

# 이 장에서 다룰 내용



## 1 그룹 함수

---

## 2 GROUP BY 절

---

## 3 HAVING 조건

---

## 01. 그룹 함수



- ❖ 그룹 함수는 하나 이상의 행을 그룹으로 묶어 연산하여 총합, 평균 등 하나의 결과로 나타냅니다.
- ❖ 다음은 그룹 함수의 종류를 정리한 표입니다.

구 분	설 명
SUM	그룹의 누적 합계를 반환합니다.
AVG	그룹의 평균을 반환합니다.
COUNT	그룹의 총 개수를 반환합니다.
MAX	그룹의 최대값을 반환합니다.
MIN	그룹의 최소값을 반환합니다.
STDDEV	그룹의 표준편차를 반환합니다.
VARIANCE	그룹의 분산을 반환합니다.

## 1.1 합계 구하는 SUM 함수



- ❖ SUM 함수는 해당 칼럼 값들에 대한 총합을 구하는 함수입니다.
- ❖ 다음은 급여 총액 구하는 예제입니다.

예

```
SELECT SUM(SAL)
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT SUM(SAL)
2  FROM EMP;

SUM(SAL)
-----
29025

SQL>
```

## 1.1 합계 구하는 SUM 함수



❖ 다음은 커미션 총액 구하는 예제입니다.

예

```
SELECT SUM(COMM)
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger


SQL> SELECT SUM(COMM)
2  FROM EMP;

SUM(COMM)
-----
      2200

SQL>
```

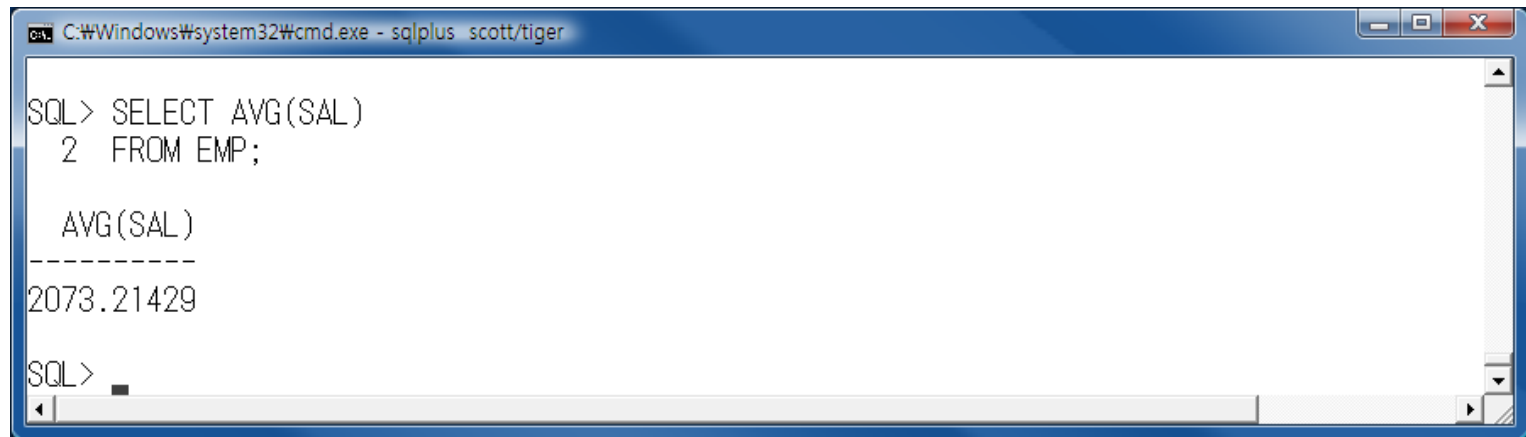
- 사원 테이블을 살펴보면 커미션 컬럼에 NULL 값이 저장된 사원이 존재합니다. NULL 은 블랙 홀이므로 NULL을 저장한 컬럼과 연산한 결과도 NULL 이라고 하였습니다. 커미션의 총합을 구해도 NULL 값으로 출력되지 않습니다. 왜냐하면 그룹 함수는 다른 연산자와는 달리 해당 칼럼 값이 NULL 인 것을 제외하고 계산하기 때문입니다.

## 1.2 평균 구하는 AVG 함수

- ❖ AVG 함수는 해당 칼럼 값들에 대해 평균을 구하는 함수입니다. 이때 해당 칼럼 값이 NULL 인 것에 대해서는 제외하고 계산합니다. 
- ❖ 다음은 급여 평균 구하는 예제입니다.

예

```
SELECT AVG(SAL)
FROM EMP;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT AVG(SAL)
2  FROM EMP;

  AVG(SAL)
-----
2073.21429

SQL>
```

## 1.3 최대값 구하는 MAX, 최소값 구하는 MIN 함수



- ❖ 지정한 칼럼 값들 중에서 최대값을 구하는 함수가 MAX이고, 최소값을 구하는 함수가 MIN입니다.
- ❖ 다음은 가장 높은 급여와 가장 낮은 급여를 구하는 예제입니다.

예

```
SELECT MAX(SAL), MIN(SAL)
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT MAX(SAL), MIN(SAL)
2 FROM EMP;

MAX(SAL)    MIN(SAL)
-----
5000         800

SQL>
```

## 1.4 그룹함수와 단순 컬럼



- ❖ **사원들의 최대 급여는 다음과 같이 구해집니다.**

예

```
SELECT MAX(SAL)
FROM EMP;
```

- ❖ **위 쿼리문의 SELECT 절에 ENAME 컬럼만 추가한다고 될까요?  
아래와 같이 사원의 이름도 함께 출력해 봅니다.**

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT ENAME, MAX(SAL)
      2  FROM EMP;
SELECT ENAME, MAX(SAL)
      *
1행에 오류:
ORA-00937: 단일 그룹의 그룹 함수가 아닙니다

SQL>
```

## <탄탄히 다지기>

1. 가장 최근에 입사한 사원의 입사일과 입사한지 가장 오래된 사원의 입사일을 출력하는 쿼리문을 작성하시오.

```
SCOTT>SELECT MAX(HIREDATE) AS "입사일", MIN(HIREDATE) AS "입사일"  
2      FROM EMP;
```

입사일	입사일
-----	-----
87/07/13	80/12/17

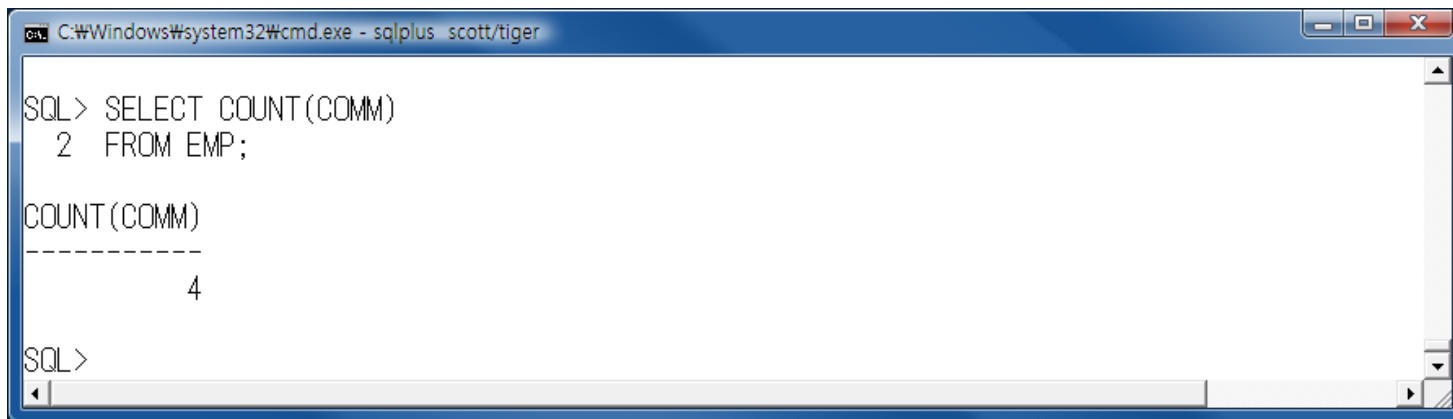


## 1.5 로우 개수 구하는 COUNT 함수

- ❖ COUNT 함수는 테이블에서 조건을 만족하는 행의 개수를 반환하는 함수입니다. COUNT 함수에 특정 칼럼을 기술하면 해당 칼럼 값을 갖고 있는 로우의 개수를 계산하여 되돌립니다.
- ❖ 다음은 사원 테이블의 사원들 중에서 커미션을 받은 사원의 수를 구하는 예제입니다.

예

```
SELECT COUNT(COMM)
FROM EMP;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT COUNT(COMM)
      2  FROM EMP;

COUNT(COMM)
-----
              4

SQL>
```

## 1.5 로우 개수 구하는 COUNT 함수



- ❖ COUNT 함수는 NULL 값에 대해서는 개수를 세지 않습니다.
- ❖ 만일 COUNT 함수에 COUNT(\*)처럼 \*를 적용하면 테이블의 전체 로우 수를 구하게 됩니다.
- ❖ 전체 사원의 수와 커미션을 받는 사원의 수를 구하는 예제입니다.

예

```
SELECT COUNT(*), COUNT(COMM)
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

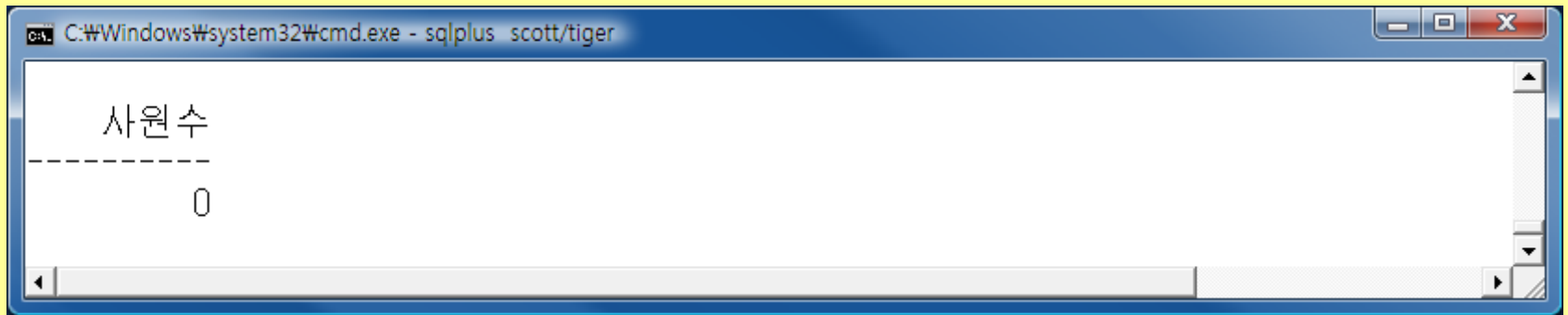
SQL> SELECT COUNT(*), COUNT(COMM)
2  FROM EMP;

COUNT(*) COUNT(COMM)
-----
14          4

SQL>
```

## 탄탄히 다지기

2. 10번 부서 소속 사원중에서 커미션을 받는 사원의 수를 구해보시오.



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The output of the query is displayed in a table format:

사원수
0

## 1.5 로우 개수 구하는 COUNT 함수



- ❖ 이번에는 사원 테이블에서 사원들의 직업의 개수를 카운트 해봅시다.

예

```
SELECT COUNT(JOB) 업무수  
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT COUNT(JOB) 업무수  
2  FROM EMP;  
  
      업무수  
-----  
          14
```

## 1.5 로우 개수 구하는 COUNT 함수



- ❖ 직업의 종류가 몇 개인지 즉, 중복되지 않은 직업의 개수를 카운트 해봅시다.
- ❖ 이 때 중복 행 제거 키워드 DISTINCT를 써서 다음과 같이 질의합니다.

**예**      **SELECT COUNT(DISTINCT JOB) 업무수**  
**FROM EMP;**

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT COUNT(DISTINCT JOB) 업무수
      2  FROM EMP;

      업무수
-----
           5

SQL>
```

## 02. GROUP BY 절



- ❖ 그룹함수를 쓰되 어떤 컬럼 값을 기준으로 그룹함수를 적용할 경우 GROUP BY 절 뒤에 해당 컬럼을 기술하면 됩니다.

형식	SELECT 컬럼명, 그룹함수 FROM 테이블명 WHERE 조건 (연산자) GROUP BY 컬럼명;
----	--

- ❖ 합계, 평균, 최대값.이나, 최소값. 등을 어떤 컬럼을 기준으로 그 컬럼의 값 별로 보고자 할 때 GROUP BY 절 뒤에 해당 컬럼을 기술하면 됩니다.
- ❖ GROUP BY 절을 사용할 때 주의할 점은 GROUP BY 절 다음에는 컬럼의 별칭을 사용할 수 없고, 반드시 컬럼명을 기술해야 한다는 점입니다.

## 02. GROUP BY 절



❖ 사원 테이블을 부서 번호로 그룹지어 봅시다.

예

```
SELECT DEPTNO  
FROM EMP  
GROUP BY DEPTNO;
```

The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered and executed:

```
SQL> SELECT DEPTNO  
2 FROM EMP  
3 GROUP BY DEPTNO;
```

The output of the query is displayed as follows:

DEPTNO
30
20
10

The window also shows the prompt "SQL>" at the bottom, indicating the command is ready for the next input.

## 02. GROUP BY 절



❖ 다음은 소속 부서별 평균 급여 구하는 예제입니다.

예

```
SELECT DEPTNO, AVG(SAL)
FROM EMP
GROUP BY DEPTNO;
```

The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The prompt is "SQL>". The user has entered the following SQL query:

```
SQL> SELECT DEPTNO, AVG(SAL)
2  FROM EMP
3  GROUP BY DEPTNO;
```

The output of the query is displayed as a table with two columns: DEPTNO and AVG(SAL). The table has three rows of data:

DEPTNO	AVG(SAL)
30	1566.66667
20	2175
10	2916.66667

The prompt "SQL>" is visible at the bottom left of the window.



## 02. GROUP BY 절

❖ 다음은 소속 부서별 최대 급여와 최소 급여를 구하는 예제입니다.



예

```
SELECT DEPTNO, MAX(SAL), MIN(SAL)
FROM EMP
GROUP BY DEPTNO;
```

The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The SQL prompt "SQL>" is followed by the query: "SELECT DEPTNO, MAX(SAL), MIN(SAL) FROM EMP GROUP BY DEPTNO;". The output is a table with three columns: DEPTNO, MAX(SAL), and MIN(SAL). The data is as follows:

DEPTNO	MAX(SAL)	MIN(SAL)
30	2850	950
20	3000	800
10	5000	1300

The prompt "SQL>" is visible at the bottom of the window.

## 02. GROUP BY 절



예

다음은 부서별로 사원 수와 커미션을 받는 사원들의 수를 계산하는 예제입니다.

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT DEPTNO, COUNT(*), COUNT(COMM)
2  FROM EMP
3  GROUP BY DEPTNO;

   DEPTNO   COUNT(*)  COUNT(COMM)
-----
       30         6         4
       20         5         0
       10         3         0

SQL>
```

## 탄탄히 다지기

3. 특정 칼럼을 기준으로 테이블에 존재하는 행들을 그룹별로 구분하기 위해  
서는\_\_\_\_\_사용합니다.

### 03. HAVING 조건



- ❖ SELECT 절에 조건을 사용하여 결과를 제한할 때는 WHERE 절을 사용하지만 그룹의 결과를 제한할 때는 HAVING 절을 사용합니다.
- ❖ 예를 들어 설명하자면 부서별로 그룹 지은 후(GROUP BY), 그룹 지어진 부서별 평균 급여가 2000 이상인(HAVING) 부서번호와 부서별 평균 급여를 출력하는 경우입니다.

예

```
SELECT DEPTNO, AVG(SAL)
FROM EMP
GROUP BY DEPTNO
HAVING AVG(SAL) >= 2000;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT DEPTNO, AVG(SAL)
2 FROM EMP
3 GROUP BY DEPTNO
4 HAVING AVG(SAL) >= 2000;

   DEPTNO    AVG(SAL)
-----
        20         2175
        10  2916.66667

SQL>
```

### 03. HAVING 조건



- ❖ 부서별 급여의 최대값과 최소값을 구하되 최대 급여가 2900초과인 부서만 출력합니다.

예

```
SELECT DEPTNO, MAX(SAL), MIN(SAL)
FROM EMP
GROUP BY DEPTNO
HAVING MAX(SAL) > 2900;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT DEPTNO, MAX(SAL), MIN(SAL)
2  FROM EMP
3  GROUP BY DEPTNO
4  HAVING MAX(SAL) > 2900;

   DEPTNO   MAX(SAL)   MIN(SAL)
-----
        20        3000         800
        10        5000        1300

SQL>
```

## <탄탄히 다지기>

4. GROUP BY 절에 특정 조건을 기술하기 위한 ① \_\_\_\_\_  
절을 사용합니다.

## 05. ROLLUP – 부서별 인원수와 총계를 구합니다.

예

```
SELECT DEPTNO, COUNT(DEPTNO)
FROM EMP
GROUP BY DEPTNO
ORDER BY DEPTNO;
```

DEPTNO	COUNT(DEPTNO)
10	3
20	5
30	6

DEPTNO	COUNT(DEPTNO)
10	3
20	5
30	6
	14

예

```
SELECT DEPTNO, COUNT(DEPTNO)
FROM EMP
GROUP BY ROLLUP(DEPTNO)
ORDER BY DEPTNO;
```

GROUP BY ROLLUP(expr1)

expr1

총계

## 05. GROUP BY – 부서와 직업별 인원수 구하기

예

```
SELECT DEPTNO, JOB, COUNT(*)  
FROM EMP  
GROUP BY DEPTNO, JOB  
ORDER BY DEPTNO;
```

DEPTNO	JOB	COUNT(*)
10	CLERK	1
10	MANAGER	1
10	PRESIDENT	1
20	ANALYST	2
20	CLERK	2
20	MANAGER	1
30	CLERK	1
30	MANAGER	1
30	SALESMAN	4



## 05. ROLLUP – 부서와 직업별 인원수, 부서별 인원수, 총계를 구합니다.

예

```
SELECT DEPTNO, JOB, COUNT(*)
FROM EMP
GROUP BY ROLLUP(DEPTNO, JOB)
ORDER BY DEPTNO;
```

DEPTNO	JOB	COUNT(*)	
10	CLERK	1	DEPTNO와 JOB 집계
10	MANAGER	1	DEPTNO와 JOB 집계
10	PRESIDENT	1	DEPTNO와 JOB 집계
10		3	DEPTNO 집계
20	ANALYST	2	DEPTNO와 JOB 집계
20	CLERK	2	DEPTNO와 JOB 집계
20	MANAGER	1	DEPTNO와 JOB 집계
20		5	DEPTNO 집계
30	CLERK	1	DEPTNO와 JOB 집계
30	MANAGER	1	DEPTNO와 JOB 집계
30	SALESMAN	4	DEPTNO와 JOB 집계
30		6	DEPTNO 집계
		14	총계

GROUP BY ROLLUP(expr1, expr2)

expr1 + expr2

expr1

총계

## 05. ROLLUP -부서와 직업 별 인원수, 직업별 인원수 총계를 구합니다.

예

```
SELECT DEPTNO, JOB, COUNT(JOB)
FROM EMP
GROUP BY ROLLUP(JOB, DEPTNO)
ORDER BY JOB;
```

DEPTNO	JOB	COUNT(JOB)	
20	ANALYST	2	JOB와 DEPTNO 집계
	ANALYST	2	JOB 집계
10	CLERK	1	JOB와 DEPTNO 집계
20	CLERK	2	JOB와 DEPTNO 집계
30	CLERK	1	JOB와 DEPTNO 집계
	CLERK	4	JOB 집계
10	MANAGER	1	JOB와 DEPTNO 집계
20	MANAGER	1	JOB와 DEPTNO 집계
30	MANAGER	1	JOB와 DEPTNO 집계
	MANAGER	3	JOB 집계
10	PRESIDENT	1	JOB와 DEPTNO 집계
	PRESIDENT	1	JOB 집계
30	SALESMAN	4	JOB와 DEPTNO 집계
	SALESMAN	4	JOB 집계
		14	총계

GROUP BY ROLLUP(expr1, expr2)

expr1 + expr2

expr1

총계

# CUBE – 모든 조합별로 집계한 결과를 반환한다.

예

```
SELECT DEPTNO, JOB, COUNT(JOB)
FROM EMP
GROUP BY CUBE(DEPTNO, JOB)
ORDER BY DEPTNO;
```

DEPTNO	JOB	COUNT(JOB)
10	CLERK	1
10	MANAGER	1
10	PRESIDENT	1
10		3
20	ANALYST	2
20	CLERK	2
20	MANAGER	1
20		5
30	CLERK	1
30	MANAGER	1
30	SALESMAN	4
30		6
	ANALYST	2
	CLERK	4
	MANAGER	3
	PRESIDENT	1
	SALESMAN	4
		14

DEPTNO와 JOB 집계  
 DEPTNO와 JOB 집계  
 DEPTNO와 JOB 집계  
 DEPTNO 집계  
 DEPTNO와 JOB 집계  
 DEPTNO와 JOB 집계  
 DEPTNO와 JOB 집계  
 DEPTNO 집계  
 DEPTNO와 JOB 집계  
 DEPTNO와 JOB 집계  
 DEPTNO와 JOB 집계  
 DEPTNO 집계  
 JOB 집계  
 JOB 집계  
 JOB 집계  
 JOB 집계  
 JOB 집계  
 총계

CUBE (expr1, expr2)	expr1 + expr2
	expr1
	expr2
	전체

## 06. LISTAGG..WITHIN GROUP 함수 사용 전

예

```
SELECT DEPTNO, ENAME  
FROM EMP  
GROUP BY DEPTNO
```

```
SCOTT>SELECT DEPTNO, ENAME  
2 FROM EMP  
3 GROUP BY DEPTNO;  
SELECT DEPTNO, ENAME  
*
```

```
ERROR at line 1:  
ORA-00979: not a GROUP BY  
expression
```

GROUP BY절에서 선언하지 않은 컬럼을 사용해서 오류 발생했습니다.

## 06. LISTAGG..WITHIN GROUP 함수

Aggregation은 여러 행의 값을 그룹핑하고 결합시키는 것입니다.  
LISTAGG는 그룹핑 컬럼이 아닌 다른 컬럼 값을 모두 나열해 줍니다.

형  
식

LISTAGG(컬럼명, 구분자) WITHIN GROUP (ORDER BY 정렬할 컬럼)

예

```
SELECT DEPTNO,  
LISTAGG(ENAME,',') WITHIN GROUP(ORDER BY ENAME) "LISTAGG"  
FROM EMP  
GROUP BY DEPTNO;
```

DEPTNO LISTAGG

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

## 06. LISTAGG..WITHIN GROUP 함수

Aggregation은 여러 행의 값을 그룹핑하고 결합시키는 것입니다.  
LISTAGG는 그룹핑 컬럼이 아닌 다른 컬럼 값을 모두 나열해 줍니다.

형식  
LISTAGG(컬럼명, 구분자) WITHIN GROUP (ORDER BY 정렬할 컬럼)

예  
SELECT DEPTNO,  
LISTAGG(ENAME,',') WITHIN GROUP(ORDER BY JOB) "LISTAGG"  
FROM EMP  
GROUP BY DEPTNO;

DEPTNO LISTAGG

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