



UNIT 6 简单查询



本讲主要目标



学完本讲后，你应该能够了解：

- 1、SELECT语句中四个子句的功能和概念性执行步骤；
- 2、使用SELECT语句：
 - 使用表别名和列别名；
 - 查询满足一定条件的元组；
 - 查询某些属性的值；
 - 利用DISTINCT去掉查询结果中的重复行；
 - 利用GROUP BY进行分组统计
 - 利用ORDER对查询结果按要求排序。



本讲主要内容

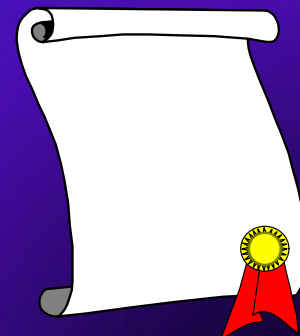
一. 查询语句概述

二. 单表查询

三. 排序

四. 集合函数

五. 分组





DreamHome 租赁数据库

DreamHome 案例的部分关系模式：

- ❁ **Branch** (branchNo, street, city, postcode)
- ❁ **Staff** (staffNo, fName, lName, position, sex, DOB, salary, branchNo)
- ❁ **PropertyForRent** (propertyNo, street, city, postcode, type, rooms, rent, ownerNo, staffNo, branchNo)
- ❁ **Client** (clientNo, fName, lName, telNo, prefType, maxRent)
- ❁ **PrivateOwner** (ownerNo, fName, lName, address, telNo)
- ❁ **Viewing** (clientNo, propertyNo, viewDate, comment)
- ❁ **Registration** (clientNo, branchNo, staffNo, dateJoined)



DreamHome租赁数据库实例:

Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005



PropertyForRent

propertyNo	street	city	postcode	type	rooms	rent	ownerNo	staffNo	branchNo
PA14	16 Holhead	Aberdeen	AB7 8SU	House	6	650	CO46	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	CO87	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	CO40		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	CO93	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	CO87	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	CO93	SG14	B003

Client

clientNo	fName	lName	telNo	prefType	maxRent	eMail
CR76	John	Kay	0207-774-5632	Flat	425	john.kay@gmail.com
CR56	Aline	Stewart	0141-848-1825	Flat	350	astewart@hotmail.com
CR74	Mike	Ritchie	01475-392178	House	750	mritchie01@yahoo.co.uk
CR62	Mary	Tregear	01224-196720	Flat	600	maryt@hotmail.co.uk



PrivateOwner

ownerNo	fName	lName	address	telNo	eMail	password
CO46	Joe	Keogh	2 Fergus Dr, Aberdeen AB2 7SX	01224-861212	jkeogh@lhh.com	*****
CO87	Carol	Farrel	6 Achray St, Glasgow G32 9DX	0141-357-7419	cfarrel@gmail.com	*****
CO40	Tina	Murphy	63 Well St, Glasgow G42	0141-943-1728	tinam@hotmail.com	*****
CO93	Tony	Shaw	12 Park Pl, Glasgow G4 0QR	0141-225-7025	tony.shaw@ark.com	*****

Viewing

clientNo	propertyNo	viewDate	comment
CR56	PA14	24-May-13	too small
CR76	PG4	20-Apr-13	too remote
CR56	PG4	26-May-13	
CR62	PA14	14-May-13	no dining room
CR56	PG36	28-Apr-13	

Registration

clientNo	branchNo	staffNo	dateJoined
CR76	B005	SL41	2-Jan-13
CR56	B003	SG37	11-Apr-12
CR74	B003	SG37	16-Nov-11
CR62	B007	SA9	7-Mar-12



一个学生-课程数据库

S

学号 S#	姓名 SN	性别 SE	年龄 SA	所在系 SD
95001	李勇	男	20	CS
95002	刘晨	女	19	IS
95003	王敏	女	18	MA
95004	张立	男	19	IS

SC

学号 S#	课程号 C#	成绩 G
95001	C1	92
95001	C2	85
95001	C3	88
95002	C2	90
95002	C3	80

C

课程号 C#	课程名 CN	先行课 CP#	学分 CC
C1	数据库	C5	4
C2	数学		2
C3	信息系统	C1	4
C4	操作系统	C6	3
C5	数据结构	C7	4
C6	数据处理		2
C7	PASCAL语言	C6	4



一、查询语句概述

1、基本语法（见教材P73-74）

SELECT [ALL | DISTINCT]

{ * | [columnExpression [AS newName]] [, ...] }

FROM TableName [alias] [, ...]

[**WHERE** condition]

[**GROUP BY** columnList [**HAVING** condition]]

[**ORDER BY** columnList]



一、查询语句概述

定义表语句的基本语法：

SELECT [ALL|DISTINCT] 〈目标列表表达式〉 [, 〈目标列表表达式〉] ...

FROM 〈表名或视图名〉 [, 〈表名或视图名〉] ...

[WHERE 〈条件表达式〉]

[GROUP BY 〈列名〉 [, 〈列名〉]...

[HAVING 〈内部函数表达式〉]]

[ORDER BY 〈列名〉 [ASC | DESC] [, 〈列名〉 [ASC | DESC]]...]]



一、查询语句概述

2、子句功能

- ◆ SELECT子句与FROM子句是必选子句；
- ◆ SELECT ---- 列出查询的结果；
- ◆ FROM ---- 指明所访问的表；
- ◆ WHERE ---- 查询的条件，条件可以有多个；
- ◆ GROUP BY ---- 将查询结果按指定字段的取值分组；
- ◆ HAVING ---- 每组应满足的条件；
- ◆ ORDER BY ---- 按指定的字段的值，以升序或降序排列查询结果



一、查询语句概述

3、整个SELECT的含义

- ◆ 根据WHERE子句中的条件表达式，从FROM子句中的基本表或视图中找出满足条件的元组；
- ◆ 按SELECT子句中的目标字段，选出元组中的分量形成结果表；
- ◆ GROUP BY子句将结果按字段分组，每个组产生结果表中的一个元组；
- ◆ 通常在每组中作用库函数，分组的附加条件用HAVING短语给出；只有满足内部函数表达式的组才予输出；
- ◆ 如果有ORDER BY子句，则结果表要根据指定的字段按升序或降序排列。



一、查询语句概述

4、子句的概念性的执行步骤

- ① 计算出现在FROM后面的表之间的笛卡儿积
- ② 实施由WHERE后面的条件规定的选择操作
- ③ 根据GROUP BY后面的属性的值进行分组
- ④ 对于分组出来的每一个小组，实施由HAVING后面的条件规定的选择操作，保留满足条件的小组
- ⑤ 将出现在SELECT后面的属性投影成结果表
- ⑥ 将结果表中的元组根据ORDER BY后面的属性及指定的顺序（升序或降序）输出

概念性执行步骤并非真正的执行步骤！



本讲主要内容

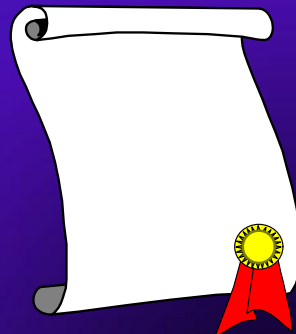
一. 查询语句概述

二. 单表查询

三. 排序

四. 集合函数

五. 分组





二、单表查询

1、单表查询

—— 从单张表中查询出结果的查询

完成关系代数中的两个单目运算：

- 投影
- 选择

可以使用的SELECT语句的语法：

```
SELECT [ALL | DISTINCT]
    { * | [columnExpression [AS newName]] [, ...] }
FROM    TableName [alias] [, ...]
[WHERE condition]
```



二、单表查询

2、投影（SELECT子句的〈目标列表表达式〉）

目标表达式可以是关系名、属性名、算术表达式、字符串常量、函数等

➤ 检索所有的列和所有的行

例6.1 列举所有职员的信息

```
SELECT staffNo, fName, lName, address,  
       position, sex, DOB, salary, branchNo  
FROM Staff;
```

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000.00	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000.00	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000.00	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000.00	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000.00	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000.00	B005

本例中目标列表表达式可用星号 ‘*’ 代替所有列

```
SELECT *  
FROM Staff;
```



二、单表查询

- 从所有行中检索指定的列

例6.2 生成所有职员的工资表，只包括职员编号、姓名及工资

```
SELECT staffNo, fName, lName, salary
```

```
FROM Staff;
```

staffNo	fName	lName	salary
SL21	John	White	30000.00
SG37	Ann	Beech	12000.00
SG14	David	Ford	18000.00
SA9	Mary	Howe	9000.00
SG5	Susan	Brand	24000.00
SL41	Julie	Lee	9000.00



二、单表查询

➤ 使用ALL|DISTINCT

例6.3 列出被查看过的所有房产的编号

```
SELECT ALL propertyNo  
FROM Viewing;
```

propertyNo
PA14
PG4
PG4
PA14
PG36

*用保留字DISTINCT消除结果中的重复行

```
SELECT DISTINCT propertyNo  
FROM Viewing;
```

propertyNo
PA14
PG4
PG36

- ALL是缺省值
- DISTINCT的使用导致系统性能的下降，如果能确认结果行中没有重复行，就避免使用DISTINCT



二、单表查询

➤ 计算列 (+, -, *, /, %)

例6.4 生成所有职员按月工资表，只包括职员编号、姓名及工资

```
SELECT staffNo, fName, lName, salary/12  
FROM Staff;
```

⊗ 用AS子句为列命名：

```
SELECT staffNo, fName, lName, salary/12 AS monthlySalary  
FROM Staff;
```

staffNo	fName	lName	col4
SL21	John	White	2500.00
SG37	Ann	Beech	1000.00
SG14	David	Ford	1500.00
SA9	Mary	Howe	750.00
SG5	Susan	Brand	2000.00
SL41	Julie	Lee	750.00



二、单表查询

3、选择（WHERE子句的<条件表达式>）（见教材P77）

- 比较运算作为查找条件（>, <, >=, <=, =, <>）

例6.5 查询工资高于10000英镑的所有职员

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE salary > 10000;
```

staffNo	fName	lName	position	salary
SL21	John	White	Manager	30000.00
SG37	Ann	Beech	Assistant	12000.00
SG14	David	Ford	Supervisor	18000.00
SG5	Susan	Brand	Manager	24000.00



二、单表查询

- 复合比较运算作为查找条件 (AND, OR, NOT)

例6.6 列出位于London或Glasgow的所有分支机构的地址

```
SELECT *  
FROM   Branch  
WHERE  city = 'London' OR city = 'Glasgow' ;
```

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B003	163 Main St	Glasgow	G11 9QX
B002	56 Clover Dr	London	NW10 6EU



二、单表查询

➤ 范围作为查找条件 (BETWEEN/NOT BETWEEN)

例6.7 列出工资在20000和30000英镑之间的所有职员

```
SELECT staffNo, fName, lName, position, salary
FROM      Staff
WHERE     salary BETWEEN 20000 AND 30000;
```

说明：◇BETWEEN 测试包含范围的端点.

◇NOT BETWEEN测试范围以外的所有值

◇BETWEEN并不能增强SQL的功能。

上面的SQL语句可写为：

```
SELECT staffNo, fName, lName, position, salary
FROM      Staff
WHERE     salary>=20000 AND salary <= 30000;
```



二、单表查询

- 集合成员测试作为查找条件 (IN/NOT IN)

例6.8 列出所有的经理与主管

```
SELECT staffNo, fName, lName, position
FROM Staff
WHERE position IN ('Manager', 'Supervisor');
```

- 说明：
- ◇ IN 测试数据是否与集合中的某一值相匹配
 - ◇ NOT IN 测试测试数据是否不在集合中
 - ◇ IN 并不能增强SQL的功能, 但可以有效表达条件运算。

上面的SQL语句可写为：

```
SELECT staffNo, fName, lName, position
FROM Staff
WHERE position='Manager' OR position='Supervisor';
```



二、单表查询

➤ 模式匹配作为查找条件 (LIKE/NOT LIKE)

例6.9 找出其地址中含有字符串 ‘Glasgow’ 的所有房主

◆ SQL中有两种特殊的模式匹配符号 (见教材P79)

- %: 百分号表示零或多个字符序列 (通配符)
- _: 下划线表示任意单个字符

```
SELECT ownerNo, fName, lName, address, telNo  
FROM   PrivateOwner  
WHERE  address LIKE '%Glasgow%';
```

◆ 如果查找的字符串本身包含上述模式匹配符呢？



二、单表查询

➤ 模式匹配中的转义符ESCAPE (LIKE/NOT LIKE)

如果查找的字符串本身包含上述模式匹配符，可用转义字符

当模式包含一个转义符时，紧随其后的字符按它的字面含义来使用

使用ESCAPE子句为每个SELECT语句定义不同的转义符。

[NOT] LIKE 字符串 [ESCAPE 转义符]

例如 查询学号的第二个字符不是“%”的学生信息

```
SELECT *  
FROM S  
WHERE S# NOT LIKE ‘_\\%%’ ESCAPE ‘\\’ ;
```



二、单表查询

➤ 空查找条件 (IS NULL/IS NOT NULL)

例6.10 列出查看过编号为PG4的房产但没有留下意见的客户信息

```
SELECT clientNo, viewDate  
FROM   Viewing  
WHERE  propertyNo = 'PG4' AND  
       comment IS NULL;
```

◆ IS NOT NULL 用于测试非空值的情况



二、单表查询

总结：WHERE子句常用的查询条件 (P77-78)

查 询 条 件	谓 词
比 较	=, >, <, >=, <=, !=, <>, !>, !<; NOT+上述比较运算符
确定范围	BETWEEN AND, NOT BETWEEN AND
确定集合	IN, NOT IN
字符匹配	LIKE, NOT LIKE
空 值	IS NULL, IS NOT NULL
多重条件（逻辑运算）	AND, OR, NOT



本讲主要内容

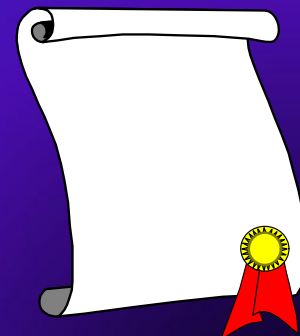
一. 查询语句概述

二. 单表查询

三. 排序

四. 集合函数

五. 分组





三、排序

1、ORDER BY 子句

语法：

[ORDER BY columnName [ASC | DESC] [, ...]]

功能说明：

➤ ORDER BY子句让查询结果按照一定顺序显示

ASC ---- 升序（缺省）

DESC ---- 降序

➤ ORDER BY子句必须是SELECT语句的最后一个子句



三、排序

2、单列排序

例6.11 按工资降序的方式产生所有职工的工资列表

```
SELECT  staffNo, fName, lName, salary
FROM    Staff
ORDER BY salary DESC;
```

❁ 或者 使用SELECT列表中相应列的序号

```
SELECT  staffNo, fName, lName, salary
FROM    Staff
ORDER BY 4 DESC;
```



三、排序

3、多列排序

例6.12 产生按类型和租金排序的一个房产简表

```
SELECT  propertyNo, type, rooms, rent
FROM    PropertyForRent
ORDER BY type, rent DESC;
```

propertyNo	type	rooms	rent
PG16	Flat	4	450
PL94	Flat	4	400
PG36	Flat	3	375
PG4	Flat	3	350
PA14	House	6	650
PG21	House	5	600



本讲主要内容

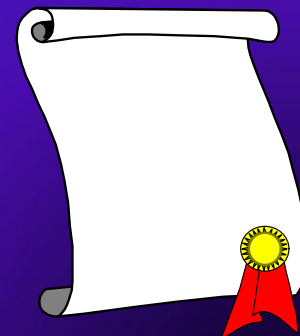
一. 查询语句概述

二. 单表查询

三. 排序

四. 集合函数

五. 分组





四、集合函数

1、ISO标准定义了五个集合函数（见教材P82）

名称	参数类型	结果类型	描述
COUNT	任意（可以是*）	数值型	列中数据计数
SUM	数值型	数值型	列中数据总和
AVG	数值型	数值型	列中数据平均值
MAX	字符型或数值型	同参数类型一样	列中数据最大值
MIN	字符型或数值型	同参数类型一样	列中数据最小值



四、集合函数

集合函数使用说明：

- COUNT(*) 用于统计元组个数；
- COUNT(<列名>) 用于统计列中值的个数
- 除COUNT(*)外，这些函数只对表中单个列进行操作，返回一个值，都可以使用DISTINCT|ALL选项，默认ALL
 - 函数名 [DISTINCT|ALL] <列名>
- 当这些函数遇到空值时，除COUNT(*)外，都跳过空值而只处理非空值；
- 这些函数只能用于SELECT列表和HAVING子句中



四、集合函数

2、COUNT(*)的使用

例6.13 每月租金超过350英镑的房产有多少

```
SELECT COUNT(*) AS myCount
FROM   PropertyForRent
WHERE  rent > 350;
```

myCount

5

列别名

PropertyForRent

propertyNo	street	city	postcode	type	rooms	rent	ownerNo	staffNo	branchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	CO46	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	CO87	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	CO40		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	CO93	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	CO87	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	CO93	SG14	B003



四、集合函数

3、COUNT (DISTINCT) 的使用

例6. 14 2004年5月有多少处不同的房产被看过

```
SELECT COUNT(DISTINCT propertyNo) AS myCount
FROM Viewing
WHERE viewDate BETWEEN '1-May-04' AND '31-May-04' ;
```

Viewing			
clientNo	propertyNo	viewDate	comment
CR56	PA14	24-May-13	too small
CR76	PG4	20-Apr-13	too remote
CR56	PG4	26-May-13	
CR62	PA14	14-May-13	no dining room
CR56	PG36	28-Apr-13	

myCount
2



四、集合函数

4、COUNT和SUM的使用

例6.15 找出经理的总人数，并计算他们工资的总和

```
SELECT COUNT(staffNo) AS myCount,  
        SUM(salary) AS mySum  
FROM    Staff  
WHERE   position = 'Manager' ;
```

Staff							
staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

myCount	mySum
2	54000.00



四、集合函数

5、MIN、MAX和AVG的使用

例6.16 找出所有职员工资的最小、最大和平均值

```
SELECT MIN(salary) AS myMin,  
       MAX(salary) AS myMax,  
       AVG(salary) AS myAvg  
FROM   Staff;
```

Staff							
staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

myMin	myMax	myAvg
9000.00	30000.00	17000.00



本讲主要内容

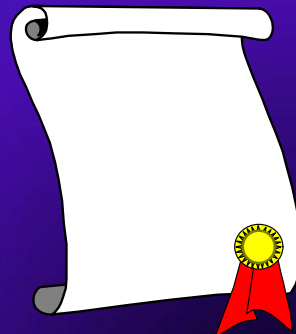
一. 查询语句概述

二. 单表查询

三. 排序

四. 集合函数

五. 分组





五、分组（见教材P83-84）

1、GROUP BY 子句的语法

➤ **语法：**

[**GROUP BY** columnName [, ...] [**HAVING** condition]]

➤ **功能：**

将查询结果按指定列的取值分组；并选取满足
HAVING中的内部函数表达式所指定条件的组；

➤ 当使用GROUP BY时， SELECT列表中的项必须**每组都有单一值**；

➤ 当使用GROUP BY时， SELECT子句仅可包含以下内容：

- 列名
- 集合函数
- 常量
- 组合上述各项的表达式



五、分组

- 当使用GROUP BY时，SELECT子句中除集合函数外，其它所有列必须在GROUP BY子句中出现；反之，GROUP BY子句出现的列不一定出现在SELECT列表中；
- 当WHERE子句和GROUP BY同时使用时，必须先使用WHERE子句，分组由满足WHERE子句的查询条件的那些行产生；
- ISO标准规定，应用GROUP BY时，两个空值被认为是相等的



五、分组

2、GROUP BY 的使用

例6.17 找出工作在每一个分支机构的职员人数和他们的工资总和。

```
SELECT    branchNo,  
          COUNT(staffNo) AS myCount,  
          SUM(salary) AS mySum  
FROM      Staff  
GROUP BY branchNo;
```

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

branchNo	myCount	mySum
B003	3	54000.00
B005	2	39000.00
B007	1	9000.00



五、分组

3、分组约束（HAVING子句）

- HAVING子句是与GROUP BY 子句一起使用，以约束哪些分组出现在最终查询结果中；
- WHERE子句将单个行“过滤”到查询结果中；而HAVING子句则将分组“过滤”到查询结果表中；
- ISO标准要求HAVING子句使用的列名必须出现在GROUP BY 子句列表中，或者包括在集合函数中
- 实践中，HAVING子句的条件表达式至少包括一个集合函数，否则，可把查询条件移到WHERE子句中来过滤单个行

注：HAVING子句是包含在GROUP BY子句中的。



五、分组

4、HAVING的使用

例6.18 对于职员人数多于1人的分支机构，计算出每一个分支机构的职员人数和他们的工资总和。

```
SELECT    branchNo,  
          COUNT(staffNo) AS myCount,  
          SUM(salary) AS mySum  
FROM      Staff  
GROUP BY  branchNo  
HAVING    COUNT(staffNo) > 1  
ORDER BY  branchNo;
```

branchNo	myCount	mySum
B003	3	54000.00
B005	2	39000.00



五、分组

例6.19 查询各科平均成绩大于等于90分的学生的学号和平均成绩。

```
SELECT  S#, AVG(G)
FROM    SC
GROUP BY S#
HAVING  AVG(G) >= 90;
```

```
SELECT  S#, AVG(G)
FROM    SC
WHERE   AVG(G) >= 90;
GROUP BY S#;
```

右侧的查询正确吗？为什么？



Questions?





本讲主要目标

学完本讲后，你应该能够了解：



- 1、SELECT语句中四个子句的功能和概念性执行步骤；
- 2、使用SELECT语句：
 - 使用表别名和列别名；
 - 查询满足一定条件的元组；
 - 查询某些属性的值；
 - 利用DISTINCT去掉查询结果中的重复行；
 - 利用GROUP BY进行分组统计
 - 利用ORDER对查询结果按要求排序。

问题讨论

- 1、比较SQL与关系代数的非过程化程度。
- 2、WHERE子句与HAVING子句的功能都是进行条件选择，它们的用法区别是什么？





作业

教材：《数据库系统原理教程》（第2版）

P112

◆ 3、

