**[문제1] PT3-67, emp 테이블에서 사원 중에 급여(sal)와 보너스(comm)를 합친 금액이 가장 많은 경우와 가장 적은 경우, 평균 금액을 구하도록  
   SQL 문을 실습하라. 단 보너스가 없을 경우는 보너스를 0 으로 계산하고 출력 금액은 모두 소수점 첫째 자리까지만 나오게 한다.**

SQL> col c\_max for 9999

SQL> col c\_min for 9999

SQL> col c\_avg for 9999.9

SQL> select \* from (select sal, comm, job from emp)

2 pivot

3 (

4 MAX(sal+NVL(comm,0)) AS MAX,

5 MIN(sal+NVL(comm,0)) AS MIN,

6 AVG(sal+NVL(comm,0)) AS AVG FOR job in('CLERK' AS "C"));

C\_MAX C\_MIN C\_AVG

----- ----- -------

1300 800 1037.5

**[실행결과]**

원래는 일반적으로 group by절을 단순하게 사용하여 구해도 되는 결과이나, 전에 배운 pivot에 대해서 상기시키고자 pivot 문을 적용하여 풀었습니다.

여기서 새로 알았던 점은 pivot을 사용하면서 round를 사용하려고 했는데 aggreate function은 사용하지 못 한다는 점을 알았습니다. 교수님께서 알려주셔서 시간 끌지 않고 이 문제는 빨리 풀 수 있었습니다.

그리고 문제의 의도가 애매한 것같다고 느꼈습니다. 분명 평사원은 CLARK 라고 알고 있었는데 모든 직책이 사원이라고 하셔서 교수님이랑 눈 마주치고 웃었던 것 같습니다. 혹시 나중에라도 Pivot을 써먹을 일이 있으면 까먹지 않고 쓸 것 같습니다.

**[문제2] PT3-68, emp 테이블에서 입사일 컬럼을 참조해서 예제와 같이 월별로 입사월의 인원수를 출력하는 SQL 문을 실습하라.**

SQL> col jan for a4

SQL> col feb for a4

SQL> col mar for a4

SQL> col apr for a4

SQL> col may for a4

SQL> col jun for a4

SQL> col jul for a4

SQL> col aug for a4

SQL> col sep for a4

SQL> col oct for a4

SQL> col nov for a4

SQL> col dec for a4

SQL> col total for a4

SQL> select

2 to\_char(count(ename))||'EA' "TOTAL",

3 count(decode(HIRE,'JAN',1))||'EA' "JAN",

4 count(decode(HIRE,'FEB',2))||'EA' "FEB",

5 count(decode(HIRE,'MAR',3))||'EA' "MAR",

6 count(decode(HIRE,'APR',4))||'EA' "APR",

7 count(decode(HIRE,'MAY',5))||'EA' "MAY",

8 count(decode(HIRE,'JUN',6))||'EA' "JUN",

9 count(decode(HIRE,'JUL',7))||'EA' "JUL",

10 count(decode(HIRE,'AUG',8))||'EA' "AUG",

11 count(decode(HIRE,'SEP',9))||'EA' "SEP",

12 count(decode(HIRE,'OCT',10))||'EA' "OCT",

13 count(decode(HIRE,'NOV',11))||'EA' "NOV",

14 count(decode(HIRE,'DEC',12))||'EA' "DEC"

15 from (select SUBSTR(HIREDATE,4,3) HIRE, ENAME from emp);

TOTA JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC

---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----

16EA 2EA 3EA 2EA 0EA 1EA 1EA 0EA 1EA 2EA 0EA 1EA 1EA

**[실행결과]**

Pivot을 쓰고 싶었는데 decode로 원시적인(?)방법도 사용해보고 싶어서 사용했습니다.(이상 없음)

**[문제11] PT3-71, emp 테이블에서 직원들의 급여와 전체 급여의 누적 급여금액이 내용과 같이 출력하도록 SQL 문을 실습하라.  
   단 급여를 오름차순으로 정렬해서 출력하라.**

SQL> col total for 99999

SQL> select deptno, ename, sal, sum(sal) over(order by sal) "TOTAL" from emp;

DEPTNO ENAME SAL TOTAL

---------- ---------- ---------- ------

20 SMITH 800 800

30 JAMES 950 1750

20 ADAMS 1100 2850

30 WARD 1250 5350

30 MARTIN 1250 5350

10 MILLER 1300 6650

30 TURNER 1500 8150

30 ALLEN 1600 9750

10 CLARK 2450 12200

30 BLAKE 2850 15050

20 JONES 2975 18025

20 SCOTT 3000 27025

30 Cat 3000 27025

20 FORD 3000 27025

10 Tiger 3600 30625

10 KING 5000 35625

**[실행결과]**

Sum(sal) 을 사용함으로써 합계를 구하고 그 누적 합계 over를 이용해서 그안에 order by sal로 sal을 기준으로 오름차순 정렬을 하면서 total 열에 누적의 합계를 전체 기준 -> sal의 합계 오름차순을 추가하여서 하였음.

해당 이미지는 인자가 비슷해서 넣어서 참고 하려는 이미지 입니다. (over 절)

텍스트이(가) 표시된 사진

자동 생성된 설명

**[문제3] PT3-73, emp 테이블에서 부서번호 컬럼을 사용하여 부서별 인원수와 전체 인원수 대비 차지하는 비율을 출력하도록 SQL 문을 실습하라.**

SQL> select deptno, count(deptno), ROUND(RATIO\_TO\_REPORT(count(deptno)) OVER()\*100,2) "RATIO" from emp group by deptno;

DEPTNO COUNT(DEPTNO) RATIO

---------- ------------- ----------

30 7 43.75

20 5 31.25

10 4 25

**[실행결과]**

여기서 중요한 문구는 RATIO\_TO\_REPORT 를 적어주게 되면 groupby절의 deptno 로 그룹화로 인해 count를 deptno로 인자로 넣어주게 되면 그룹별 deptno 개수로 출력이 되는데 이것을 인자로 넣으면 그룹별 개수가 전체 개수의 비중을 몇퍼센트 차지하는지 알 수 있습니다.

처음 .1265564 이런식으로 나와서 100을 곱하고 round 함수로 소수 두번째 자리에서 반올림 한 결과를 나타내었습니다.

**[문제4] PT3-74, emp 테이블에서 내용과 같이 부서별로 급여 누적 합계가 나오도록 출력하는 SQL 문을 실습하라. 단 부서번호로 오름차순 출력하라.**

SQL> select deptno, ename, sal, sum(sal) over(partition by deptno order by sal) as "TOTAL" from emp;

DEPTNO ENAME SAL TOTAL

---------- ---------- ---------- ------

10 MILLER 1300 1300

10 CLARK 2450 3750

10 Tiger 3600 7350

10 KING 5000 12350

20 SMITH 800 800

20 ADAMS 1100 1900

20 JONES 2975 4875

20 FORD 3000 10875

20 SCOTT 3000 10875

30 JAMES 950 950

30 WARD 1250 3450

30 MARTIN 1250 3450

30 TURNER 1500 4950

30 ALLEN 1600 6550

30 BLAKE 2850 9400

30 Cat 3000 12400

**[실행결과]**

여기서도 이전에 말했던 것인데 처음에는 헷갈려서 ~별로 라고하면 일단 먼저 group by를 하려고 했는데 뭔가 이상해서 deptno와 ename 컬럼을 확인했더니 다 나와있는 것을 확인하고 group by가 아니라 누적합계를 ~별로 하라는 뜻이라서 아차 하고 partition by를 떠올렸습니다.

인상깊게 남았던 문제라서 이것도 다시는 안까먹을 것 같습니다.

**[문제5] PT3-75, emp 테이블을 사용하여 아래와 같이 각 사원의 급여액이 전체 직원 급여총액에서 몇 %의 비율을 차지하는지 출력하는  
   SQL 문을 실습하라.. 단 급여 비중이 높은 사람이 먼저 출력하라.**

SQL> col % for 99.99

SQL> select deptno, ename, sal, sum(sal) over() "TOTAL\_SAL" , ROUND(RATIO\_TO\_REPORT(SAL) OVER()\*100,2) "%" from emp order by sal desc;

DEPTNO ENAME SAL TOTAL\_SAL %

---------- ---------- ---------- ---------- ------

10 KING 5000 35625 14.04

10 Tiger 3600 35625 10.11

30 Cat 3000 35625 8.42

20 SCOTT 3000 35625 8.42

20 FORD 3000 35625 8.42

20 JONES 2975 35625 8.35

30 BLAKE 2850 35625 8.00

10 CLARK 2450 35625 6.88

30 ALLEN 1600 35625 4.49

30 TURNER 1500 35625 4.21

10 MILLER 1300 35625 3.65

30 WARD 1250 35625 3.51

30 MARTIN 1250 35625 3.51

20 ADAMS 1100 35625 3.09

30 JAMES 950 35625 2.67

20 SMITH 800 35625 2.25

**[실행결과]**

앞의 내용을 종합해 sum – over과 ratio to report – over 절을 이용해서 잘 종합하여 출력했습니다.(이상 없음)

**[문제6] PT3-76, emp 테이블을 조회하여 아래와 같이 각 직원들의 급여가 해당 부서 합계금액에서 몇 %의 비중을 차지하는지를 출력하는  
   SQL 문을 실습하라. 단 부서번호를 기준으로 오름차순으로 출력하라.**

SQL> select deptno, ename, sal, sum(sal) over(partition by deptno) "SUM\_DEPTNO", ROUND(RATIO\_TO\_REPORT(sal) OVER(partition by deptno)\*100,2) "%" from emp;

DEPTNO ENAME SAL SUM\_DEPTNO %

---------- ---------- ---------- ---------- ------

10 KING 5000 12350 40.49

10 MILLER 1300 12350 10.53

10 Tiger 3600 12350 29.15

10 CLARK 2450 12350 19.84

20 SMITH 800 10875 7.36

20 JONES 2975 10875 27.36

20 ADAMS 1100 10875 10.11

20 FORD 3000 10875 27.59

20 SCOTT 3000 10875 27.59

30 JAMES 950 12400 7.66

30 TURNER 1500 12400 12.10

30 MARTIN 1250 12400 10.08

30 WARD 1250 12400 10.08

30 ALLEN 1600 12400 12.90

30 Cat 3000 12400 24.19

30 BLAKE 2850 12400 22.98

**[실행결과]**

전 문제의 이해과정과 동일.

**[문제7] PT4-7, 테이블 생성 및 입력(cat\_a, cat\_b, cat\_c)**

SQL> CREATE TABLE cat\_a(no NUMBER, name VARCHAR2(1));

Table created.

SQL> INSERT INTO cat\_a VALUES(1, 'A');

1 row created.

SQL> INSERT INTO cat\_a VALUES(2, 'B');

1 row created.

SQL> CREATE TABLE cat\_b(no NUMBER, name VARCHAR2(1));

Table created.

SQL> INSERT INTO cat\_b VALUES(1, 'C');

1 row created.

SQL> INSERT INTO cat\_b VALUES(2, 'D');

1 row created.

SQL> CREATE TABLE cat\_c(no NUMBER, name VARCHAR2(1));

Table created.

SQL> INSERT INTO cat\_c VALUES(1, 'E');

1 row created.

SQL> INSERT INTO cat\_c VALUES(2, 'F');

1 row created.

**[실행결과]**

간단하게 테이블을 만들으라는 문제였습니다.

테이블이 이미 존재하는게 있는데 데이터가 예제와 맞지 않아서 다 drop table로 날려버리고 다시 만들었습니다.

**[문제8] PT4-9, 2가지 유형의 키티션 곱 저굥하기**

SQL> select a.name, b.name from cat\_a a, cat\_b b where a.no=b.no;

N N

- -

A C

B D

SQL> select a.name, b.name from cat\_a a, cat\_b b;

N N

- -

A C

A D

B C

B D

**[실행결과]**

두번째 방식으로 하면 카티션이 2x2로 중복으로 나온다는 것을 알았습니다.

**[문제9] PT4-10, step4, step5 실행하기**

SQL> select a.name, b.name, c.name from cat\_a a, cat\_b b, cat\_c c where a.no = b.no AND a.no = c.no;

N N N

- - -

A C E

B D F

**[문제10] PT4-12, step 3 테스트 하기**

SQL> select \* from (select empno, ename, job, sal from emp where deptno=10),(select level c1 from dual connect by level <=3);

EMPNO ENAME JOB SAL C1

---------- ---------- --------- ---------- ----------

7839 KING PRESIDENT 5000 1

7782 CLARK MANAGER 2450 1

7934 MILLER CLERK 1300 1

1000 Tiger 3600 1

7839 KING PRESIDENT 5000 2

7782 CLARK MANAGER 2450 2

7934 MILLER CLERK 1300 2

1000 Tiger 3600 2

7839 KING PRESIDENT 5000 3

7782 CLARK MANAGER 2450 3

7934 MILLER CLERK 1300 3

1000 Tiger 3600 3

**[실행결과]**

**[ 카티션 곱을 사용하는 이유 ]**

**첫째, 데이터를 복제해서 원본 테이블을 반복해서 읽는 것을 피하기 위해**

**둘째, 실수로 조인 조건 컬럼 중 일부를 빠뜨리는 경우.**