



## 3.5 视图

---

### 👉 本节内容:

3.5.1 什么是视图(View)?

3.5.2 视图定义

3.5.3 删除视图

3.5.4 查询视图

3.5.5 更新视图

3.5.6 视图的作用



## 3.5.1 什么是视图(View)?

- \* 视图是从一个或几个基本表（或视图）导出的表，它与基本表不同，是一个虚表。
- \* 在数据字典中只存放视图的定义，不会出现数据冗余。
- \* 基表中的数据发生变化，从视图中查询出的数据也随之改变。
- \* 视图一经定义，就可以和基本表一样被查询和删除，并且可以在视图之上再定义新的视图。
- \* 视图的更新(增加、删除、修改)操作会受到一定的限制。
- \* 视图对应三级模式体系结构中的**外模式**。



## 3.5.2 视图定义

视图定义语法:

```
CREATE VIEW <视图名> [( <列名> [, <列名>] ... )]  
AS <子查询>  
[WITH CHECK OPTION];
```

注:

- \* CREATE VIEW 子句中的列名可以省略, 此时视图的属性由子查询中SELECT目标列中的诸字段组成。但在下列情况下明确指定视图的所有列名:
  - (1) 某个目标列是集函数或列表表达式
  - (2) 多表连接时选出了几个同名列作为视图的字段
  - (3) 需要在视图中为某个列启用新的更合适的名字
- \* 子查询中的属性列不允许定义别名, 不允许含有ORDER BY子句和DISTINCT短语。
- \* WITH CHECK OPTION表示对视图进行更新操作的数据必须满足视图定义的谓词条件(子查询的条件表达式)。



## 3.5.2 视图定义

DBMS执行CREATE VIEW语句时只是把视图的定义存入数据字典，并不执行其中的SELECT语句。在对视图进行查询时，DBMS按视图的定义从基本表中将数据查出(执行视图定义中的SELECT语句)。

[例1] 建立信息系学生的视图。

```
CREATE VIEW IS_Student
AS
SELECT Sno, Sname, Sage
FROM Student
WHERE Sdept= 'IS' ;
```

- ▶ 若一个视图是从单个基本表导出的，并且只是去掉了基本表的某些行和某些列，但保留了码，这类视图称为行列子集视图。
- ▶ 上例所建立的视图即为行列子集视图。



## 3.5.2 视图定义

[例2] 建立信息系学生的视图，并要求透过该视图进行的更新操作只涉及信息系学生。

```
CREATE VIEW IS_Student
AS
SELECT Sno, Sname, Sage
FROM Student
WHERE Sdept= 'IS'
WITH CHECK OPTION ;
```

针对此视图，当进行以下操作时，

- 修改操作：DBMS自动加上Sdept= 'IS'的条件
- 删除操作：DBMS自动加上Sdept= 'IS'的条件
- 插入操作：DBMS自动检查Sdept属性值是否为'IS'
  - (1) 如果不是，则拒绝该插入操作
  - (2) 如果没有提供Sdept属性值，则自动定义Sdept为'IS'



## 3.5.2 视图定义

[例3] 建立信息系选修了1号课程的学生视图。

```
CREATE VIEW IS_S1( Sno, Sname, Grade )
AS
SELECT Student.Sno, Sname, Grade
FROM Student, SC
WHERE Sdept= 'IS' AND
      Student.Sno=SC.Sno AND
      SC.Cno= '1' ;
```

[例4] 建立信息系选修1号课程且成绩在90分以上的学生的视图。

```
CREATE VIEW IS_S2
AS
SELECT Sno, Sname, Grade
FROM IS_S1
WHERE Grade>=90;
```



## 3.5.2 视图定义

[例5] 定义一个反映学生出生年份的视图。

```
CREATE VIEW BT_S(Sno, Sname, Sbirth)
AS
SELECT Sno, Sname, 2007-Sage
FROM Student ;
```

注：当子查询中有由表达式构成的派生属性列(也称为虚拟列)时，必须明确定义组成视图的各个属性列名。

[例6] 将学生的学号及他的平均成绩定义为一个视图

```
CREATE VIEW S_G (Sno, Gavg)
AS
SELECT Sno, AVG (Grade)
FROM SC
GROUP BY Sno ;
```



## 3.5.2 视图定义

[例7] 将Student表中所有女生记录定义为一个视图

```
CREATE VIEW F_Student1 (stdnum, name, sex, age, dept)
AS SELECT *
FROM Student
WHERE Ssex='女' ;
```

潜在问题:

修改基表Student的结构后(在非末尾增加一列), Student表与F\_Student1视图的映象关系会被破坏, 导致该视图不能正常工作。

处理方法:

在子查询的SELECT子句中明确指出各属性列的名称, 可以避免对基本表的属性列增加而破坏与视图间的映象关系, 但不能解决修改列名的问题, 因此上对基本表修改后采用重建视图的方法。