

[연산자]

= : 같다

!=, ^=, <> : 같지 않다

>=, <=, >, < : 크거나 같다, 작거나 같다, 크다, 작다

and, or, between and, in, like, is null, is not null

※ select

select [distinct] [컬럼1, 컬럼2.....][*]

from 테이블명

[where 조건절]

[order by 컬럼명 asc|desc]

order by : 정렬

asc - 오름차순 (생략가능)

desc - 내림차순

컬럼명 : 숫자로도 가능

ex1) 사원명, 부서ID, 입사일을 부서별로 내림차순 정렬하시오

select last_name, department_id, hire_date

from employees

order by 2 desc;

ex2) 사원명, 부서ID, 입사일을 부서별로 내림차순 정렬하시오

같은 부서가 있을 때는 입사일순으로 정렬하시오

select last_name, department_id, hire_date

from employees

order by 2 desc, 3 asc;

[문제1] 사원들의 연봉을 구한 후 연봉 순으로 내림차순 정렬하시오

이름	연봉
King	288000
Kochhar	204000

[단일행 함수]

1. 숫자함수 : mod, round, trunc, ceil
2. 문자함수 : lower, upper, length, substr, ltrim, rtrim, trim
3. 날짜함수 : sysdate, add_month, month_between
4. 변환함수

(1) 암시적(implicit) 변환 : 자동

VARCHAR2 또는 CHAR	----->	NUMBER
VARCHAR2 또는 CHAR	----->	DATE
NUMBER	----->	VARCHAR2
DATE	----->	VARCHAR2

(2) 명시적(explicit) 변환 : 강제

	TO_NUMBER	TO_DATE
	<-----	----->
NUMBER	CHARACTER	DATE
	----->	<-----
	TO_CHAR	TO_CHAR

- 날짜 형식 -

YYYY : 네 자리 연도(숫자) (ex. 2005)
YEAR : 연도(문자)
MM : 두 자리 값으로 나타낸 달 (ex. 01, 08, 12)
MONTH : 달 전체이름 (ex. January)
MON : 세 자리 약어로 나타낸 달 (ex. Jan)
DY : 세 자리 약어로 나타낸 요일 (ex. Mon)
DAY : 요일전체 (ex. Monday)
DD : 숫자로 나타낸 달의 일 (ex. 31, 01)
HH, HH24(24시간제)
MI
SS

- 숫자 형식 -

9 : 숫자를 표시
0 : 0을 강제로 표시
\$: 부동\$기호를 표시
L : 부동 지역통화기호 표시
. : 소수점출력
, : 천단위 구분자 출력

5. 그룹(집합)함수 : avg, sum, max, min, count
6. 기타함수 : nvl, dcode, case

ex1) 이름을 소문자로 바꾼 후 검색

'Higgins'사원의 사원번호, 이름, 부서번호를 검색하시오

```
select employee_id, last_name, department_id
from employees
where lower(last_name)='higgins';
```

ex2) 10을 3으로 나눈 나머지 구하시오(mod)

```
select mod(10, 3) from dual; → 가상의 테이블
```

ex3) 35765.357을 반올림(round)

위치가 n일 때 n이 양수이면 (n+1)에서 반올림이 되고

n이 음수이면 n의 위치에서 반올림 된다

```
select round(35765.357, 2) from dual; -- 35765.36
select round(35765.357, 0) from dual; -- 35765
select round(35765.357, -3) from dual; -- 36000
```

ex4) 35765.357을 내림(trunc)

위치가 n일 때 n이 양수이면 (n+1)에서 반올림이 되고

n이 음수이면 n의 위치에서 반올림 된다

```
select trunc(35765.357, 2) from dual; -- 35765.35
select trunc(35765.357, 0) from dual; -- 35765
select trunc(35765.357, -3) from dual; -- 35000
```

ex5) concat('문자열1', '문자열2') : 문자열의 결합(문자열1+문자열2)

```
select concat('Hello', ' World') from dual;
```

ex6) length('문자열') : 문자열의 길이

lengthb('문자열') : 문자열의 길이

```
create table text (
str1 char(20),
str2 varchar2(20));
```

char : 고정문자길이

varchar2 : 가변문자길이

```
insert into text(str1, str2) values('angel', 'angel');
insert into text(str1, str2) values('사천사', '사천사');
commit;
```

```
select lengthb(str1), lengthb(str2) from text;
```

```
20    5
```

```
20    9
```

```
select length(str1), length(str2) from text;
```

```
20    5
```

```
14    3
```

ex7)

```
select length('korea') from dual; -- 5
```

```
select length('코리아') from dual; -- 3
```

```
select lengthb('korea') from dual; -- 5
```

```
select lengthb('코리아') from dual; -- 9
```

ex8) 지정한 문자열 찾기 : instr(표현식, 찾는 문자, [위치]) 양수: 앞(생략가능), 음수: 뒤

```
select instr('HelloWorld', 'W') from dual; -- 6
```

```
select instr('HelloWorld', 'o', -5) from dual; -- 5
```

```
select instr('HelloWorld', 'o', -1) from dual; -- 7
```

ex9) 지정한 길이의 문자열을 추출 : substr(표현식, 시작, [개수])

```
select substr('I am very happy', 6, 4) from dual; -- very
```

```
select substr('I am very happy', 6) from dual; -- very happy
```

[문제2] 사원의 레코드를 검색하시오 (concat, length)

조건1) 성과 이름을 연결하시오 (concat)

조건2) 구해진 이름의 길이를 구하시오 (length)

조건3) 이름이 n으로 끝나는 사원 (substr)

EMPLOYEE_ID	NAME	LENGTH
102	Lex De Haan	11
105	David Austin	12

ex10) 임의의 값이 지정된 범위 내에 어느 위치에 있는지를 찾는다

: width_bucket(표현식, 최소값, 최대값, 구간)

최소-최대값을 설정하고 10개의 구간을 설정 후 위치 찾기

0-100까지의 구간을 나눈 후 74가 포함되어 있는 구간을 표시하시오

```
select width_bucket(74, 0, 100, 10) from dual; -- 8
```

ex11) 공백제거 : ltrim(왼), rtrim(오른), trim(양쪽)

```
select rtrim('test ') || 'exam' from dual;
```

ex12) sysdate : 시스템에 설정된 시간표시

```
select sysdate from dual;
select to_char(sysdate, 'YYYY"년" MM"월" DD"일"') as 오늘날짜 from dual;
select to_char(sysdate, 'HH"시" MI"분" SS"초"') as 오늘날짜 from dual;
select to_char(sysdate, 'HH24"시" MI"분" SS"초"') as 오늘날짜 from dual;
```

ex13) add_months(date, 달수) : 날짜에 달수 더하기

```
select add_months(sysdate, 7) from dual;
```

ex14) last_day(date) : 해당달의 마지막 날

```
select last_day(sysdate) from dual;
select last_day('2004-02-01') from dual; -- 29
select last_day('2005-02-01') from dual; -- 28
```

[문제3] 오늘부터 이번 달 말까지 총 남은 날수를 구하시오

ex15) months_between(date1, date2) : 두 날짜 사이의 달 수

```
select round(months_between('95-10-21', '94-10-20'), 0) from dual; ← 자동 형변환
```

명시적인 변환(강제)

```
select last_name, to_char(salary, 'L99,999.00')
from employees
where last_name='King';
```

ex16)

```
select to_char(to_date('97/9/30', 'YY-MM-DD'), 'YYYY-MON-DD') from dual; -- 2097
select to_char(to_date('97/9/30', 'RR-MM-DD'), 'RRRR-MON-DD') from dual; -- 1997

select to_char(to_date('17/9/30', 'YY-MM-DD'), 'YYYY-MON-DD') from dual; -- 2017
select to_char(to_date('17/9/30', 'RR-MM-DD'), 'RRRR-MON-DD') from dual; -- 2017
```

(반환일:현재년도의 세기)	지정한 연도(뒤 두자리)		
	0-49	50-99	
현재년도(뒤 두자리)	0-49	반환일의 현재세기	반환일의 이전세기
	50-99	반환일의 다음세기	반환일의 현재세기

[문제4] 2005년 이전에 고용된 사원을 찾으시오

LAST_NAME	HIRE_DATE
King	17-6월 -2003
De Haan	13-1월 -2001

ex17) fm형식 : 형식과 데이터가 반드시 일치해야함(fm - fm 사이 값만 일치)
fm를 표시하면 숫자 앞의 0을 나타나지 않는다.

```
select last_name, hire_date from employees where hire_date='05/09/30';  
select last_name, hire_date from employees where hire_date='05/9/30';
```

```
select to_char(sysdate, 'YYYY-MM-DD') from dual;  
select to_char(sysdate, 'YYYY-fmMM-DD') from dual;
```

```
select to_char(to_date('2011-03-01','YYYY-MM-DD'), 'YYYY-MM-DD') from dual;  
select to_char(to_date('2011-03-01','YYYY-MM-DD'), 'YYYY-fmMM-DD') from dual;  
select to_char(to_date('2011-03-01','YYYY-MM-DD'), 'YYYY-fmMM-fmDD') from dual;
```

ex18) count(컬럼명), max(컬럼명), min(컬럼명), avg(컬럼명), sum(컬럼명) 함수
employees테이블에서 급여의 최대, 최소, 평균, 합을 구하시오
조건) 평균은 소수이하절삭, 합은 세자리마다 콤마 찍고 ₩표시

```
select max(salary),  
       min(salary),  
       trunc(avg(salary), 0),  
       to_char(sum(salary), 'L9,999,999') from employees;
```

[문제5] 커미션(commission_pct)을 받지 않은 사원의 인원수를 구하시오

ex19) employees테이블에서 없는 부서 포함해서 총 부서의 수를 구하시오

```
select department_id from employees; -- 107  
select count(department_id) from employees; -- 106  
select count(*) from employees; -- 107  
select count(distinct department_id) from employees; -- 11  
select count(distinct nvl(department_id, 0)) from employees; -- 12  
select distinct nvl(department_id, 0) from employees; -- nvl은 null값을 0으로 대치
```

ex20) ① decode(표현식, 검색1,결과1, 검색2,결과2....[default])
: 표현식과 검색을 비교하여 결과 값을 반환 다르면 default

```
② case value when 표현식 then 구문1  
            when 표현식 then 구문2  
            else 구문3  
end
```

업무 id가 'SA_MAN' 또는 'SA_REP'이면 'Sales Dept' 그 외 부서이면 'Another'로 표시
조건) 분류별로 오름차순 정렬

```
select job_id, decode(job_id,  
                      'SA_MAN', 'Sales Dept',  
                      'SA_REP', 'Sales Dept',  
                      'Another') "분류"  
  
from employees  
order by 2;
```

```
-----  
select job_id, case job_id  
                when 'SA_MAN' then 'Sales Dept'  
                when 'SA_REP' then 'Sales Dept'  
                else 'Another'  
            end "분류"  
  
from employees  
order by 2;
```

```
-----  
select job_id, case  
                when job_id='SA_MAN' then 'Sales Dept'  
                when job_id='SA_REP' then 'Sales Dept'  
                else 'Another'  
            end "분류"  
  
from employees  
order by 2;
```

[문제6] 급여가 10000 미만이면 초급, 20000 미만이면 중급 그 외면 고급을 출력하시오

조건1) 제목은 직원번호, 직원명, 구분으로 표시하시오

조건2) 구분 컬럼으로 오름차순 정렬하고, 같으면 직원명 컬럼으로 오름차순 하시오

조건3) case 사용하시오

직원번호	직원명	구분
100	King	고급
174	Abel	중급
204	Baer	중급

ex21) rank함수 : 전체 값을 대상으로 순위를 구함

rank(표현식) within group(order by 표현식)

rank() over(쿼리파티션) → 전체 순위를 표시

급여가 3000인 사람의 상위 급여 순위를 구하시오

```
select rank(3000) within group(order by salary desc) "rank" from employees;
```

전체사원의 급여순위를 구하시오

```
select employee_id, salary, rank() over(order by salary desc) "rank" from employees;
```

ex22) first_value함수 : 정렬된 값 중에서 첫 번째 값 반환

first_value(표현식) over(쿼리파티션)

전체사원의 급여와 함께 각 부서의 최고급여를 나타내고 비교하시오

```
select employee_id,  
       salary,  
       department_id,  
       first_value(salary) over(partition by department_id order by salary desc)  
       "highsal_deptID"  
from employees;
```

★ PARTITION BY 절은 GROUP BY 절과 동일한 역할을 진행 합니다.

단, GROUP BY 절을 사용하지 않고 필요한 집합으로 행들을 그룹화 시킴

Partition by 절을 사용 함으로 GROUP BY 절 없이 다양한 GROUPING 집합의 집계 결과들을 함께 출력 할 수 있습니다.

ORDER BY 절은 Partition by 로 정의된 WINDOW 내에서의 행들의 정렬 순서를 정의 한다.

```
select employee_id,  
       last_name,  
       salary,  
       department_id,  
       row_number( ) over ( PARTITION BY department_id ORDER BY salary DESC ) rnum  
from employees ;
```

부서별 급여를 내림차순으로 정렬 했을 경우 Row Number

EMPLOY...	LAST_NAME	SALARY	DEPARTMENT_ID	RNUM
200	Whalen	4400	10	1
201	Hartstein	13000	20	1
202	Fay	6000	20	2
114	Raphaely	11000	30	1
115	Khoo	3100	30	2
116	Baida	2900	30	3
117	Tobias	2800	30	4
118	Himuro	2600	30	5

부서 번호가 바뀔 때 Row Number 는 새로 시작 되는 것을 확인 할 수 있습니다.

NULL 값은 정렬 시 가장 큰 값으로 인식 (기본설정)

[문제7] 사원테이블에서 사원번호, 이름, 급여, 커미션, 연봉을 출력하시오

조건1) 연봉은 \$ 표시와 세자리마다 콤마를 사용하시오

조건2) 연봉 = 급여 * 12 + (급여 * 12 * 커미션)

조건3) 커미션을 받지 않는 사원도 포함해서 출력하시오

[문제8] 매니저가 없는 사원의 MANAGER_ID를 1000번으로 표시

조건1) 제목은 사원번호, 이름, 매니저ID

조건2) 모든 사원을 표시하시오

EMPLOYEE_ID	LAST_NAME	MANAGER_ID
100	King	1000
101	Kochhar	100