

SELECT 데이터 조회



DML (Data Manipulation Language) 데이터에 대한 CRUD 를 수행하는 쿼리 문 집합

select → 테이블 내의 데이터를 조회하는 쿼리문법

▼ 기본 구조

- select 폴더안에 파일명 * (*은 모든 데이터를 다 불러옴)
- from 폴더명
- where 조건

```
select * from department;
select * from job;
select * from employee;

// where 조건
select *
from employee
where emp_name = '선동일'
```

▼ asc 오름차순 (기본) desc 내림차순

▼ 정렬 order by

```
select * // 모든 것을
from employee
order by salary; // 줄 세워서, 뒤에는 기준!
```

▼ null 검사

is null

is not null

```
select *
from employee
WHERE phone is null; --null
```

```
select * from employee WHERE phone is not null; --null이 아닌
```

▼ nvl

보너스가 null 이면 0으로 바꿔줌

```
select nvl(bonus,0) from employee;
```

▼ 문자열 연결 ||

```
select
    emp_name as 직원명,
    salary * 12 || '원' as 연봉,
    'Oracle' 오라클쉽다
```

▼ or 와 and

```
select *
from employee
where emp_name = '선동일' or emp_name = '윤은해';
```

```
select *
```

```
from employee
where dept_code = 'D9' and salary <= 5000000;
```

▼ [and] **between** (사이)

and 특정 범위의 값을 조건 지정할 때 사용할 수 있는 직관적 문법 (3~5사이면 3,4,5 다 나옴)

```
select
    emp_name,salary from employee
where
    salary >= 3000000 and salary <= 4000000;

select
    emp_name,salary from employee
WHERE
    salary between 3000000 and 3700000;
```

▼ [or] **in**

or 연산자가 다수 필요할 경우 편리하게 줄여 사용할 수 있는 문법 (and는 ❌)

```
select emp_name,dept_code,phone
from employee
WHERE dept_code = 'D5' or dept_code = 'D9' or dept_code = 'D1';
```

```
select emp_name,dept_code,phone
from employee
WHERE dept_code in('D5','D9','D1');
```

```
WHERE dept_code in('D5','D9','D1');
WHERE dept_code not in('D5','D9','D1'); -- 'D5','D9','D1' 빼고
```

▼ 날짜 출력 방법 **sysdate**

timestamp 값을 모두 반환하지만 (시분초까지 포함한) sql developer 가 출력될때 날짜 형식으로 저장하여 연월일 출력된다

```
select emp_name, sysdate
from employee;

//날짜 타입의 데이터에 덧셈 또는 뺄셈 연산은 일로 계산된다.
select emp_name, sysdate-1
from employee;

//sysdate 오늘날짜 - hire_date 입사날짜
sysdate - hire_date
//날짜 타입끼리의 뺄셈은 두 날짜간 일수 차이를 만들어낸다.
```

▼ floor 소수점 제거

```
select floor(3.1231232121)
from employee;
```

▼ Like (~같은) 와 함께 사용되는 특수문자 % 와 _

문자열을 비교할 때 같다가 아닌 **contains** 또는 **startswith** 또는 **endswiths** 를 비교할 수 있다

% 여러 글자가 존재할 수도 있고 없을 수도 있다라는 의미

```
SELECT * from employee where emp_name like '재%'; -- startwith
SELECT * from employee where emp_name like '%재'; -- endswith
SELECT * from employee where emp_name like '%재%'; --contains
```

_ 딱 한글자가 존재 해야함 의미

```
SELECT * from employee where emp_name like '전__';

-- not 도 가능
SELECT * from employee where emp_name not like '웅';
```

전화번호가 010으로 시작하지 않은 사람들 출력

```
SELECT * from employee where phone not like '010%
```

특정 날짜보다 작음
