Chap07. 다중행 함수와 데이터 그룹화

- ▶ 07-I 하나의 열에 출력 결과를 담는 다중행 함수
- ▶ 07-2 결과 값을 원하는 열로 묶어 출력하는 GROUP BY 절
- ▶ 07-3 GROUP BY절에 조건을 줄 때 사용하는 HAVING절
- ▶ 07-4 그룹화와 관련된 여러 함수

▶ SUM : 합계

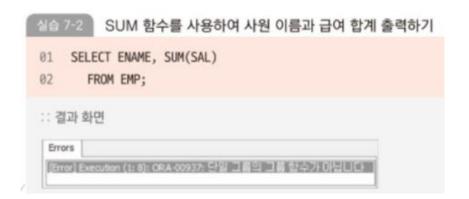
▶ COUNT : 데이터 개수

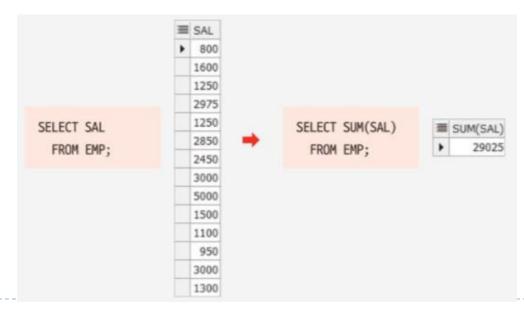
▶ MAX : 최댓값

▶ MIN : 최솟값

▶ AVG : 평균값

실습	7-1 S	UM 함수를 사용하여 급여 합계 출력하기
01	SELECT	SUM(SAL)
02	FROM	EMP;
11	결과 화면	
=	SUM(SAL)	
1		

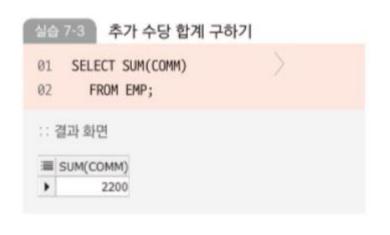






▶ SUM : 합계

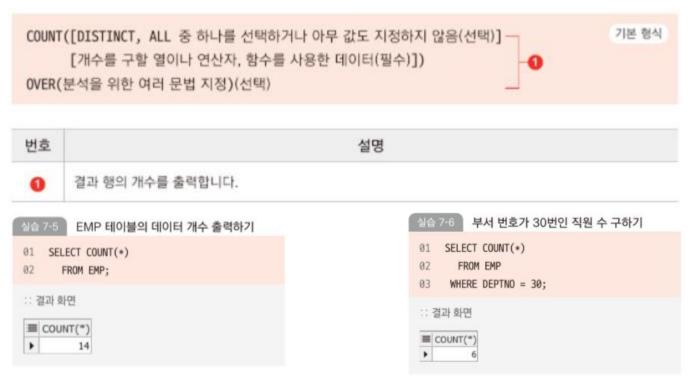
	DISTINCT, ALL 중 하나를 선택하거나 아무 값도 지정하지 않음(선택)] 합계를 구할 열이나 연산자, 함수를 사용한 데이터(필수)])	기본 형식
번호	설명	
0	합계를 구할 데이터를 지정합니다.	







▶ COUNT : 데이터 개수





▶ COUNT : 데이터 개수



▶ MAX : 최댓값



▶ MIN : 최솟값

MIN([DISTINCT, ALL 중 하나를 선택하거나 아무 값도 지정하지 않음(선택)] [최솟값을 구할 열이나 연산자, 함수를 사용한 데이터(필수)])

OVER(분석을 위한 여러 문법을 지정)(선택)





▶ AVG : 평균값

AVG([DISTINCT, ALL 중 하나를 선택하거나 아무 값도 지정하지 않음(선택)] 기본 형식 [평균 값을 구할 열이나 연산자, 함수를 사용한 데이터(필수)]) 1 0VER(분석을 위한 여러 문법을 지정)(선택)

 번호
 설명

 ① 결과 행의 평균 값을 반환합니다.

실습 7-14 부서 번호가 30인 사원들의 평균 급여 출력하기

01 SELECT AVG(SAL)

02 FROM EMP

03 WHERE DEPTNO = 30;

:: 결과 화면

■ AVG(SAL)

▶ 1566.6666666667



07-2 결과 값을 원하는 열로 묶어 줄력하는 GROUP BY 절

실습 7-16 집합 연산자를 사용하여 각 부서별 평균 급여 출력하기

- 01 SELECT AVG(SAL), '10' AS DEPTNO FROM EMP WHERE DEPTNO = 10
- 02 UNION ALL
- 03 SELECT AVG(SAL), '20' AS DEPTNO FROM EMP WHERE DEPTNO = 20
- 04 UNION ALL
- 05 SELECT AVG(SAL), '30' AS DEPTNO FROM EMP WHERE DEPTNO = 30;

:: 결과 화면

4.7.7		AVG(SAL)	DEPTNO
	٠	2916.6666666667	10
		2175	20
		1566.6666666667	30



07-2 결과 값을 원하는 열로 묶어 줄력하는 GROUP BY 절

▶ GROUP BY 절의 기본 사용법

```
SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]
```

FROM [조회할 테이블 이름]

WHERE [조회할 행을 선별하는 조건식]

GROUP BY [그룹화할 열을 지정(여러 개 지정 가능)]

ORDER BY [정렬하려는 열 지정]

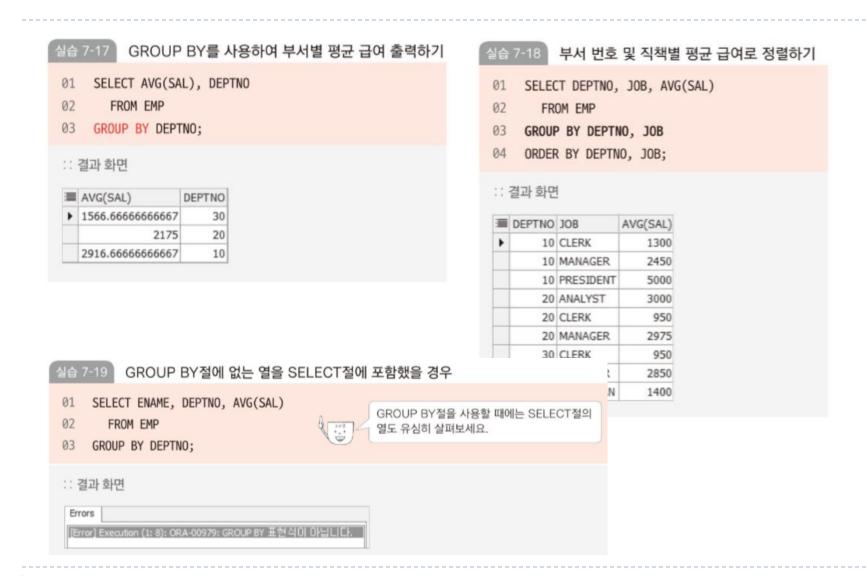
번호	키워드	필수 요소	선택 요소	설명
0	GROUP BY	그룹화할 열 또는 데이터 지정	-	특정 열 또는 데이터를 기준으로 데이터를 그룹으로 묶습니다.

▶ GROUP BY절을 사용할 때 유의점

◎ GROUP BY절에는 별칭이 인식되지 않습니다. 즉 열 이름이나 연산식을 그대로 지정해 주어야 합니다.



07-2 결과 값을 원하는 열로 묶어 줄력하는 GROUP BY 절



07-3 GROUP BY설에 조건을 술 때 사용하는 HAVING절

▶ HAVING 절의 기본 사용법

```
SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름] FROM [조회할 테이블 이름]
```

WHERE [조회할 행을 선별하는 조건식]

GROUP BY [그룹화할 열 지정(여러 개 지정 가능)]

HAVING [출력 그룹을 제한하는 조건식]

ORDER BY [정렬하려는 열 지정];

번호	키워드	필수 요소	선택 요소	설명
0	HAVING	조건식	-	GROUP BY절을 사용해 그룹화된 결과 중 출력 그룹을 선별하는 조건식을 지정합니다.



07-3 GROUP BY설에 조건을 술 때 사용하는 HAVING절

실습 7-20 GROUP BY절과 HAVING절을 사용하여 출력하기

- 01 SELECT DEPTNO, JOB, AVG(SAL)
- 02 FROM EMP
- 03 GROUP BY DEPTNO, JOB
- 04 HAVING AVG(SAL) >= 2000
- 05 ORDER BY DEPTNO, JOB;

:: 결과 화면

	DEPTNO	JOB	AVG(SAL)
٠	10	CLERK	1300
	10	MANAGER	2450
	10	PRESIDENT	5000
	20	ANALYST	3000
	20	CLERK	950
	20	MANAGER	2975
	30	CLERK	950
	30	MANAGER	2850
	30	SALESMAN	1400

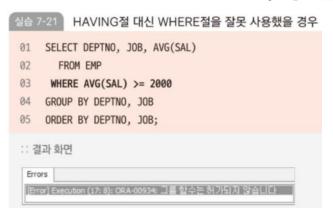
≡	DEPTNO	JOB	AVG(SAL)
٠	10	MANAGER	2450
	10	PRESIDENT	5000
	20	ANALYST	3000
	20	MANAGER	2975
	30	MANAGER	2850

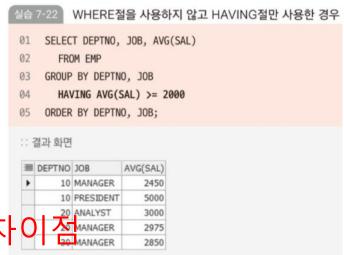
HAVING절을 추가하지 않았을 때

HAVING절을 추가했을 때

07-3 GROUP BY설에 조건을 술 때 사용하는 HAVING절

▶ HAVING절을 사용할 때 유의점





▶ WHERE절과 HAVING절의 차이



- ▶ ROLLUP, CUBE: 그룹화 데이터의 합계
- ▶ GROUPING SETS : 지정한 각 열별 그룹화
- ▶ GROUPING : ROLLUP, CUBE와 함께 사용 (하나)
- ▶ GROUPING_ID : ROLLUP, CUBE와 함께 사용 (여럿)



▶ ROLLUP, CUBE : 그룹화 데이터의 합계

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]

기본 형식

FROM [조회할 테이블 이름]

WHERE [조회할 행을 선별하는 조건식]

GROUP BY ROLLUP [그룹화 열 지정(여러 개 지정 가능)]; -1

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]

기본 형식

FROM [조회할 테이블 이름]

WHERE [조회할 행을 선별하는 조건식]

GROUP BY CUBE [그룹화 열 지정(여러 개 지정 가능)]; -2

번호	키워드	필수 요소	선택 요소	설명
0,0	ROLLUP, CUBE	그룹화 열 지정	-	그룹화 데이터의 합계를 함께 출력하는 데 사용합니다.



ROLLUP

실습 7-24 기존 GROUP BY절만 사용한 그룹화

- 01 SELECT DEPTNO, JOB, COUNT(*), MAX(SAL), SUM(SAL), AVG(SAL)
- 02 FROM EMP
- 03 GROUP BY DEPTNO, JOB
- 04 ORDER BY DEPTNO, JOB;

:: 결과 화면

	DEPTNO	JOB	COUNT(*)	MAX(SAL)	SUM(SAL)	AVG(SAL)
٠	10	CLERK	1	1300	1300	1300
	10	MANAGER	1	2450	2450	2450
	10	PRESIDENT	1	5000	5000	5000
	20	ANALYST	2	3000	6000	3000
	20	CLERK	2	1100	1900	950
	20	MANAGER	1	2975	2975	2975
	30	CLERK	1	950	950	950
	30	MANAGER	1	2850	2850	2850
	30	SALESMAN	4	1600	5600	1400

실습 7-25 ROLLUP 함수를 적용한 그룹화

- 01 SELECT DEPTNO, JOB, COUNT(*), MAX(SAL), SUM(SAL), AVG(SAL)
- 02 FROM EMP
- 03 GROUP BY ROLLUP(DEPTNO, JOB);

:: 결과 화면

=	DEPTNO	JOB	COUNT(*)	MAX(SAL)	SUM(SAL)	AVG(SAL)
٠	10	CLERK	1	1300	1300	1300
	10	MANAGER	1	2450	2450	2450
	10	PRESIDENT	1	5000	5000	5000
	10		3	5000	8750	2916.6666666667
	20	CLERK	2	1100	1900	950
	20	ANALYST	2	3000	6000	3000
	20	MANAGER	1	2975	2975	2975
	20		5	3000	10875	2175
	30	CLERK	1	950	950	950
	30	MANAGER	1	2850	2850	2850
	30	SALESMAN	4	1600	5600	1400
	30		6	2850	9400	1566.6666666667
			14	5000	29025	2073.21428571429

► CUBE

실습 7-26 CUBE 함수를 적용한 그룹화

- 01 SELECT DEPTNO, JOB, COUNT(*), MAX(SAL), SUM(SAL), AVG(SAL)
- 02 FROM EMP
- 03 GROUP BY CUBE(DEPTNO, JOB)
- 04 ORDER BY DEPTNO, JOB;

:: 결과 화면

	DEPTNO	JOB	COUNT(*)	MAX(SAL)	SUM(SAL)	AVG(SAL)
١	10	CLERK	1	1300	1300	1300
	10	MANAGER	1	2450	2450	2450
	10	PRESIDENT	1	5000	5000	5000
	10		3	5000	8750	2916.6666666667
	20	ANALYST	2	3000	6000	3000
	20	CLERK	2	1100	1900	950
	20	MANAGER	1	2975	2975	2975
	20		5	3000	10875	2175
	30	CLERK	1	950	950	950
	30	MANAGER	1	2850	2850	2850
	30	SALESMAN	4	1600	5600	1400
	30		6	2850	9400	1566.6666666667
_		ANALYST	2	3000	6000	3000
		CLERK	4	1300	4150	1037.5
		MANAGER	3	2975	8275	2758.33333333333
		PRESIDENT	1	5000	5000	5000
_		SALESMAN	4	1600	5600	1400
			14	5000	29025	2073.21428571429



ROLLUP VS CUBE

ROLLUP(A, B, C)

- 1. A 그룹별 B 그룹별 C 그룹에 해당하는 결과 출력
- 2. A 그룹별 B 그룹에 해당하는 결과 출력
- 3. A 그룹에 해당하는 결과 출력
- 4. 전체 데이터 결과 출력

CUBE(A, B, C)

- 1. A 그룹별 B 그룹별 C 그룹에 해당하는 결과 출력
- 2. A 그룹별 B 그룹의 결과 출력
- 3. B 그룹별 C 그룹의 결과 출력
- 4. A 그룹별 C 그룹의 결과 출력
- 5. A 그룹 결과
- 6. B 그룹 결과
- 7. C 그룹 결과
- 8. 전체 데이터 결과

ROLLUP VS CUBE

실습 7-27 DEPTNO를 먼저 그룹화한 후 ROLLUP 함수에 JOB 지정하기 실습 7-28 JOB을 먼저 그룹화한 후 ROLLUP 함수에 DEPTNO 지정하기 01 SELECT DEPTNO, JOB, COUNT(*) 01 SELECT DEPTNO, JOB, COUNT(*) FROM EMP FROM EMP GROUP BY JOB, ROLLUP(DEPTNO); 03 GROUP BY DEPTNO, ROLLUP(JOB); :: 결과 화면 :: 결과 화면 ■ DEPTNO JOB COUNT(*) **≡** DEPTNO JOB COUNT(*) 10 CLERK 10 CLERK 10 MANAGER 20 CLERK 10 PRESIDENT 30 CLERK 10 CLERK 20 CLERK 20 ANALYST 20 ANALYST ANALYST 20 MANAGER 10 MANAGER 20 MANAGER 30 CLERK 30 MANAGER 30 MANAGER MANAGER 30 SALESMAN 30 SALESMAN 6 30 SALESMAN 10 PRESIDENT PRESIDENT



▶ GROUPING SETS : 지정한 각 열별 그룹화

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]
FROM [조회할 테이블 이름]
WHERE [조회할 행을 선별하는 조건식]
GROUP BY GROUPING SETS [그룹화 열 지정(여러 개 지정 가능)]; 4

번호	키워드	필수 요소	선택 요소	설명
0	GROUPING SETS	그룹화 열	-	여러 그룹화 대상 열의 결과 값을 각각 같은 수준으로 출력합니다.

실습 7-29 GROUPING SETS 함수를 사용하여 열별로 그룹으로 묶어 출력하기 SELECT DEPTNO, JOB, COUNT(*) 02 FROM EMP GROUP BY GROUPING SETS(DEPTNO, JOB) ORDER BY DEPTNO, JOB; :: 결과 화면 ■ DEPTNO JOB COUNT(*) 10 20 30 ANALYST CLERK MANAGER PRESIDENT

SALESMAN

▶ GROUPING : ROLLUP, CUBE와 함께 사용 (하나)

```
SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]
GROUPING [GROUP BY절에 ROLLUP 또는 CUBE에 명시한 그룹화 할 열 이름]
FROM [조회할 테이블 이름]
WHERE [조회할 행을 선별하는 조건식]
GROUP BY ROLLUP 또는 CUBE [그룹화할 열]; —1
```

번호	키워드	필수 요소	선택 요소	설명
0	GROUPING	그룹화 여부 를 확인할 열	-	현재 결과가 그룹화 대상 열의 그룹화가 이루어진 상태 의 집계인지 여부를 출력합니다.

기본 형식



▶ GROUPING : ROLLUP, CUBE와 함께 사용 (하나)

```
실습 7-30 DEPTNO, JOB열의 그룹화 결과 여부를 GROUPING 함수로 확인하기
       SELECT DEPTNO, JOB, COUNT(*), MAX(SAL), SUM(SAL), AVG(SAL),
  02
                GROUPING(DEPTNO),
                GROUPING(JOB)
  04
          FROM EMP
       GROUP BY CUBE(DEPTNO, JOB)
      ORDER BY DEPTNO, JOB;
:: 결과 화면
■ DEPTNO JOB
                COUNT(*) MAX(SAL) SUM(SAL) AVG(SAL)
                                                           GROUPING(DEPTNO) GROUPING(JOB
       10 CLERK
                               1300
                                        1300
                                                       1300
      10 MANAGER
                               2450
                                        2450
                                                       2450
      10 PRESIDENT
                               5000
                                        5000
                                                       5000
                                                                          0
                               5000
                                        8750 2916.6666666667
                                                                          0
                                                                          0
      20 ANALYST
                               3000
                                        6000
                                                       3000
       20 CLERK
                               1100
                                        1900
                                                       950
       20 MANAGER
                               2975
                                        2975
                                                       2975
                                                                                      0
       20
                               3000
                                       10875
                                                       2175
       30 CLERK
                                950
                                        950
                                                       950
                                                                          0
      30 MANAGER
                               2850
                                        2850
                                                       2850
      30 SALESMAN
                                        5600
                                                       1400
                                                                          0
                               1600
                               2850
                                        9400 1566,66666666667
         ANALYST
                               3000
                                        6000
                                                       3000
         CLERK
                               1300
                                        4150
                                                     1037.5
         MANAGER
                                        8275 2758.33333333333
                               2975
         PRESIDENT
                                        5000
                                                       5000
                               5000
         SALESMAN
                                        5600
                                                       1400
                               1600
                               5000
                                       29025 2073.21428571429
```

▶ GROUPING : ROLLUP, CUBE와 함께 사용 (하나)

:: 결과 화면

```
일을 7-31 DECODED문으로 GROUPING 함수를 적용하여 결과 표기하기

81 SELECT DECODE(GROUPING(DEPTNO), 1, 'ALL_DEPT', DEPTNO) AS DEPTNO,

82 DECODE(GROUPING(JOB), 1, 'ALL_JOB', JOB) AS JOB,

83 COUNT(*), MAX(SAL), SUM(SAL), AVG(SAL)

84 FROM EMP

85 GROUP BY CUBE(DEPTNO, JOB)

86 ORDER BY DEPTNO, JOB;
```

	DEPTNO	JOB	COUNT(*)	MAX(SAL)	SUM(SAL)	AVG(SAL)
٠	10	ALL_JOB	3	5000	8750	2916.6666666667
	10	CLERK	1	1300	1300	1300
	10	MANAGER	1	2450	2450	2450
	10	PRESIDENT	1	5000	5000	5000
	20	ALL_JOB	5	3000	10875	2175
	20	ANALYST	2	3000	6000	3000
	20	CLERK	2	1100	1900	950
	20	MANAGER	1	2975	2975	2975
	30	ALL_JOB	6	2850	9400	1566.6666666667
	30	CLERK	1	950	950	950
	30	MANAGER	1	2850	2850	2850
	30	SALESMAN	4	1600	5600	1400
	ALL_DEPT	ALL_JOB	14	5000	29025	2073.21428571429
	ALL_DEPT	ANALYST	2	3000	6000	3000
	ALL_DEPT	CLERK	4	1300	4150	1037.5
	ALL_DEPT	MANAGER	3	2975	8275	2758.33333333333
	ALL_DEPT	PRESIDENT	1	5000	5000	5000
	ALL DEPT	SALESMAN	4	1600	5600	1400



▶ GROUPING_ID : ROLLUP, CUBE와 함께 사용 (여럿)

기본 형식

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]
 GROUPING_ID [그룹화 여부를 확인할 열(여러 개 지정 가능)]
FROM [조회할 테이블 이름] WHERE [조회할 행을 선별하는 조건식]
GROUP BY ROLLUP 또는 CUBE [그룹화 할 열]; —1

번호	키워드	필수 요소	선택 요소	설명
0	GROUPING_ID	그룹화 여부를 확인할 열	-	GROUPING 함수처럼 특정 열의 그룹화 여부 를 출력할 수 있으며, 검사할 열을 여러 개 지 정할 수 있습니다.

GROUPING_ID(a, b)

그룹화 된 열	그룹화 비트 벡터	최종 결괴
a, b	0 0	0
а	0 1	1
b	10	2
없음	11	3



▶ GROUPING_ID : ROLLUP, CUBE와 함께 사용 (여럿)

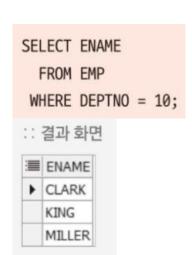


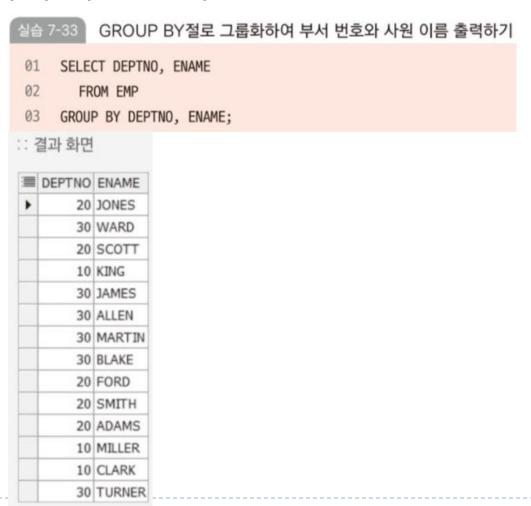


▶ LISTAGG : 그룹 데이터 가로 출력

▶ PIVOT, UNPIVOT : 행/열 바꾸어 출력

▶ LISTAGG : 그룹 데이터 가로 출력







▶ LISTAGG : 그룹 데이터 가로 출력

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]

기본 형식

LISTAGG([나열할 열(필수)], [각 데이터를 구분하는 구분자(선택)])

WITHIN GROUP(ORDER BY 나열할 열의 정렬 기준 열 (선택)) — 1

FROM [조회할 테이블 이름]

WHERE [조회할 행을 선별하는 조건식];

번호	키워드	필수 요소	선택 요소	설명
0	LISTAGG ~ WITHIN GROUP	나열할 열	각 데이터를 구분하는 구분자, 지정하지 않을 경우 NULL이 기본값이 됩니다.	그룹화 데이터를 하나의 열에 가로로 나열하여 출력하는 데 사용합니다.



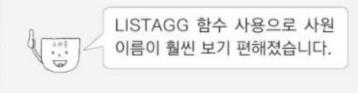
▶ LISTAGG : 그룹 데이터 가로 출력

실습 7-34 부서별 사원 이름을 나란히 나열하여 출력하기

```
01 SELECT DEPTNO,
02 LISTAGG(ENAME, ', ')
03 WITHIN GROUP(ORDER BY SAL DESC) AS ENAMES
04 FROM EMP
05 GROUP BY DEPTNO;
```

:: 결과 화면

	DEPTNO	ENAMES
٠	10	KING, CLARK, MILLER
	20	FORD, SCOTT, JONES, ADAMS, SMITH
	30	BLAKE, ALLEN, TURNER, MARTIN, WARD, JAMES





▶ PIVOT : 행/열 바꾸어 출력

실습 7-35 부서별·직책별로 그룹화하여 최고 급여 데이터 출력하기

01 SELECT DEPTNO, JOB, MAX(SAL)

02 FROM EMP

03 GROUP BY DEPTNO, JOB

04 ORDER BY DEPTNO, JOB;

	DEPTNO	JOB	MAX(SAL)
٠	10	CLERK	1300
	/ 10	MANAGER	2450
	10	PRESIDENT	5000
	20	ANALYST	3000
	20	CLERK	1100
	20	MANAGER	2975
	30	CLERK	950
	30	MANAGER	2850
	30	SALESMAN	1600

실습 7-36 PIVOT 함수를 사용하여 직책별·부서별 최고 급여를 2차원 표 형태로 출력하기 :: 결과 화면 01 SELECT * FROM(SELECT DEPTNO, JOB, SAL **≡** JOB 10 20 30 03 FROM EMP) ▶ ANALYST 3000 PIVOT(MAX(SAL) CLERK 1300 1100 950 MANAGER 2450 2975 2850 FOR DEPTNO IN (10, 20, 30) PRESIDENT 5000 SALESMAN 1600 ORDER BY JOB;

```
실습 7-37 PIVOT 함수를 사용하여 부서별·직책별 최고 급여를 2차원 표 형태로 출력하기
 01 SELECT *
       FROM(SELECT JOB, DEPTNO, SAL
             FROM EMP)
      PIVOT(MAX(SAL)
           FOR JOB IN ('CLERK' AS CLERK,
                      'SALESMAN' AS SALESMAN,
 07
                      'PRESIDENT' AS PRESIDENT,
 08
                      'MANAGER' AS MANAGER,
 09
                      'ANALYST' AS ANALYST)
 11 ORDER BY DEPTNO;
:: 결과 화면
■ DEPTNO CLERK SALESMAN PRESIDENT MANAGER ANALYST
      10 1300
                                   2450
      20 1100
                                   2975
                                           3000
         950
                   1600
                                   2850
```

▶ PIVOT: 행/열 바꾸어 출력

```
DECODE문을 활용하여 PIVOT 함수와 같은 출력 구현하기
    SELECT DEPTNO.
02
           MAX(DECODE(JOB, 'CLERK', SAL)) AS "CLERK",
03
           MAX(DECODE(JOB, 'SALESMAN', SAL)) AS "SALESMAN",
04
           MAX(DECODE(JOB, 'PRESIDENT', SAL)) AS "PRESIDENT",
05
           MAX(DECODE(JOB, 'MANAGER', SAL)) AS "MANAGER",
06
           MAX(DECODE(JOB, 'ANALYST', SAL)) AS "ANALYST"
07
      FROM EMP
08
    GROUP BY DEPTNO
    ORDER BY DEPTNO:
:: 결과 화면
 ■ DEPTNO CLERK SALESMAN PRESIDENT MANAGER ANALYST
          1300
                             5000
                                    2450
       10
       20
          1100
                                    2975
                                            3000
       30
           950
                    1600
                                    2850
```



▶ UNPIVOT : 행/열 바꾸어 출력

```
실습 7-39 UNPIVOT 함수를 사용하여 열로 구분된 그룹을 행으로 출력하기
     SELECT *
02
      FROM(SELECT DEPTNO,
                  MAX(DECODE(JOB, 'CLERK', SAL)) AS "CLERK",
03
04
                  MAX(DECODE(JOB, 'SALESMAN', SAL)) AS "SALESMAN",
05
                  MAX(DECODE(JOB, 'PRESIDENT', SAL)) AS "PRESIDENT",
06
                  MAX(DECODE(JOB, 'MANAGER', SAL)) AS "MANAGER",
07
                  MAX(DECODE(JOB, 'ANALYST', SAL)) AS "ANALYST"
              FROM EMP
08
        GROUP BY DEPTNO
09
10
         ORDER BY DEPTNO)
11
     UNPIVOT(
       SAL FOR JOB IN (CLERK, SALESMAN, PRESIDENT, MANAGER, ANALYST))
12
   ORDER BY DEPTNO, JOB;
```

