数据库原理及应用

Principle and Application of Database

华东交通大学
喻佳



3.4

关系数据库标准语言SQL

------ 单表查询（续）

复习与回顾

查询语句的一般格式：

SELECT [ALL|DISTINCT] <目标列表达式> [, <目标列表达式>] …

FROM <表名或视图名> [, <表名或视图名>] …

[WHERE <条件表达式>]

[GROUP BY <列名1> [HAVING <条件表达式>]]

[ORDER BY <列名2> [ASC|DESC]];

三、使用聚集函数的查询

五类主要的聚集函数：

函数名称	功能
COUNT（[DISTINCT|ALL] *） COUNT（[DISTINCT|ALL] <列名>）	元组计数 按列值元组计数
SUM（[DISTINCT|ALL] <列名>）	按列计算值得总和
AVG（[DISTINCT|ALL] <列名>）	按列计算平均值
MAX（[DISTINCT|ALL] <列名>）	求一列中的最大值
MIN（[DISTINCT|ALL] <列名>）	求一列中的最小值

三、使用聚集函数的查询

例1：查询学生总人数。

```
SELECT  Count ( Sno )
FROM    Student;
```

亦可写成： Count (*)

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
201815121	李勇	男	20	CS
201815122	刘晨	女	19	CS
201815123	王敏	女	18	MA
201815125	张立	男	19	IS

三、使用聚集函数的查询

例1：查询学生总人数。

```
SELECT  Count ( Sno )
FROM    Student;
```

亦可写成： Count (*)

例2：查询选修了课程的学生人数。

```
SELECT  Count ( DISTINCT  Sno )
FROM    SC;
```

避免重复计算学生人数

学号 Sno	课程号 Cno	成绩 Grade
201815121	1	92
201815121	2	85
201815121	3	88
201815122	2	90
201815122	3	80

三、使用聚集函数的查询

例3：查询学生201815122选修的课程的总分和平均分。

```
SELECT  SUM ( Grade ), AVG ( Grade )
FROM    SC
WHERE   Sno = '201815122' ;
```

例4：查询选修2号课程的最高分和最低分。

```
SELECT  MAX ( Grade ), MIN ( Grade )
FROM    SC
WHERE   Cno = '2';
```

学号 Sno	课程号 Cno	成绩 Grade
201815121	1	92
201815121	2	85
201815121	3	88
201815122	2	90
201815122	3	80

四、对查询结果分组

通过**GROUP BY**子句实现。

分组方法：按指定的一列或多列值分组，值相等的为一组。

SELECT子句的目标列表达式列表中只能出现**分组属性**和**聚集函数**。

GROUP BY子句细化了聚集函数的作用对象：

- 未对查询结果分组，聚集函数将作用于整个查询结果；
- 对查询结果分组后，聚集函数将分别作用于每个组。

四、对查询结果分组

例5：查询每个课程号及其选课人数。

```
SELECT  Cno, COUNT(Sno)
FROM    SC
GROUP   BY  Cno;
```

查询结果：

Cno	COUNT(Sno)
1	1
2	2
3	2

学号 Sno	课程号 Cno	成绩 Grade
201815121	1	92
201815121	2	85
201815121	3	88
201815122	2	90
201815122	3	80

学号 Sno	课程号 Cno	成绩 Grade
201815121	1	92
201815121	2	85
201815122	2	90
201815121	3	88
201815122	3	80

四、对查询结果分组

例5：查询每个课程号及其选课人数。

```
SELECT  Cno, COUNT(Sno)
FROM    SC
GROUP   BY  Cno;
```

查询结果：

Cno	COUNT(Sno)
1	1
2	2
3	2

例6：查询至少有2人选修的课程号及其选课人数。

```
SELECT  Cno, COUNT(Sno)
FROM    SC
GROUP   BY  Cno
HAVING   COUNT(Sno) >= 2;
```

对分组后的查询结果进行筛选，输出满足HAVING指定条件的组

四、对查询结果分组

HAVING与WHERE子句的区别：作用对象不同

- **WHERE作用于基本表或视图，从中选择满足条件的元组。**
- **HAVING作用于分组，从中选择满足条件的分组。**

五、对查询结果排序

通过**ORDER BY** 子句实现。

可以按一个或多个属性列排序。

升序：**ASC**；降序：**DESC**；缺省值为**ASC**。

当排序列含空值时：

ASC：排序列为空值的元组**最后**显示；

DESC：排序列为空值的元组**最先**显示。

五、对查询结果排序

例7：查询选修了3号课程的学生的学号及其成绩，查询结果按分数降序排列。

```
SELECT  Sno, Grade
FROM    SC
WHERE  Cno= '3'
ORDER  BY  Grade  DESC;
```

学号 Sno	课程号 Cno	成绩 Grade
201815121	1	92
201815121	2	85
201815121	3	88
201815122	2	90
201815122	3	80
201815123	3	

查询结果：

Sno	Grade
201815121	88
201815122	80

Sno	Grade
201815123	
201815121	88
201815122	80

五、对查询结果排序

例8：查询选修了1号、2号、或3号课程的学号、课程号和成绩，查询结果按学号升序排列，学号相同再按成绩降序排列。

```
SELECT  Sno, Cno, Grade
FROM    SC
WHERE   Cno IN ('1', '2', '3')
ORDER  BY Sno, Grade DESC;
```

学号 Sno	课程号 Cno	成绩 Grade
201815121	1	92
201815121	2	85
201815121	3	88
201815122	2	90
201815122	3	80

查询结果：

Sno	Cno	Grade
201815121	1	92
201815121	3	88
201815121	2	85
201815122	2	90
201815122	3	80

单表查询综合题

例9：查询有3门以上选修课程成绩超过90分的学生学号及超过90分的课程数，查询结果按课程数降序排列。

```
SELECT  Sno, COUNT(Cno)
FROM    SC
WHERE  Grade>=90
GROUP  BY Sno
HAVING  COUNT(Cno) >= 3;
ORDER  BY  COUNT(Cno) DESC;
```

谢　　谢

THANK YOU FOR YOUR WATCHING

