

6. 그룹 함수(1) [Group Function의 종류]

6.1 Group Function의 종류

** 그룹함수란? **

- 그룹함수란 여러 행 또는 테이블 전체의 행에 대해 함수가 적용되어 하나의 결과값을 가져오는 함수를 말합니다.
- GROUP BY절을 이용하여 그룹 당 하나의 결과가 주어지도록 그룹화 할 수 있습니다.
- HAVING절을 사용하여 그룹 함수를 가지고 조건비교를 할 수 있습니다.
- COUNT(*)를 제외한 모든 그룹함수는 NULL값을 고려하지 않습니다.
- MIN, MAX 그룹함수는 모든 자료형에 대해서 사용 할 수 있습니다.

** COUNT **

COUNT 함수는 검색된 행의 수를 반환 합니다.

```
SQL> SELECT COUNT(deptno) FROM DEPT;
```

```
          COUNT(DEPTNO)
          -----
                4
검색된 행의 총 수 4개를 반환합니다. 즉 4개의 부서가 존재합니다.
```

** MAX **

MAX함수는 컬럼중의 최대값을 반환 합니다.

```
SQL> SELECT MAX(sal) salary FROM emp;
```

```
          SALARY
          -----
          5000
sal컬럼중에서 제일 큰값을 반환합니다. 즉 가장 큰 급여를 반환합니다.
```

** MIN **

MIN 함수는 컬럼중의 최소값을 반환 합니다.

```
SQL> SELECT MIN(sal) salary FROM emp;
```

```
          SALARY
          -----
           800
```

sal컬럼중에서 가장 작은 값 반환합니다. 즉 가장 적은 급여를 반환합니다

**** AVG ****

AVG함수는 평균값을 반환 합니다.

```
SQL> SELECT ROUND(AVG(sal),1) salary FROM emp WHERE deptno = 30
```

```
      SALARY
-----
      1566.7
```

30부서 사원의 평균 급여를 소수점 1자리 이하에서 반올림해서 보여줍니다.

**** SUM ****

SUM함수는 검색된 컬럼의 합을 반환 합니다.

```
SQL> SELECT SUM(sal) salary FROM emp WHERE deptno = 30;
```

```
      SALARY
-----
      9400
```

30부서 사원의 급여 합계를 보여줍니다.

**** STDDEV ****

STDDEV함수는 표준편차를 반환 합니다.

```
SQL> SELECT ROUND(STDDEV(sal),3) salary FROM emp WHERE deptno = 30;
```

```
      SALARY
-----
      668.331
```

30부서 사원의 급여 표준편차를 반환합니다.

6.1.1 ROLLUP operator

- ROLLUP 구문은 GROUP BY절과 같이 사용 되며, GROUP BY절에 의해서 그룹 지어진 집합결과에 대해서 좀 더 상세한 정보를 반환하는 기능을 수행 합니다
- SELECT절 ROLLUP을 사용함으로써 보통의 SELECT된 데이터와 그 데이터의 총계를 구할 수 있습니다.

※ 우선 아주 간단한 예제부터 살펴 보겠습니다.

(scott유저의 emp테이블을 가지고 테스트 했습니다.)

-- Group BY를 사용해서 직업별로 급여 합계를 구하는 예제 입니다.

```
SELECT job, SUM(sal)
FROM emp
GROUP BY job
```

JOB	SUM(SAL)
ANALYST	600
CLERK	3200
MANAGER	33925
PRESIDENT	5000
SALESMAN	4000

-- 아래 SQL문은 위의 예제에 ROLLUP구문을 사용해서 직업별로 급여 합계를 구하고 총계를 구하는 예제 입니다.

```
SELECT job, SUM(sal)
FROM emp
GROUP BY ROLLUP(job)
```

JOB	SUM(SAL)
ANALYST	6000
CLERK	3200
MANAGER	33925
PRESIDENT	5000
SALESMAN	4000
	52125

--> 급여 합계에 대한 총계가 추가 되었습니다.

우선 간단하게 ROLLUP Operator의 예제를 살펴보았습니다.

조금 더 복잡한(?) 예제를 하나더 해보겠습니다.

-- 부서의 직업별로 인원수와 급여 합계를 구하는 예제를 하나 더 해보겠습니다.

-- 일반적인 Group By절을 사용해서 SQL문을 구현해보면, 아래와 같이 되겠습니다.

```
SELECT b.dname, a.job, SUM(a.sal) sal, COUNT(a.empno) emp_count
FROM emp a, dept b
WHERE a.deptno = b.deptno
GROUP BY b.dname, a.job
```

DNAME	JOB	SAL	EMP_COUNT
ACCOUNTING	CLERK	1300	1
ACCOUNTING	MANAGER	2450	1
ACCOUNTING	PRESIDENT	5000	1
RESEARCH	ANALYST	6000	2
RESEARCH	CLERK	1900	2
RESEARCH	MANAGER	2975	1
SALES	MANAGER	28500	1
SALES	SALESMAN	4000	3

-- 결과를 보면, 부서별로 인원이 몇명이고, 또 급여합계가 얼마가 되는지 한눈에 보이지 않죠..

일일이 부서에 해당하는 직업별 급여와 인원수를 일일이 더해야 되죠.

-- 이런 경우 ROLLUP Operator를 적용해서 구현을 하면은 부서별 급여합계와 인원 총수를 쉽게 볼 수 있습니다.

```
SQL> SELECT b.dname, a.job, SUM(a.sal) sal, COUNT(a.empno) emp_count
      FROM emp a, dept b
      WHERE a.deptno = b.deptno
      GROUP BY ROLLUP(b.dname, a.job)
```

DNAME	JOB	SAL	EMP_COUNT	
ACCOUNTING	CLERK	1300	1	
ACCOUNTING	MANAGER	2450	1	
ACCOUNTING	PRESIDENT	5000	1	
ACCOUNTING		8750	3	--> ACCOUNTING 부서의 급여합계와 전체 인원수..
RESEARCH	ANALYST	6000	2	
RESEARCH	CLERK	1900	2	
RESEARCH	MANAGER	2975	1	
RESEARCH		10875	5	--> RESEARCH 부서의 급여합계와 전체 인원수..
SALES	MANAGER	28500	1	
SALES	SALESMAN	4000	3	
SALES		32500	4	--> SALES부서의 급여합계와 전체 인원수..
		52125	12	-> 전체 급여 합계와 전체 인원수

위와 같이 ROLLUP Operator을 일반적인 누적에 대한 총계를 구할때 사용하면 아주 편리하게 사용 할 수 있습니다.

6.1.2 CUBE Operator

ROLLUP 강좌예제 중에서 아래 SQL문 예제를 가지고 CUBE강좌를 진행 하려 합니다.

```
===== ROLLUP 강좌의 예제입니다. =====

SELECT b.dname, a.job, SUM(a.sal) sal, COUNT(a.empno) emp_count
FROM emp a, dept b
WHERE a.deptno = b.deptno
GROUP BY ROLLUP(b.dname, a.job)
```

DNAME	JOB	SAL	EMP_COUNT
ACCOUNTING	CLERK	1300	1
ACCOUNTING	MANAGER	2450	1
ACCOUNTING	PRESIDENT	5000	1
ACCOUNTING		8750	3 --> ACCOUNTING 부서의 급여합계와 전체 사원수..
RESEARCH	ANALYST	6000	2
RESEARCH	CLERK	1900	2
RESEARCH	MANAGER	2975	1
RESEARCH		10875	5 --> RESEARCH 부서의 급여합계와 전체 사원 수..
SALES	MANAGER	28500	1
SALES	SALESMAN	4000	3
SALES		32500	4 --> SALES부서의 급여합계와 전체 사원수..
			52125 12 -> 전체 급여 합계와 전체 사원수

```
=====
```

위의 SQL의 결과를 보면 부서별로 각 직업에 해당하는 급여와 사원수를 볼 수 있습니다.

하지만 부서별로 각 직업의 급여와 사원수, 그리고 또 각 직업별로 급여 합계와 사원수를 보기 위해서는 두개의 ROLLUP을 사용해서 SQL문을 작성해야 합니다.

```
SQL> SELECT b.dname, a.job, SUM(a.sal) sal, COUNT(a.empno)emp_count
      FROM emp a, dept b
      WHERE a.deptno = b.deptno
      GROUP BY ROLLUP(b.dname, a.job)
      UNION
      SELECT ' ', job, SUM(sal) sal, COUNT(empno) emp_count
      FROM emp
      GROUP BY ROLLUP(job)
```

DNAME	JOB	SAL	EMP_COUNT
ACCOUNTING	CLERK	1300	1
ACCOUNTING	MANAGER	2450	1
ACCOUNTING	PRESIDENT	5000	1
ACCOUNTING		8750	3
RESEARCH	ANALYST	6000	2
RESEARCH	CLERK	1900	2
RESEARCH	MANAGER	2975	1
RESEARCH		10875	5
SALES	MANAGER	28500	1
SALES	SALESMAN	4000	3
SALES		32500	4
구하고요			
	ANALYST	6000	2
	CLERK	3200	3
	MANAGER	33925	3
	PRESIDENT	5000	1
	SALESMAN	4000	3
		52125	12

=> 요기 까지는 첫 번째 ROLLUP를 이용해서

했습니다.

CUBE Operator를 사용하면 편하게 하나의 SQL문으로 위의 결과를 얻을 수 있습니다.

직접 SQL문을 실행시켜 보면 쉽게 이해가 갑니다.

```
SQL> SELECT b.dname, a.job, SUM(a.sal) sal, COUNT(a.empno) emp_count
      FROM emp a, dept b
      WHERE a.deptno = b.deptno
      GROUP BY CUBR(b.dname, a.job)
```

DNAME	JOB	SAL	EMP_COUNT
ACCOUNTING	CLERK	1300	1
ACCOUNTING	MANAGER	2450	1
ACCOUNTING	PRESIDENT	5000	1
ACCOUNTING		8750	3
RESEARCH	ANALYST	6000	2
RESEARCH	CLERK	1900	2
RESEARCH	MANAGER	2975	1
RESEARCH		10875	5
SALES	MANAGER	28500	1
SALES	SALESMAN	4000	3
SALES		32500	4
	ANALYST	6000	2
	CLERK	3200	3
	MANAGER	33925	3
	PRESIDENT	5000	1
	SALESMAN	4000	3
		52125	12

=>ACCOUNTING 부서의 직업별 급여의 총계와 사원수.

=>RESEARCH 부서의 직업별 급여의 총계와 사원수.

=>SALES 부서의 직업별 급여 총계와 사원수.

=> 직업별로 급여의 총계와 사원수를 보여줍니다.

CUBE를 어느 경우에 사용 할 수 있는지 이해 되셨죠..

CUBE Operator는 Cross-Tab에 대한 Summary를 추출 하는데 사용 됩니다.

즉 ROLLUP에 의해 나타내어지는 Item Total값과 Column Total값을 나타 낼 수 있습니다.

**** GROUPING() 함수 ****

GROUPING Function은 ROLLUP, CUBE Operator에 모두 사용할 수 있습니다.

GROUPING Function는 해당 Row가 GROUP BY에 의해서 산출된 Row인 경우에는 0을 반환 하고, ROLLUP이나 CUBE에 의해서 산출된 Row인 경우에는 1을 반환하게 됩니다.

따라서 해당 Row가 결과집합에 의해 산출된 Data인지,

ROLLUP이나 CUBE에 의해서 산출된 Data인지를 알 수 있도록 지원하는 함수입니다.

```
SQL> SELECT b.dname, a.job, SUM(a.sal) sal, COUNT(a.empno) emp_count,
        GROUPING(b.dname)"D", GROUPING(a.job)"S"
        FROM emp a, dept b
        WHERE a.deptno = b.deptno
        GROUP BY CUBE(b.dname, a.job)
```

DNAME	JOB	SAL	EMP_COUNT	D	S
ACCOUNTING	CLERK	1300	1	0	0
ACCOUNTING	MANAGER	2450	1	0	0
ACCOUNTING	PRESIDENT	5000	1	0	0
ACCOUNTING		8750	3	0	1
RESEARCH	ANALYST	6000	2	0	0
RESEARCH	CLERK	1900	2	0	0
RESEARCH	MANAGER	2975	1	0	0
RESEARCH		10875	5	0	1
SALES	MANAGER	28500	1	0	0
SALES	SALESMAN	4000	3	0	0
SALES		32500	4	0	1
	ANALYST	6000	2	1	0
	CLERK	3200	3	1	0
	MANAGER	33925	3	1	0
	PRESIDENT	5000	1	1	0
	SALESMAN	4000	3	1	0