

SQL 的查詢語言

查詢的基本語法

```
SELECT [* | DISTINCT | Top n] <欄位串列>  
FROM (資料表名稱{<別名>} | JOIN資料表名稱)  
[WHERE <條件式>]  
[GROUP BY <群組欄位> ]  
[HAVING <群組條件>]  
[ORDER BY <欄位> [ASC | DESC]]
```

【說明】

1. Select後面要接所要列出的欄位名稱。
2. [* | Distinct|Topn]中括號的部份可以省略。
 - (1) " *" 表示列印出所有的欄位(欄位1,欄位2,.....,欄位n)
 - (2) Distinct代表從資料表中選擇不重複的資料。
它是利用先排序來檢查是否有重複。
 - (3) Top n指在資料表中取出名次排序在前的n筆記錄。

查詢的基本語法 <續>

【說明】

3. From 後面接資料表名稱，它可以接一個以上的資料表。
4. Where後面要接條件式 (它包括了各種運算子)
5. Group By 欄位1,欄位2,...,欄位n [Having 條件式]
 - (1) Group By 可單獨存在，它是將數個欄位組合起來，以做為每次動作的依據。
 - (2) [Having 條件式]是將數個欄位中加以有條件的組合。它不可以單獨存在。
6. Order By 欄位1, 欄位2, ..., 欄位n [Asc|Desc]

它是依照某一個欄位來進行排序。

例如：(1) ORDER BY 成績 Asc ←可以省略(由小至大)
(2) ORDER BY成績 Desc ←不可以省略(由大至小)

使用Select子句

【定義】

Select是指在資料表中，選擇**全部**或**部份欄位**顯示出來，這就是所謂的「**投影運算**」。

【格式】

```
Select 欄位串列  
From 資料表名稱
```

- 查詢全部欄位

如果**沒有指定欄位**的話，我們可以直接利用**星號「*」**代表**所有的欄位名稱**。

【優點】**不需輸入全部**的欄位名稱。

[實例]

在「學生資料表」中顯示「所有學生基本資料」

【解答】

SQL指令1

```
SELECT *  
FROM 學生資料表
```

一、學生資料表

	學號	姓名	系碼
#1	S0001	張三	D001
#2	S0002	李四	D002
#3	S0003	王五	D003
#4	S0004	陳明	D001
#5	S0005	李安	D004

【查詢結果】



學號	姓名	系碼
S0001	張三	D001
S0002	李四	D002
S0003	王五	D003
S0004	陳明	D001
S0005	李安	D004

SQL指令2與SQL指令1有相同的結果

```
SELECT 學號,姓名,系碼  
FROM 學生資料表
```

查詢指定欄位(垂直篩選)

【定義】

由於上一種方法，只能直接選擇全部的欄位資料，無法顧及隱藏私人資料及自行調整欄位順序的問題，因此，我們利用指定欄位來查詢資料。

[實例]

在「學生資料表」中查詢所有學生的「姓名及系碼」

【解答】

SQL指令
SELECT 姓名, 系碼
FROM 學生資料表

欄位與欄位名稱之間，必須要以逗號「，」隔開

一、學生資料表

	學號	姓名	系碼
#1	S0001	張三	D001
#2	S0002	李四	D002
#3	S0003	王五	D003
#4	S0004	陳明	D001
#5	S0005	李安	D004

【查詢結果】

查詢1	姓名	系碼
	張三	D001
	李四	D002
	王五	D003
	陳明	D001
	李安	D004

說明：在「學生資料表」中將「姓名」及「系碼」投射出來。

使用「比較運算子」

如果我們所想要的資料是要符合某些條件，而不是全部的資料時，那就必須要在Select子句中再使用Where條件式即可。並且也可以配合使用「比較運算子」來搜尋資料。若條件式成立的話則會傳回「True(真)」，若不成立的話則會傳回「False(假)」。

SQL指令
Select 欄位集合
From 資料表名稱
Where 條件式

運算子	功能	例子	條件式說明
= (等於)	判斷A與B是否相等	A=B	成績=60
<> (不等於)	判斷A是否不等於B	A<>B	成績<>60
< (小於)	判斷A是否小於B	A<B	成績<60
<= (小於等於)	判斷A是否小於等於B	A<=B	成績<=60
> (大於)	判斷A是否大於B	A>B	成績>60
>= (大於等於)	判斷A是否大於等於B	A>=B	成績>=60

查詢滿足條件的值組(水平篩選)

【定義】

當我們所想要的資料是要符合某些條件，而不是全部的資料時，那就必須要在Select子句中再使用Where條件式即可。

【優點】 1.可以依照使用者的需求來查詢。

2.資訊較為集中。

[實例]

在「選課資料表」中查詢修課號為「C005」的學生的「學號及成績」

【解答】

SQL指令

```
SELECT 學號, 成績 FROM 選課資料表  
WHERE 課號='C005'
```

【查詢結果】

學號	成績
S0002	63
S0001	73
S0003	70
S0004	68
S0005	

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

[實例]

在「選課資料表」中查詢任何課程成績「不及格60」的學生的「學號、課程代號及成績」

【解答】

```
SQL指令
SELECT 學號, 課號, 成績
FROM 選課資料表
WHERE 成績 < 60
```

「60分」是數值資料
不須要加「左右單引

號」

【查詢結果】

學號	課號	成績
S0001	C001	56

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

使用「邏輯運算子」

在Where條件式中除了可以設定「比較運算子」之外，還可以設定「邏輯運算子」來將數個「比較運算子」條件組合起來，成為較複雜的條件式。

運算子	功能	條件式說明
And(且)	判斷A且B兩個條件式是否皆成立	成績 >= 60 And 課程代號 = 'C005'
Or(或)	判斷A或B兩個條件式是否有一個成立	課程代號 = 'C004' Or 課程代號 = 'C005'
Not(反)	非 A的條件式	Not 成績 >= 60

IS NULL(空值)

【定義】 NULL值是表示沒有任何的值(空值)，在一般的資料表中有些欄位中並沒有輸入任何的值。例如：學生月考缺考，使得該科目成績是空值。

【實例1】

在「選課資料表」中查詢那些學生「缺考」的「學號、課號及成績」。

【解答】

```
SQL指令
SELECT 學號, 課號, 成績
FROM 選課資料表
WHERE 成績 IS NULL
```

設定IS NULL條件，其回傳的值True或False

【查詢結果】



學號	課號	成績
S0005	C005	

注意：這裡的「IS」不能用等號(=)代替它。

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

[實例2]

在「選課資料表」中查詢那些學生「沒有缺考」的「學號、課號及成績」

【解答】

SQL指令

```
SELECT 學號, 課號, 成績  
FROM 選課資料表  
WHERE 成績 IS NOT NULL
```

設定 IS NOT NULL 條件

09_IS NOT NULL(空值)			
學號	課號	成績	
S0001	C001	56	
S0002	C005	63	
S0003	C004	92	
S0001	C005	73	
S0002	C002	92	
S0003	C005	70	
S0004	C003	75	
S0004	C004	88	
S0004	C005	68	

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

使用「模糊條件與範圍」

【定義】

在Where條件式中除了可以設定「**比較運算子**」與「**邏輯運算子**」之外，還可以設定「**模糊或範圍條件**」來查詢。

【例如】奇摩的搜尋網站，使用者只要輸入**某些關鍵字**，就可以即時查詢出相關的資料。其常用的**模糊或範圍運算子**如表所示：

運算子	功能	條件式
1. Like	模糊相似條件	Where 系所 LIKE '資管%'
2. IN	集合條件	Where課程代號IN('C001','C002')
3. Between.....And	範圍條件	Where成績 Between 60 And 80

Like模糊相似條件

【定義】

LIKE運算子利用萬用字元(%及_)來比較相同的內容值。

- (1)萬用字元(%)百分比符號代表零個或一個以上的任意字元；
- (2)萬用字元(_)底線符號代表單一個數的任意字元。

【注意事項】 Like模糊相似條件的萬用字元之比較

撰寫SQL 語法環境	Access	SQL Server
比對一個字元	「?」	「_」
比對多個字元	「*」	「%」
比對一個數字	「#」	「#」
包含指定範圍	[A-C]代表包含A到C的任何單一字元	
排除包含指定範圍	[^A-C]代表排除A到C的任何單一字元	

【實例1】

在「學生資料表」中查詢姓名開頭姓“李”的學生基本資料。

【解答】

SQL指令
<pre>SELECT * FROM 學生資料表 WHERE 姓名 Like '李*'</pre>

【查詢結果】

學號	姓名	系碼
S0002	李四	D002
S0005	李安	D004

一、學生資料表

	學號	姓名	系碼
#1	S0001	張三	D001
#2	S0002	李四	D002
#3	S0003	王五	D003
#4	S0004	陳明	D001
#5	S0005	李安	D004

【實例2】

在「學生資料表」中查詢姓名開頭姓「李」或「王」的學生基本資料。

【解答】

```
SQL指令  
SELECT *  
FROM 學生資料表  
WHERE 姓名 Like '[李王]*';
```

【查詢結果】

學號	姓名	系碼
S0002	李四	D002
S0003	王五	D003
S0005	李安	D004

一、學生資料表

	學號	姓名	系碼
#1	S0001	張三	D001
#2	S0002	李四	D002
#3	S0003	王五	D003
#4	S0004	陳明	D001
#5	S0005	李安	D004

IN集合條件

【定義】IN為集合運算子，只要符合集合之其中一個元素，將會被選取。

【使用時機】篩選的對象是兩個或兩個以上。

【實例 1】

在「選課資料表」中查詢學生任選一個「課程代號為C004 或 課程代號為C005」的學生的「學號、課號及成績」

【解答】

```
SQL指令
SELECT 學號, 課號, 成績
FROM 選課資料表
WHERE 課號 In ('C004','C005')
```

【查詢結果】

學號	課號	成績
S0002	C005	63
S0003	C004	92
S0001	C005	73
S0003	C005	70
S0004	C004	88
S0004	C005	68
S0005	C005	

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

【實例2】

請在「學生資料表」中，列出學號為S0001~S0003的同學之「學號，姓名及系碼」

【解答】

SQL指令

```
SELECT 學號,姓名,系碼  
FROM 學生資料表  
WHERE 學號 In ('S0001', 'S0002', 'S0003')
```

一、學生資料表

	學號	姓名	系碼
#1	S0001	張三	D001
#2	S0002	李四	D002
#3	S0003	王五	D003
#4	S0004	陳明	D001
#5	S0005	李安	D004

【查詢結果】

	學號	姓名	系碼
	S0001	張三	D001
	S0002	李四	D001
	S0003	王五	D002

Between / And範圍條件

【定義】

Between/And是用來指定一個範圍，表示資料值必須在最小值(含)與最大值(含)之間的範圍資料。註：等同於「大於最小值 And 小於最大值」

【實例】在「選課資料表」中查詢成績60到90
課號及成績」

SQL指令

```
SELECT 學號, 課號, 成績  
FROM 選課資料表  
WHERE 成績 Between 60 And 90
```

等同於

學號	課號	成績
S0002	C005	63
S0001	C005	73
S0003	C005	70
S0004	C003	75
S0004	C004	88
S0004	C005	68

成績 >= 60 And 成績 <= 90

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

使用「聚合函數」

【定義】

在SQL中提供**聚合函數**來讓使用者**統計**資料表中數值資料的**最大值**、**最小值**、**平均值**及**合計值**等等。其常用的聚合函數的種類如表所示：

聚合函數	說明
Count(*)	計算個數函數
Count(欄位名稱)	計算該欄位名稱之不具NULL值列的總數
Avg	計算平均函數
Sum	計算總合函數
Max	計算最大值函數
Min	計算最小值函數

記錄筆數(Count)

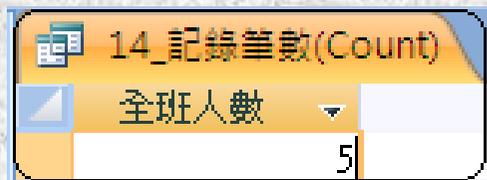
【定義】 COUNT函數是用來計算橫列記錄的筆數。

【實例1】 在「學生資料表」中查詢目前選修課程的全班人數

【解答】

SQL指令
SELECT Count(*) AS 全班人數 FROM 學生資料表

【查詢結果】



The screenshot shows a window titled "14_記錄筆數(Count)". It contains a table with one row and one column. The column header is "全班人數" (Total Class Size) and the value in the row is "5".

全班人數
5

一、學生資料表

	學號	姓名	系碼
#1	S0001	張三	D001
#2	S0002	李四	D002
#3	S0003	王五	D003
#4	S0004	陳明	D001
#5	S0005	李安	D004

平均數(AVG)

【定義】 **AVG**函數用來傳回一組記錄在某欄位內容值中的**平均值**。

【實例】在「選課資料表」中查詢有選修「課程代號為C005」的全班
平均成績

【解答】

SQL指令

```
SELECT AVG(成績) AS 資料庫平均成績  
FROM 選課資料表  
WHERE 課號='C005'
```

【查詢結果】

15_平均數(AVG)	
資料庫平均成:	▼
	68.5

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

總和(Sum)

【定義】SUM函數是用來傳回一組記錄在某欄位內容值的總和。

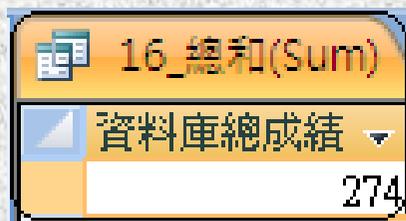
【實例】在「選課資料表」中查詢有選修「課程代號為C005」的全班
總成績

【解答】

SQL指令

```
SELECT SUM(成績) AS 資料庫總成績  
FROM 選課資料表  
WHERE 課號='C005'
```

【查詢結果】



16_總和(Sum)	
資料庫總成績	274

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

最大值(Max)

【定義】 MAX函數用來傳回一組記錄在某欄位內容值中的最大值。

【實例】 在「選課資料表」中查詢有選修「課程代號為C005」的全班
成績最高分
【解答】

SQL指令

```
SELECT MAX(成績) AS 資料庫最高分  
FROM 選課資料表  
WHERE 課號='C005'
```

【查詢結果】

17_最大值(Max)	
資料庫最高分	73

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

最小值(Min)

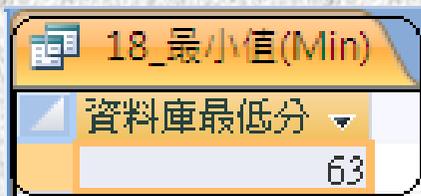
【定義】MIN函數用來傳回一組記錄在某欄位內容值中的最小值。

【實例】在「選課資料表」中查詢有選修「課程代號為C005」的全班成績最低分
【解答】

SQL指令

```
SELECT MIN(成績) AS 資料庫最低分  
FROM 選課資料表  
WHERE 課程代號='C005'
```

【查詢結果】



18_最小值(Min)	
資料庫最低分	63

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

使用「排序及排名次」

【定義】

雖然撰寫SQL指令來查詢所須的資料非常容易，但如果顯示的結果筆數非常龐大而沒有按照某一順序及規則來顯示，可能會顯得非常混亂。還好SQL指令還有提供排序的功能。

其常用的排序及排名次的子句種類如表所示：

排序及排名次指令	說明
ORDER BY成績 Asc	Asc ←可以省略(由小至大) <預設模式>
ORDER BY成績 Desc	Desc ←不可以省略(由大至小)
Top N	取排名前N名
Top N Percent	取排名前N%名

Asc遞增排序

【定義】資料記錄的排序方式是由小至大排列。

【實例】在「選課資料表」中查詢全班成績由低到高分排序

【解答】

SQL指令

```
SELECT 學號, 課號, 成績  
FROM 選課資料表  
ORDER BY 成績 Asc
```

【查詢結果】

學號	課號	成績
S0005	C005	
S0001	C001	56
S0002	C005	63
S0004	C005	68
S0003	C005	70
S0001	C005	73
S0004	C003	75
S0004	C004	88
S0003	C004	92
S0002	C004	92

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

Desc遞減排序

【定義】資料記錄的排序方式是**由大至小排列**。

【實例】在「選課資料表」中查詢的全班成績**由高到低分排序**

【解答】

SQL指令

```
SELECT 學號, 課號, 成績  
FROM 選課資料表  
ORDER BY 成績 DESC
```

【查詢結果】

查詢1	學號	課號	成績
	S0003	C004	92
	S0002	C004	92
	S0004	C004	88
	S0004	C003	75
	S0001	C005	73
	S0003	C005	70
	S0004	C005	68
	S0002	C005	63
	S0001	C001	56
	S0005	C005	

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

比較複雜的排序

【定義】指定一個欄位以上來做排序時，則先以第一個欄位優先排序，當資料相同時，則再進行第二個欄位進行排序，依此類推。

【實例】在「選課資料表」中查詢結果按照學號昇冪排列之後，再依成績昇冪排列。

【解答】

```
SQL指令
SELECT 學號, 課號, 成績
FROM 選課資料表
ORDER BY 學號, 成績
```

欄位名稱之間必須要以「,(逗點)來做區隔」

【查詢結果】

依學號
低→高

學號	課號	成績
S0001	C001	56
S0001	C005	73
S0002	C005	63
S0002	C004	92
S0003	C005	70
S0003	C004	92
S0004	C005	68
S0004	C003	75
S0004	C004	88
S0005	C005	NULL

依成績
低→高

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

依學號

未依成績

使用「群組化」

【定義】

利用SQL語言，我們可以將某些**特定欄位的值相同的記錄**全部組合起來，以進行**群組化**，接著就可以在這個群組內求出**各種統計分析**。

【語法】 Group By欄位1,欄位2,...,欄位n [Having 條件式]

(1) Group By 可單獨存在，它是將**數個欄位組合**起來，以做為**每次動作的依據**。

(2) [Having 條件式]是將數個欄位中以**有條件的組合**。
它**不可以單獨存在**。

(3) WHERE子句與HAVING子句之差別

	WHERE子句	HAVING子句
執行順序	GROUP BY之前	GROUP BY之後
聚合函數	不能使用聚合函數	可以使用

Group By 欄位

【定義】 Group By 可單獨存在，它是將數個欄位組合起來，以做為每次動作的依據。

【語法】

```
Select 欄位1 · 欄位2 · 聚合函數運算  
From 資料表  
Where 過濾條件  
Group By 欄位1 · 欄位2
```

[實例1]

在「選課資料表」中，查詢每一位同學各選幾門科目。

【解答】

SQL指令
SELECT 學號, Count(*) AS 選科目數
FROM 選課資料表
GROUP BY 學號

【註】在Select所篩選的**非聚合函數**。

例如：學號，一定會在 Group By後出現。

【查詢結果】



學號	選科目數
S0001	2
S0002	2
S0003	2
S0004	3
S0005	1

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

[實例2]

在「選課資料表」中計算每一位同學所修之科目的平均成績

【解答】

SQL指令

```
SELECT 學號, AVG(成績) AS 平均成績  
FROM 選課資料表  
GROUP BY 學號
```

【查詢結果】

學號	平均成績
S0001	64.5
S0002	77.5
S0003	81
S0004	77
S0005	

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

Having 條件式

【定義】 Having 條件式是將數個欄位中以有條件的組合。

它不可以單獨存在。

【實例1】 在「選課資料表」中，計算所修之科目的平均成績，大於等於70者顯示出來。

【解答】

SQL指令

```
SELECT 學號, AVG(成績) AS 平均成績  
FROM 選課資料表  
GROUP BY 學號  
HAVING AVG(成績) >= 70
```

【查詢結果】

學號	平均成績
S0002	77.5
S0003	81
S0004	77

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL

[實例2]

在「選課資料表」中，將選修課程在二科及二科以上的學生學號資料列出來。

【解答】

SQL指令

```
SELECT 學號, Count(*) AS 選修數目  
FROM 選課資料表  
GROUP BY 學號  
HAVING COUNT(*)>=2
```

【查詢結果】



學號	選修數目
S0001	2
S0002	2
S0003	2
S0004	3

三、選課資料表

	學號	課號	成績
#1	S0001	C001	56
#2	S0001	C005	73
#3	S0002	C002	92
#4	S0002	C005	63
#5	S0003	C004	92
#6	S0003	C005	70
#7	S0004	C003	75
#8	S0004	C004	88
#9	S0004	C005	68
#10	S0005	C005	NULL