



07장(다중행 함수와 데이터 그룹화)

다중행

다중행 함수

- 그룹 함수 or 복수행 함수라고도 불림
- 여러 행을 바탕으로 하나의 결과 값을 도출해 내기위한 함수
- 여러 행이 입력되어 하나의 행으로 결과가 출력됨
- 기본적으로 여러 행이 결과로 나올 수 있는 여을 함께 사용 불가

```
01 SELECT ENAME, SUM(SAL)
02    FROM EMP;
```

:: 결과 화면

Errors
[Error] Execution (1: 8): ORA-00937: 단일 그룹의 그룹 함수가 아닙니다

▼ 자주사용하는 다중행 함수

- SUM() → 더하기 함수

```
SUM([DISTINCT, ALL 중 하나를 선택] '합계를 구할 열 or 연산자 or 함수를 사용한 데이터');
```

- COUNT() → 데이터 갯수를 추출

```
COUNT([DISTINCT, ALL 중 하나를 선택] '갯수를 구할 열 or 연산자 or 함수를 사용한 데이터');
```

- MIN(), MAX() → 최솟값, 최댓값 추출

```
MIN([DISTINCT, ALL 중 하나를 선택] '최소를 구할 열 or 연산자 or 함수를 사용한 데이터');  
MAX([DISTINCT, ALL 중 하나를 선택] '최대를 구할 열 or 연산자 or 함수를 사용한 데이터');
```

- AVG() → 평균 값 추출

```
AVG([DISTINCT, ALL 중 하나를 선택] '평균을 구할 열 or 연산자 or 함수를 사용한 데이터');
```

그룹화

GROUP BY절

- 사용시 유의점
 - 다중행 함수를 사용하지 않은 일반 열은 GROUP BY 절에 명시하지 않으면 SELECT절에 사용 불가

```
SELECT '조회할 컬럼들'  
FROM '조회할 테이블'  
WHERE '조건절'  
GROUP BY '그룹화할 컬럼 지정(다중 가능)'  
ORDER BY '정렬기준';
```

- HAVING 절
 - GROUP BY절에 조건을 줄 때 사용
 - WHERE절과 HAVING절의 쓰임새는 다름!!!
 - WHERE절 → 출력 대상 행을 제한
 - HAVING절 → 그룹화된 대상을 출력에서 제한
 - 실행순서 : WHERE절 → HAVING절

▼ 기본형태

```
SELECT '조회할 컬럼들'  
FROM '조회할 테이블'  
GROUP BY '그룹화할 컬럼'
```

HAVING '출력 그룹을 제한하는 조건식'
ORDER BY '정렬기준';

▼ 예시 코드

실습 7-22 WHERE절을 사용하지 않고 HAVING절만 사용한 경우

```
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)
10	MANAGER	2450
10	PRESIDENT	5000
20	ANALYST	3000
20	MANAGER	2975
30	MANAGER	2850

실습 7-23 WHERE절과 HAVING절을 모두 사용한 경우

```
01 SELECT DEPTNO, JOB, AVG(SAL)
02    FROM EMP
03   WHERE SAL <= 3000
04   GROUP BY DEPTNO, JOB
05   HAVING AVG(SAL) >= 2000
06   ORDER BY DEPTNO, JOB;
```



WHERE절이 GROUP BY절, HAVING절보다
먼저 실행됩니다.

:: 결과 화면

DEPTNO	JOB	AVG(SAL)
10	MANAGER	2450
20	ANALYST	3000
20	MANAGER	2975
30	MANAGER	2850

• GROUP BY절 함수

◦ ROLLUP()

→ 그룹화 할 컬럼을 소그룹 ~ 대그룹 순서로 각 그룹별 결과를 출력하고 마지막에 total 데이터 출력

◦ CUBE()

→ 지정한 모든 열에서 가능한 조합의 결과를 모두 출력

→ n개의 열을 지정하면 2^n 개 조합이 출력

- GROUPING SETS()

→ 지정한 모든 열을 각각 대그룹으로 처리하여 출력

▼ 예시

```
SQL> SELECT
2  DEPTNO,
3  JOB,
4  ENAME,
5  COUNT(*)
6 FROM
7  EMP
8 GROUP BY
9  GROUPING SETS(DEPTNO, JOB, ENAME)
10 ORDER BY
11  DEPTNO,
12  JOB;
```

Select all rows Save as: CSV

DEPTNO	JOB	ENAME	COUNT(*)
10			3
20			5
30			6
	ANALYST		2
	CLERK		4
	MANAGER		3
	PRESIDENT		1
	SALESMAN		4
		TURNER	1
		ADAMS	1
		BLAKE	1
		KING	1
		JAMES	1
		SCOTT	1
		MARTIN	1
		WARD	1
		SMITH	1
		MILLER	1
		CLARK	1
		FORD	1

Page 1 of 2 |< >| (1-20 of 22 rows)

그룹화 함수

- GROUPING()

- 그룹화를 했는지 여부를 반환
- 한번에 한 컬럼만 지정
- 0 → 그룹화 함
- 1 → 그룹화 안함

▼ 기본형태

```
SELECT '조회할 컬럼들'
      GROUPING('GROUP BY절에 ROLLUP 또는 CUBE에 명시한 그룹화 한 컬럼')
FROM '조회할 테이블'
```

```
WHERE '조건절'
GROUP BY ROLLUP( ) or CUBE( )'그룹화할 컬럼 지정'
```

- GROUPING_ID()

- 그룹화를 했는지 여부를 반환
- 한번에 여러 컬럼을 지정 가능

▼ 기본형태

```
SELECT '조회할 컬럼들'
      GROUPING_ID( '그룹화 여부를 확인할 컬럼(여러개 지정가능)' )
FROM '조회할 테이블'
WHERE '조건절'
GROUP BY ROLLUP( ) or CUBE( )'그룹화할 컬럼 지정'
```

LISTAGG()

- 오라클 11g버전부터 사용 가능
- 그룹화 데이터를 하나의 열에 가로로 나열하여 출력

```
SELECT '조회할 컬럼'
      LISTAGG( '나열할 컬럼', [ '데이터 구분자' ] )
      WITHIN GROUP( ORDER BY '정렬기준 컬럼' )
FROM '조회할 테이블'
WHERE '조회할 행 조건식';
```

▼ 예시

실습 7-34 부서별 직원 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
10	KING, CLARK, MILLER
20	FORD, SCOTT, JONES, ADAMS, SMITH
30	BLAKE, ALLEN, TURNER, MARTIN, WARD, JAMES



LISTAGG 함수 사용으로 직원 이름이 훨씬 보기 편해졌습니다.

PIVOT()

- 오라클 11g버전부터 사용 가능
- 기존 테이블 행을 열로 바꿔서 출력

```
SELECT '조회할 컬럼'  
FROM '조회할 테이블'  
PIVOT(  
    FOR '컬럼' IN ( '바꿀 데이터' )  
)  
ORDER BY '정렬 할 컬럼'
```

▼ 예시

실습 7-37 PIVOT 함수를 사용하여 부서별·직책별 최고 급여를 2차원 표 형태로 출력하기

```
01 SELECT *
02   FROM(SELECT JOB, DEPTNO, SAL
03          FROM EMP)
04   PIVOT(MAX(SAL)
05          FOR JOB IN ('CLERK' AS CLERK,
06                     'SALESMAN' AS SALESMAN,
07                     'PRESIDENT' AS PRESIDENT,
08                     'MANAGER' AS MANAGER,
09                     'ANALYST' AS ANALYST)
10          )
11  ORDER BY DEPTNO;
```

:: 결과 화면

DEPTNO	CLERK	SALESMAN	PRESIDENT	MANAGER	ANALYST
10	1300		5000	2450	
20	1100			2975	3000
30	950	1600		2850	

UNPIVOT()

- 오라클 11g버전부터 사용 가능
- 기존 테이블 열을 행으로 바꿔서 출력