#################연습문제 ##########################

#################set operator(집합 연산자)#####################

교집합

합집합

차집합

conn hr/oracle

desc employees

select count(\*) from employees; --현재 근무하는 사원들의 정보(107)

desc job\_history

select count(\*) from employees;--사원들이 과거에 근무 이력 정보(10)

Q> 사원들의 현재 근무정보와 과거 근무 정보를 모두 출력하는 SQL 작성

select employee\_id, department\_id, job\_id

from employees

union all

select employee\_id, department\_id, job\_id

from job\_history; ---(117)

Q> 사원들의 현재 근무정보와 과거 근무 정보를 모두 출력하되

동일한 직무와 부서에서의 근무인 경우 한번만 출력하는 SQL 작성

select employee\_id, department\_id, job\_id

from employees

union

select employee\_id, department\_id, job\_id

from job\_history; ---()

Q> 사원들의 현재 직무를 과거에 동일한 직무를 수행했던 사원번호, 직무를 출력하는 SQL 작성

select employee\_id, job\_id

from employees

intersect

select employee\_id, job\_id

from job\_history;

Q> 사원들중에서 입사한 이후 한번도 부서나 직무를 변경한적이 없는

사원번호를 출력하는 SQL 작성

select employee\_id

from employees

minus

select employee\_id

from job\_history;

--set연산자를 사용하는 select문에서는 set연산자와 함께 사용되는 select문중 마지막 select문에서만 선언 가능합니다.

select employee\_id, job\_id

from employees

order by job\_id desc

intersect

select employee\_id, job\_id

from job\_history; --error

select employee\_id, job\_id

from employees

intersect

select employee\_id, job\_id

from job\_history

order by job\_id desc;

Q> 전체 사원의 급여 평균과

부서별 사원들의 급여 평균과

부서와 직무별 사원들의 급여 평균을 단일 결과 집합으로 생성하는 SQL을 작성하시오

select to\_number(null), to\_char(null),avg(salary)

from employees

union all

select department\_id, to\_char(null), avg(salary)

from employees

group by department\_id

union all

select department\_id, job\_id, avg(salary)

from employees

group by department\_id, job\_id;

Q> 전체 사원의 급여 평균과

부서별 사원들의 급여 평균과

직무별 사원들의 급여 평균과

부서와 직무별 사원들의 급여 평균을 단일 결과 집합으로 생성하는 SQL을 작성하시오

select to\_number(null), to\_char(null),avg(salary)

from employees

union all

select department\_id, to\_char(null), avg(salary)

from employees

group by department\_id

union all

select to\_number(null), job\_id, avg(salary)

from employees

group by job\_id

union all

select department\_id, job\_id, avg(salary)

from employees

group by department\_id, job\_id;

#############group function(multiple row function)###########################################

--테이블 전체 레코드를 하나의 그룹으로 함수에 적용

--테이블의 레코드를 특정 컬럼으로 그룹핑하고 그룹핑된 레코드들에 함수 적용

--count(), min(), max(), sum(), avg(), stddev(), variance()

select count(sal), min(sal), max(sal), sum(sal), avg(sal), stddev(sal), variance(sal)

from emp;

--count(), min(), max()는 모든 컬럼타입 적용 가능

select count(hiredate), min(hiredate), max(hiredate)

from emp;

select count(ename), min(ename), max(ename)

from emp;

--count는 null이 아닌 컬럼값의 개수를 리턴

--count는 인수로 \*를 사용할 수 있음=> 테이블의 행수를 리턴 (not null제약조건이 선언된 컬럼값의 개수 계산 리턴)

select count(\*), count(comm), count(deptno), count(distinct deptno)

from emp;

※ 그룹함수는 null을 함수 계산에 포함하지 않습니다.(무시합니다.)

※ avg

select avg(comm), sum(comm)/count(empno), sum(comm)/count(comm)

from emp; ---comm컬럼에 null포함되어 있으므로 전체 사원의 커미션 평균이 아닌 커미션을 받는 사원들의 커미션 평균이 출력됨

select avg(nvl(comm, 0)), sum(comm)/count(empno)

from emp;

Q> 어떤 컬럼이 모두 null인 경우 count함수의 결과는?

1, 오류

2, null

3, 0

alter table emp add (address varchar2(50));

desc emp

select empno, ename, address from emp;

select count(address) from emp;

Q> 어떤 컬럼이 모두 null인 경우 sum함수의 결과는?

1, 오류

2, null

3, 0

alter table emp drop (address );

alter table emp add (price number(6));

desc emp

select empno, ename, price from emp;

select sum(price) from emp;

select deptno, avg(comm), sum(comm)

from emp; ---error

select deptno, avg(comm), sum(comm)

from emp

group by deptno;

--1차 그룹된 레코드에 2차 그룹핑 적용 가능

select deptno, job, avg(comm), sum(comm)

from emp

group by deptno, job;

Q> 각 부서별 평균 급여가 2500이상인 부서와 해당 부서의 급여 평균을 검색 출력하시오

select deptno, avg(sal), sum(sal)

from emp

where avg(sal) >= 2500

group by deptno; ---error , where절은 group by 전에 수행되며, 그룹함수 사용 불가

--그룹함수의 조건 지정, group by 후에 수행

select deptno, avg(sal), sum(sal)

from emp

group by deptno

hvaing avg(sal) >= 2500;

conn hr/oracle

Q> employees 테이블의 department\_id, salary 컬럼...

부서별 급여의 평균을 평균의 내림차순으로 출력하도록 SQL 작성

select department\_id, avg(salary)

from employees

group by department\_id

order by 2 desc;

Q> employees 테이블의 department\_id, salary, manager\_id 컬럼...

관리자가 있는 사원들을 관리자로 그룹핑해서 동일한 관리자로부터

관리를 받는 피관리자의 최소 급여가 6000미만인 사원의 관리자와 최소급여를 내림차순으로 출력하는 SQL 작성

select manager\_id, min(salary) ---5

from employees ----1

where manager\_id is not null -----2

group by manager\_id ----3

having min(salary) < 6000 ----4

order by 2 desc; ---6

Q> 전체 사원의 급여 평균과

부서별 사원들의 급여 평균과

부서와 직무별 사원들의 급여 평균을 단일 결과 집합으로 생성하는 SQL을 작성하시오

select department\_id, job\_id, avg(salary)

from employees

group by rollup (department\_id, job\_id);

Q> 전체 사원의 급여 평균과

부서별 사원들의 급여 평균과

직무별 사원들의 급여 평균과

부서와 직무별 사원들의 급여 평균을 단일 결과 집합으로 생성하는 SQL을 작성하시오

select department\_id, job\_id, avg(salary)

from employees

group by cube (department\_id, job\_id);

Q> 전체 사원의 급여 평균과

부서별 사원들의 급여 평균과

관리자와 직무별 사원들의 급여 평균과

부서와 관리자별 사원들의 급여 평균을 단일 결과 집합으로 생성하는 SQL을 작성하시오

select department\_id, job\_id, avg(salary)

from employees

group by grouping sets ((department\_id, manager\_id, job\_id), (department\_id),

(department\_id, manager\_id),

(department\_id, job\_id));

#3장, 4장, 5장 , 8장의 슬라이드

######################converion function#######################

select sysdate, to\_char(sysdate, 'YYYY"년" MM"월" DD"일" Day')

from dual; --to\_char의 1번째 argument 값 형식과 변환할 두번째 argument값 형식은 달라도 변환됨

select 1234.56, to\_char(1234.56 , '$999,999.990')

from dual;

select '$1,234.56' , to\_number('$1,234.56' , '999,999.990')

from dual; ---error

select '$1,234.56' , to\_number('$1,234.56' , '$999,999.990')

from dual; --?

select '2020년 2월 5일 수'

, to\_date('2020년 2월 5일 수요일', ' RR/MM/DD Day')

from dual; --error

----to\_date의 1번째 argument 값 형식과 변환할 두번째 argument값 형식은 달라도 변환됨

select '2020년 2월 5일 수'

, to\_date('2020년 2월 5일 수요일', 'YYYY"년" MM"월" DD"일" Day')

from dual; --?

#######################date function#######################

select ename, hiredate, months\_between(sysdate, hiredate)

from emp;

select ename, hiredate, trunc(months\_between(sysdate, hiredate))

from emp;

select trunc(to\_date('2021/7/16'), 'Month')

, trunc(to\_date('2021/7/14'), 'Month')

, trunc(to\_date('2021/7/16'), 'Year')

, trunc(to\_date('2021/6/16'), 'Month')

from dual; --round함수도 날짜 타입에 적용 가능 함수

select sysdate, add\_months(sysdate, 3)

from dual;

alter session set nls\_date\_format='RR/MM/DD HH24:MI:SS';

select sysdate, current\_date, current\_timestamp, sessiontimezone

from dual;

--sysdate함수는 시스템(운영체제)의 현재 시간을 date타입값으로 반환

--current\_date 함수는 DB접속 client 세션의 timezone기반으로 date타입값으로 반환

--current\_date 함수는 DB접속 client 세션의 timezone기반으로 timestamp타입(date+fractional second)값으로 반환

alter session set time\_zone='+3:00';

select sysdate, current\_date, current\_timestamp, sessiontimezone

from dual;

select dbtimezone from dual;

alter session set nls\_date\_format='RR/MM/DD';

select last\_day(sysdate), last\_day(to\_date('1900/02/03'))

, last\_day(to\_date('2000/02/03')) , last\_day(to\_date('1996/02/03'))

from dual;

select next\_day(sysdate, '금'), next\_day(sysdate, '월')

from dual;

select hiredate, extract( month from hiredate)

from emp;

select hiredate, extract( day from hiredate)

from emp;

#날짜 시간 관련 컬럼 타입

timestamp with timezone

interval year to month

interval day to second

select hiredate, hiredate + TO\_DSINTERVAL('100 00:00:00')

, hiredate + to\_yminterval('01-02')

from emp;

###################generall function########################

--null처리 함수

--nvl(arg1, arg2) : if arg1 is not null then return arg1

else return arg2

반드시 arg1, arg2가 동일한 타입이어야 함

--nvl2(arg1, arg2, arg3)) : if arg1 is not null then return arg2

else return arg3

반드시 arg2, arg3가 동일한 타입이어야 함

--coalesce(arg1, arg2,......argn) : if arg1 is not null then return arg1

else if arg2 is not null then return arg2

....

null아닌 argument를 리턴하고 함수는 종료

select nvl(comm, 'No Commission')

from emp; ---error

select nvl(to\_char(comm), 'No Commission')

from emp;

Q> emp테이블에서 commission을 받는 사원은 sal+comm 리턴하고,

commission을 받지 않는 사원은 sal 리턴한 결과 출력

select ename, sal, comm, nvl2(comm, sal+comm, sal)

from emp;

select coalesce(1,2,3,4,5) , coalesce(null, null, 3, 4, 5)

, coalesce(null, null, null,null, 5)

from dual;

조건처리 : decode(), 표준sql 구문 case when then ...else end

Q> 10번 부서 사원은 급여를 5%인상,

20번 부서 사원은 급여를 7%인상,

30번 사원은 급여를 3%인상된 급여를 현재급여와 함께 출력

select ename, deptno, sal

, decode(deptno, 10, sal\*1.05, 20, sal\*1.07, 30, sal\*1.03, sal) "Increase"

from emp;

select ename, deptno, sal

, case deptno when 10 then sal\*1.05

when 20 then sal\*1.07

when 30 then sal\*1.03

else sal end "Increase"

from emp;

Q> 급여에 대한 세금을 급여와 함께 출력하시오

급여가 1000미만이면 세금 0원

급여가 1000이상 2000미만이면 세금 급여의 5%

급여가 2000이상 3000미만이면 세금 급여의 10%

급여가 3000이상 4000미만이면 세금 급여의 15%

급여가 4000이상 이면 세금 급여의 20%

컬럼 별칭은 Tax

select ename, deptno, sal

, decode(trunc(sal/1000), 0, 0 ,

1, sal\*0.05,

2, sal\*0.1,

3, sal\*0.15,

sal\*0.2) "Tax"

from emp;

select ename, deptno, sal

, case when sal< 1000 then 0

when sal< 2000 then sal\*0.05

when sal< 1000 then sal\*0.1

when sal< 1000 then sal\*0.15

else sal\*0.2 end "Tax"

from emp;

####################number function#################################

select round(1234.567, 2), round(1234.567, 0), round(1234.567, -2)

from dual;

select trunc(1234.567, 2), trunc(1234.567), trunc(1234.567, -2)

from dual;

select mod(100, 35), remainder(100, 35)

from dual;

select ceil(34.56), floor(34.56), power(2, 10)

from dual;

Q> emp테이블로부터 82년도에 입사한 사원 조회

select ename

from emp

where hiredate like '87%';

select ename

from emp

where '87' = substr(hiredate , 1, 2);

select ename

from emp

where '87' = to\_char(hiredate, 'YY') ;

Q> emp테이블로부터 사번이 홀수인 사원 조회

select ename

from emp

where 1 = mod(empno, 2);

#################character function####################

ascii-127개문자

ebcdic-256개문자

select chr(48), chr(65), chr(97), chr(13)

from dual;

select upper('hello'), lower('HELLO'), initcap('HELLO SQL')

from dual;

select concat('hello' , ' SQL')

from dual;

select substr('hello world', 6), substr('hello world', 3,2)

, substr('hello world', -5, 3)

from dual;

select replace('Jack and Jue', 'J', 'Bl')

from dual;

select instr('hello world', 'o'), instr('hello world', 'o', 6)

, instr('hello world', 'o', 1, 2) , instr('hello world', 'o',-5)

from dual; --값의 존재 유무를 체크할 때 유용함

select length('korea'), length('대한민국')

lengthb('korea'), lengthb('대한민국')

from dual;

select length(' k o r e a ') , length(trim(' k o r e a '))

from dual; --함수 내부에 함수를 사용할 수 있으며, 실행시 안쪽 함수부터 평가됨

※ rtrim, ltrim함수는 왼쪽이나 오른쪽 문자 또는 공백을 제거할 수 있습니다.

select translate('Jack and Jue', 'acJ', '137')

from dual;

select sal, lpad(sal, 10 , '$'), rpad(sal, 10, '$')

from emp;

conn hr/oracle

desc employees

select phone\_number from employees;

SELECT

REGEXP\_REPLACE(phone\_number,

'([[:digit:]]{3})\.([[:digit:]]{3})\.([[:digit:]]{4})',

'(\1) \2-\3') "REGEXP\_REPLACE"

FROM employees

ORDER BY "REGEXP\_REPLACE";

SELECT country\_name,

REGEXP\_REPLACE(country\_name, '(.)', '\1 ') "REGEXP\_REPLACE"

FROM countries;

SELECT

REGEXP\_SUBSTR('500 Oracle Parkway, Redwood Shores, CA',

',[^,]+,') "REGEXPR\_SUBSTR"

FROM DUAL;

SELECT REGEXP\_COUNT('123123123123123', '(12)3', 1, 'i') REGEXP\_COUNT

FROM DUAL;

SELECT REGEXP\_COUNT('123123153123123', '(12)3', 1, 'i') REGEXP\_COUNT

FROM DUAL;

SELECT

REGEXP\_INSTR('500 Oracle Parkway, Redwood Shores, CA',

'[^ ]+', 1, 6) "REGEXP\_INSTR"

FROM DUAL;

#######################Review######################

selection검색 - where 조건

where 컬럼 (비교)연산자 비교값

비교 연산자 , 범위연산자 (between 하한값 and 상한값), in (값 리스트), like '% \_', is null, is not null

논리 연산자 not and or

heap(일반) table은 insert한 순서대로 block에 저장되며, 조회 결과는 block에 저장된 순서대로 반환됩니다.

order by :조회 결과를 특정 컬럼기준으로 정렬

order by 컬럼 정렬방식(asc|desc)

order by 표현식 정렬방식(asc|desc)

order by 별칭 정렬방식(asc|desc)

order by 컬럼 position 정렬방식(asc|desc)