。

Sql 语句:

```
INSERT  
INTO S1  
VALUES (60000001,'Lily',2001)  
SELECT * FROM S1  
WHERE sname='Lily'
```

查询结果如下:

	sid	sname	grade
1	60000001	Lily	2001

(7) 定义包括更新和插入约束的视图 S1,尝试向视图插入记录(60000001,Lily,1997),删除所有年级为 1999 的学生记录,讨论更新和插入约束带来的影响。

之前已经创建了含 WITH CHECK OPTION 的 S1 视图,先删除原先的 Lily 记录,再进行下面的操作。

Sql:

```
INSERT  
INTO S1  
VALUES (60000001,'Lily',1997)
```



结果:

消息 550, 级别 16, 状态 1, 第 1 行  
试图进行的插入或更新已失败, 原因是目标视图或者目标视图所跨越的某一视图指定了 WITH CHECK OPTION, 而该操作的一个或多个结果行又不符合 CHECK OPTION 约束。  
语句已终止。

WITH CHECK OPTION 会对插入或更新操作进行检查。由于插入的数据不符合视图定义  $grade > 1998$ , 所以数据插入失败。

Sql:

```
DELETE
FROM S1
WHERE grade=1999
```

结果:

消息 547, 级别 16, 状态 0, 第 1 行  
DELETE 语句与 REFERENCE 约束 'FK\_CHOICES\_STUDENTS' 冲突。该冲突发生于数据库 'School', 表 'dbo.CHICES', column 'sid'。  
语句已终止。

在视图中删除学生信息会在 STUDENTS 表中一起删掉, 而我由于 CHOICES 表中含有指向 STUDENTS 表的外键, 所以删除操作失败。

(8) 在视图 VIEWS 中将姓名为 “uxjof” 的学生的选课成绩都加上 5 分。

Sql 语句:

```
UPDATE VIEWS
SET score=score+5
WHERE sname='uxjof'

SELECT *
FROM VIEWS
WHERE sname='uxjof'
```

查询结果如下:

	sname	no	sid	cid	tid	score
1	uxjof	567316431	800023963	10037	258375444	103
2	uxjof	506978093	800023963	10046	220667042	99
3	uxjof	541221076	800023963	10018	238341990	89

原分数在第 4 题, 对比可知分数都增加了 5 分。

(9) 取消以上建立的所有视图。

Sql 语句如下:

```
EXECUTE
DROP VIEW VIEWC
DROP VIEW VIEWS
DROP VIEW S1
```

SELECT \* FROM VIEWC

消息 208, 级别 16, 状态 1, 第 1 行  
对象名 'VIEWC' 无效。

SELECT \* FROM VIEWS

消息 208, 级别 16, 状态 1, 第 1 行  
对象名 'VIEWS' 无效。

SELECT \* FROM S1

消息 208, 级别 16, 状态 1, 第 1 行  
对象名 'S1' 无效。



## 四、 实验感想

这次实验是熟悉 sql 语句中的跟视图操作相关的部分。创建视图优势在于其实就是把本来要联合几个表的数据放在一张表里，后面只要直接对这张表查询就可以，这样如果查询次数多的话，sql 语句书写也很简单，可以节省不少时间。