



關聯式查詢



關聯查詢

因為資料庫設計與正規化的原因，單一資料庫表格的查詢似乎比較少，反而因為正規化把在單一表格分割成兩個或兩個以上的表格，所以原本可以在原單一表格中查詢到的資料，現在必須透過兩個或兩個以上的表格查詢才能得到，這種跨表格的查詢稱為關聯查詢。

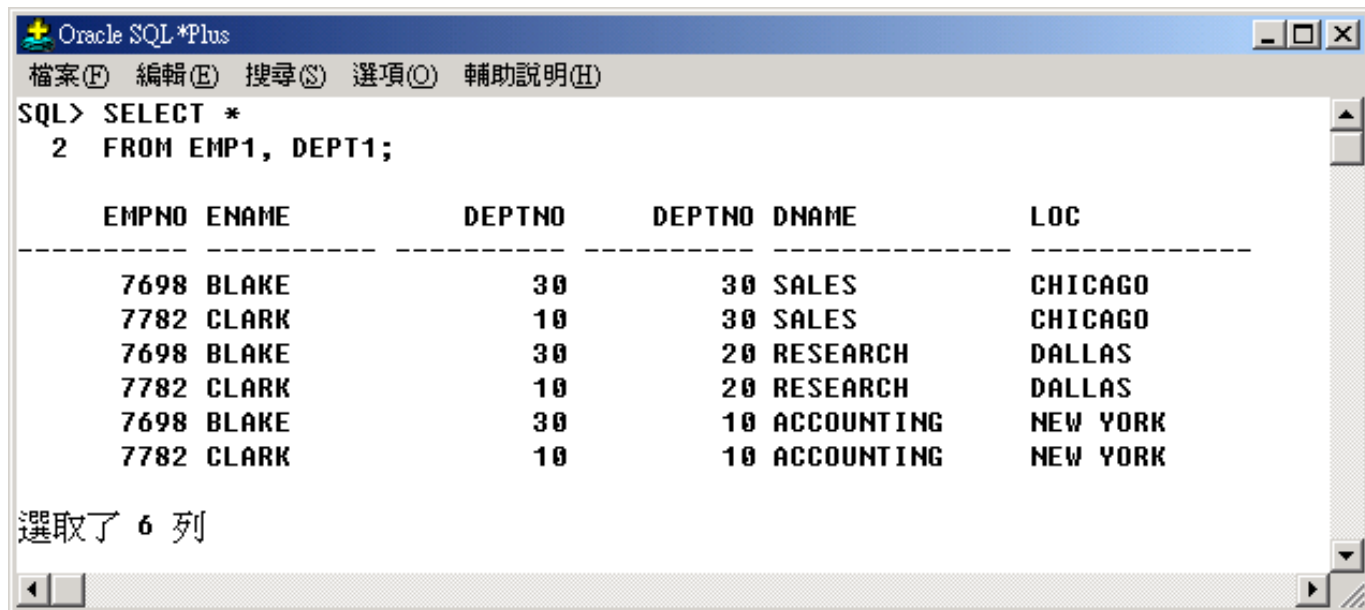


卡笛生乘積(Cartesian Product)

卡笛生乘積實際上就是一種無條件的關聯，這種操作往往會產生大量的資料列，其結果並沒有太大的意義。因此在相關的關聯操作時，往往會加上限制條件，再進行關聯運算。

兩個表格的“卡笛生乘積”形式

SELECT <欄位名11>, <欄位名12>, ..., <欄位名1m> ,
<欄位名21>, <欄位名22>, ..., <欄位名2n> ,
FROM <表格名1>, <表格名2>



The screenshot shows the Oracle SQL*Plus interface. The command prompt displays the following SQL query:

```
SQL> SELECT *  
2 FROM EMP1, DEPT1;
```

The output is a Cartesian product of the EMP1 and DEPT1 tables, showing 6 rows. The columns are EMPNO, ENAME, DEPTNO, DEPTNO, DNAME, and LOC. The results are as follows:

EMPNO	ENAME	DEPTNO	DEPTNO	DNAME	LOC
7698	BLAKE	30	30	SALES	CHICAGO
7782	CLARK	10	30	SALES	CHICAGO
7698	BLAKE	30	20	RESEARCH	DALLAS
7782	CLARK	10	20	RESEARCH	DALLAS
7698	BLAKE	30	10	ACCOUNTING	NEW YORK
7782	CLARK	10	10	ACCOUNTING	NEW YORK

選取了 6 列

等值關聯(Equal Join)

SELECT ...

FROM <表格名1>, <表格名2>

WHERE <表格名1.欄位名1> = <表格名2.欄位名2>;

P.K.			F.K.		P.K.		
<u>Empno</u>	<u>Ename</u>	...	<u>Deptno</u>		<u>Deptno</u>	<u>Dname</u>	<u>Loc</u>
7369	SMITH	...	20		10	ACCOUNTING	NEW YORK
7499	ALLEN	...	10		20	RESEARCH	DALLAS
:	:	:	:		30	SALES	CHICAGO
7566	JONES	...	10		40	OPERATIONS	BOSTON
R1(員工基本資料表格 EMP)					R2(部門基本資料表格 DEPT)		



非等值關聯

非等值關聯就是指關聯條件中不使用“ = ”運算子的關聯運算。非等值關聯能夠使用的比較運算子包括 != 、 < 、 > 、 <= 、 >= 、 BETWEEN... AND 和 LIKE 等。

自身關聯

- 自身關聯可以把一個表格看成兩個完全相同的表格 (副本)，然後再對這兩個表格在相關欄位上進行關聯，其關聯方式與多表格關聯完全相同。

<u>Empno</u>	<u>Ename</u>	<u>Mgr</u>	<u>Sal</u>	<u>Deptno</u>
7369	SMITH	7499	800	20
7499	FORD	7566	1800	10
7521	WARD	7698	1450	30
:	:	:	:	:
7566	JONES	7839	2975	10
7839	KING	-	3975	10

員工基本資料表格 EMP



▶ 表格自身關聯

SELECT ...

FROM <表格名> <表格別名1>, <表格名> <表格 別名2>
WHERE <表格別名1.欄位名1> = <表格別名2.欄位名2>;

其中：<欄位名1> 和 <欄位名2> 標記自關聯表中的兩個不同欄位，這兩個欄位要求有相同的資料型態和寬度。



子查詢

所謂子查詢是指在 **WHERE** 子句或 **HAVING** 子句的條件中出現的查詢。相對地，稱包含子查詢的查詢為父查詢或主查詢。因為子查詢使得一系列簡單查詢可以構成複雜的查詢，因此子查詢也稱為嵌套查詢。

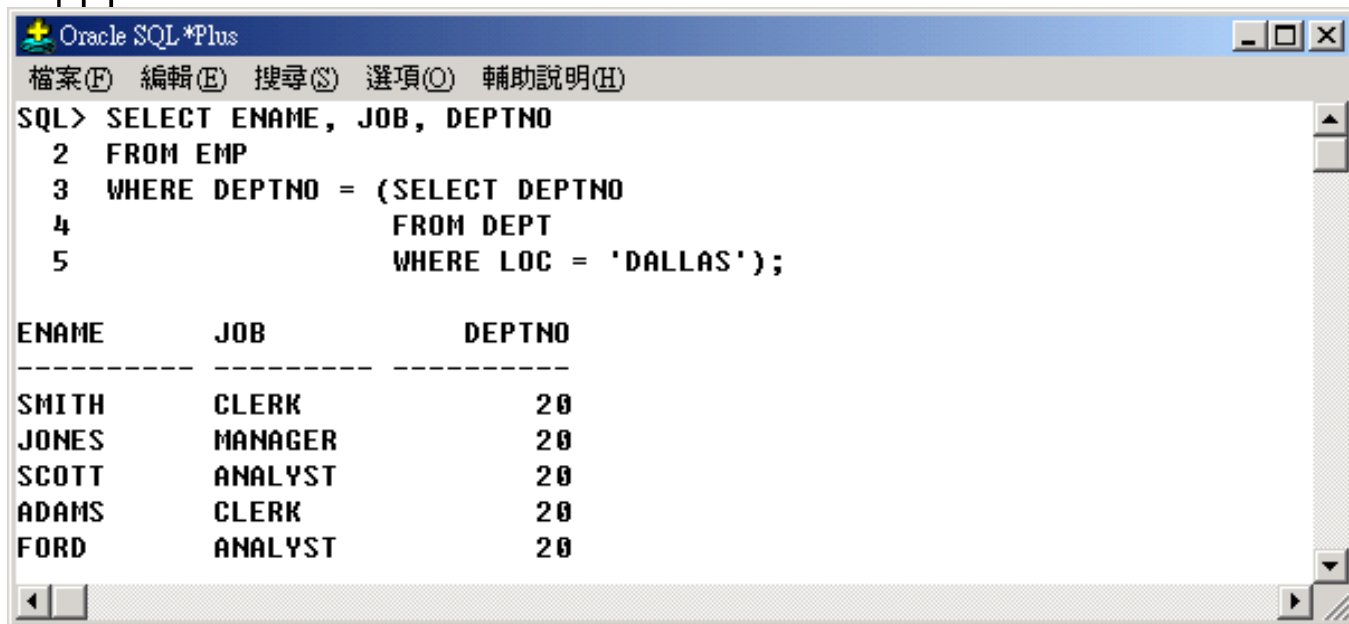


▶ 子查詢－SELECT...FROM...WHERE

```
SELECT ...  
FROM <表格名>  
WHERE <欄位名或欄位運算式> <比較運算子>  
      (SELECT ... FROM <表格名>  
        WHERE <條件> );
```

單一記錄值子查詢

此類子查詢只傳回單一記錄值，因此所有邏輯運算子（如 $>$ ， $=$ ， $<$ 等等）都可以

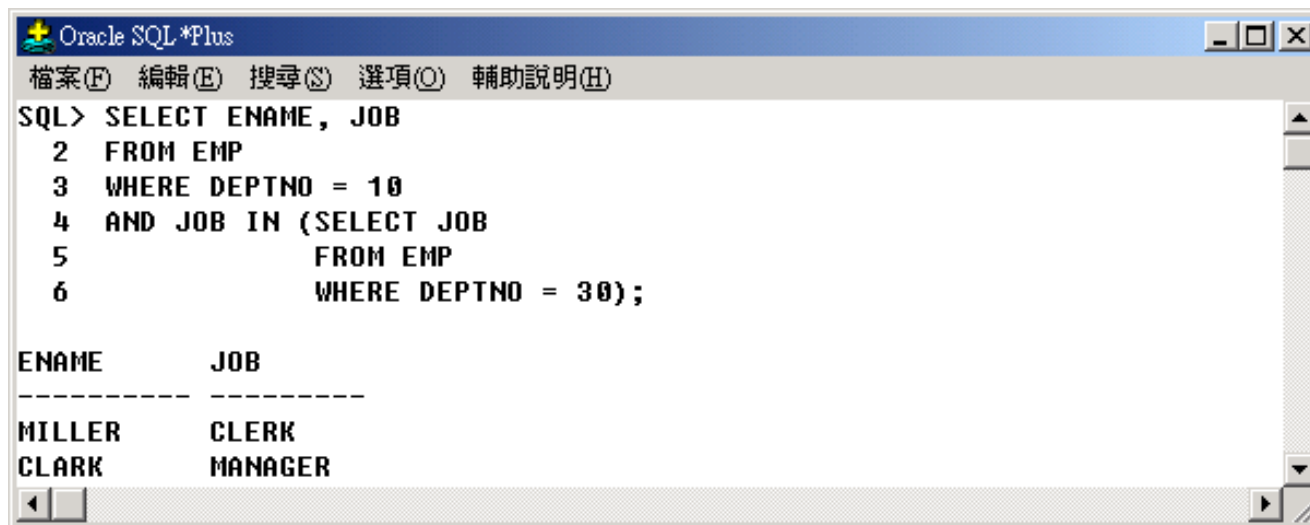


```
Oracle SQL*Plus
檔案(F) 編輯(E) 搜尋(S) 選項(O) 輔助說明(H)
SQL> SELECT ENAME, JOB, DEPTNO
2 FROM EMP
3 WHERE DEPTNO = (SELECT DEPTNO
4                 FROM DEPT
5                 WHERE LOC = 'DALLAS');

ENAME      JOB          DEPTNO
-----
SMITH      CLERK        20
JONES      MANAGER      20
SCOTT      ANALYST      20
ADAMS      CLERK        20
FORD       ANALYST      20
```

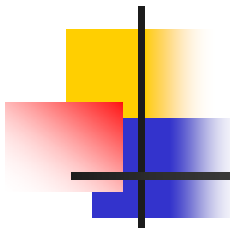
多記錄值子查詢

此類子查詢傳回不是單一值而是一組資料列。這種查詢必須用多值比較運算子與主查詢相連繫。



```
Oracle SQL*Plus
檔案(F) 編輯(E) 搜尋(S) 選項(O) 輔助說明(H)
SQL> SELECT ENAME, JOB
2 FROM EMP
3 WHERE DEPTNO = 10
4 AND JOB IN (SELECT JOB
5             FROM EMP
6             WHERE DEPTNO = 30);
```

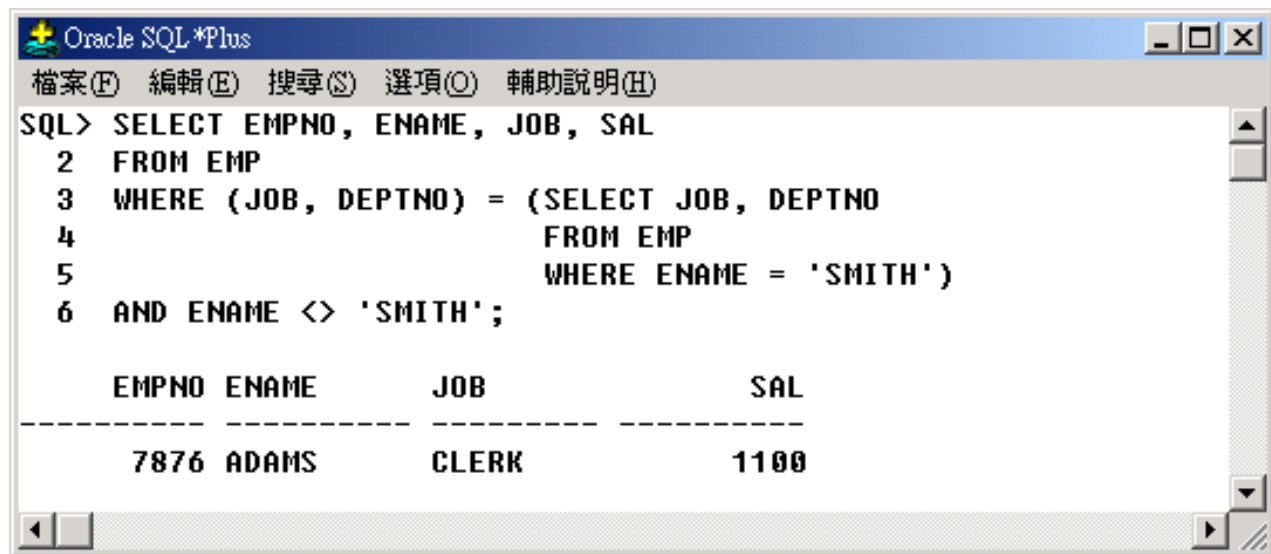
ENAME	JOB
MILLER	CLERK
CLARK	MANAGER

- 
-
- ▶ [NOT] IN 表示 [不] 屬於某集中成員的關係
[NOT IN 格式]

```
SELECT ...  
FROM    <表格名1>  
WHERE   <表格名1.欄位名1> NOT IN  
        (SELECT <表格名2.欄位名2>  
         FROM <表格名2>  
         WHERE <表格名2.欄位名3> = 運算式);
```

多欄位子查詢

子查詢中不但可以查出一個欄位的值，還可以查多個欄位。子查詢傳回欄位的個數及型態必須要與主查詢欄位的個數和型態匹配

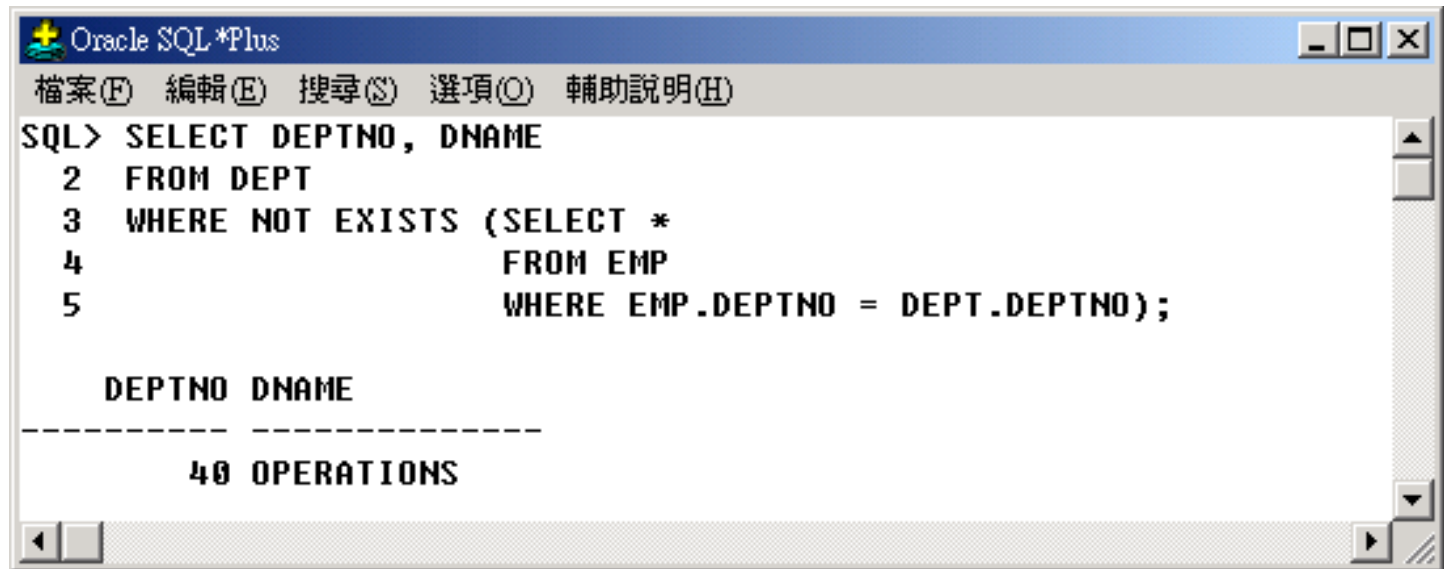


```
Oracle SQL*Plus
檔案(F) 編輯(E) 搜尋(S) 選項(O) 輔助說明(H)
SQL> SELECT EMPNO, ENAME, JOB, SAL
2 FROM EMP
3 WHERE (JOB, DEPTNO) = (SELECT JOB, DEPTNO
4 FROM EMP
5 WHERE ENAME = 'SMITH')
6 AND ENAME <> 'SMITH';

EMPNO ENAME      JOB      SAL
-----
7876 ADAMS      CLERK    1100
```

相關子查詢

相關子查詢是指子查詢的 **WHERE** 條件子句中有引用主查詢的查詢列。反之我們稱之不相關子查詢。



```
Oracle SQL*Plus
檔案(F) 編輯(E) 搜尋(S) 選項(O) 輔助說明(H)
SQL> SELECT DEPTNO, DNAME
2 FROM DEPT
3 WHERE NOT EXISTS (SELECT *
4                   FROM EMP
5                   WHERE EMP.DEPTNO = DEPT.DEPTNO);

DEPTNO DNAME
-----
40 OPERATIONS
```



子查詢的其他用法

- ▶ 在 UPDATE 敘述中提供更新的值，或是為 WHERE 子句中提供進行比較的值。

```
UPDATE <表格名>  
SET (<欄位名1>, <欄位名2>, ...) =  
  (SELECT <欄位名1>, <欄位名2>, ...  
   FROM <表格名>  
   WHERE <條件> )  
WHERE <欄位名或欄位運算式> <比較運算子> (SELECT <欄位名>  
                                             FROM <表格名>  
                                             WHERE <條件>);
```


- 
- 在 DELETE 敘述的 WHERE 子句中提供進行比較的值。

```
DELETE FROM <表格名>  
WHERE <欄位名或欄位運算式> <比較運算子> (DELETE <欄位名>  
FROM <表格名>  
WHERE <條件> );
```



- 在 INSERT 敘述中提供插入的資料列

INSERT INTO <表格名> (<欄位名>, <欄位名>, ...)

SELECT (<欄位名>, <欄位名>, ...)

FROM <表格名>

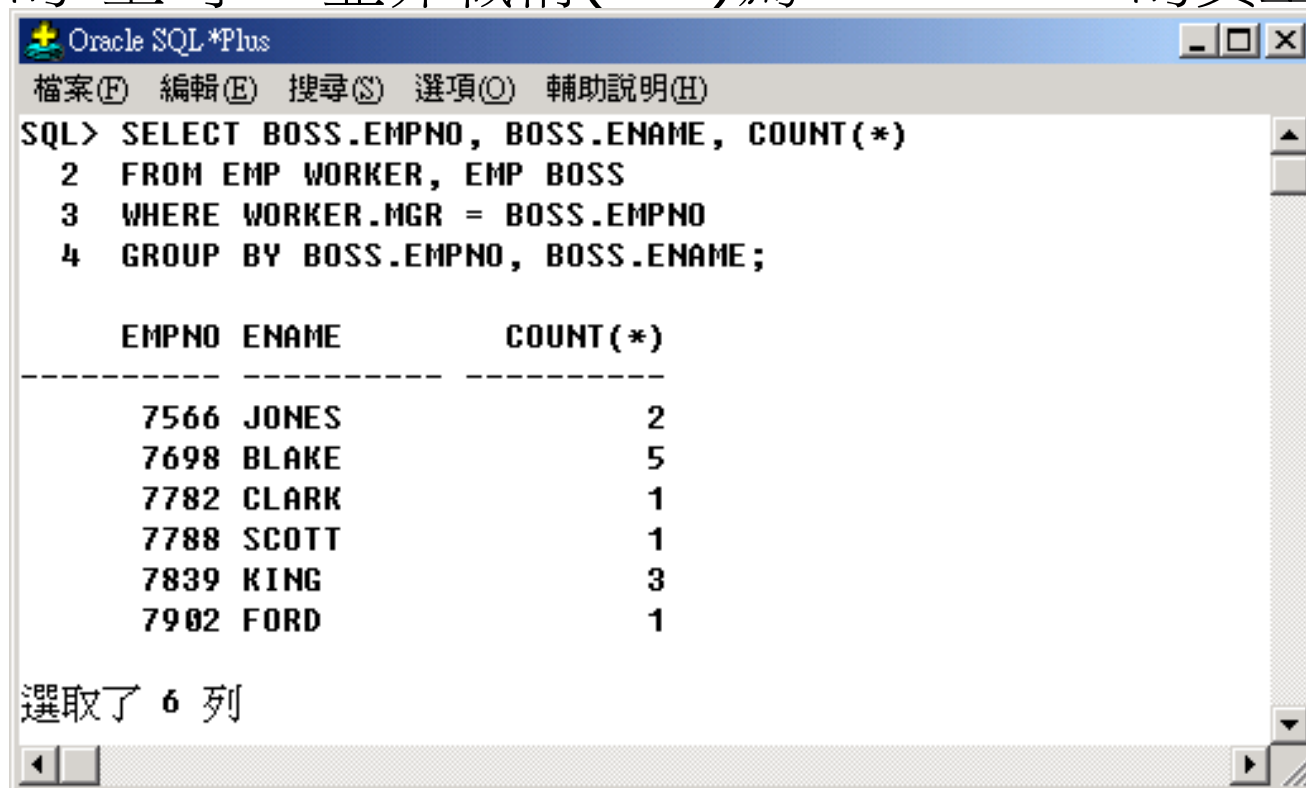
WHERE <欄位名或欄位運算式> <比較運算子>(SELECT <欄位名>
FROM <表格名>
WHERE<條件>);

- 
- 在 CREATE TABLE 命令中提供插入的資料列

```
CREATE TABLE <新表格名> AS  
SELECT <欄位名>, <欄位名>, ...  
FROM <表格名>  
WHERE <欄位名或欄位運算式> <比較運算子> (SELECT <欄位名>  
FROM <表格名>  
WHERE <條件>);
```

分組查詢

- 計算每位上司所直接管理的員工有幾位。(這裡所謂的”上司”，並非職稱(JOB)為'MANAGER'的員工)



```
Oracle SQL*Plus
檔案(F) 編輯(E) 搜尋(S) 選項(O) 輔助說明(H)
SQL> SELECT BOSS.EMPNO, BOSS.ENAME, COUNT(*)
2 FROM EMP WORKER, EMP BOSS
3 WHERE WORKER.MGR = BOSS.EMPNO
4 GROUP BY BOSS.EMPNO, BOSS.ENAME;

EMPNO ENAME          COUNT(*)
-----
7566 JONES           2
7698 BLAKE          5
7782 CLARK          1
7788 SCOTT          1
7839 KING           3
7902 FORD           1
```

選取了 6 列