database-6

```
 태그 날짜 @2023년 6월 12일
```

조금 더 복잡한 조회를 위해 쿼리 안에 쿼리를 넣어서 select하는 방법

내용

- subquery
- outer query
- IN 키워드와 함께 subquery 사용하기
- correlated subquery
- EXISTS 키워드와 함께 subquery 사용하기
- ANY 키워드와 함께 subquery 사용하기
- ALL 키워드와 함께 subquery 사용하기

SELECT with subquery

• ID가 14인 임직원보다 생일이 빠른 임직원의 ID, 이름, 생일을 알고 싶다.

SELECT birth_date FROM employee WHERE id = 14; // 1



database-6

```
SELECT id, name, birth_date FROM employee WHERE birth_data < '1992-08-04'; // 2
두개의 쿼리를 한번에 실행할 순 없을까?
SELECT id, name, birth_date FROM employee
WHERE birth_date < (SELECT birth_date FROM employee WHERE id = 14; );
```

```
용어 정리

- subquery (nested query or inner query) :
SELECT, INSERT, UPDATE, DELETE에 포함된 query

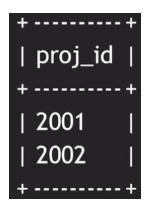
- outer query (main query) :
subquery를 포함하는 query

-subquery는 () 안에 기술된다.
위에서는 (SELECT birth_date FROM employee WHERE id = 14; );<< 이게 서브쿼리
```

```
SELECT with subquery
ID가 1인 임직원과 같은 부서 같은 성별인 임직원들의 ID와 임직원들의 ID와 이름과 직군을 알고 싶다.
SELECT id, name, position
FROM employee
WHERE (dept_id, sex) = (SELECT dept_id, sex FROM employee WHERE id = 1);
```

```
id가 5인 임직원과 같은 프로젝트에 참여한 임직원들의 id
SELECT proj_id FROM works_on WHERE empl_id=5;
```

database-6 2



```
SELECT DISTINCT empl_id FROM works_on
    WHERE empl_id != 5 AND (proj_id = 2001 OR proj_id = 2002);

(proj_id = 2001 OR proj_id = 2002);를 간단하게 작성하는 방법

SELECT DISTINCT empl_id FROM works_on
    WHERE empl_id != 5 AND proj_id IN (2001,2002);
```

```
SELECT DISTINCT empl_id FROM works_on

WHERE empl_id != 5 AND proj_id IN (SELECT proj_id FROM works_on WHERE empl_id = 5);
```

```
정리
v IN (v1, v2, v3, ...) : v가 (v1, v2, v3, ...)중에 하나와 값이 같다면 true를 return한다.
(v1, v2, v3, ...)는 명시적인 값들의 집할일 수도 있고 subquery의 결과 (set or multiset)일 수도 있다.
v NOT IN(v1, v2, v3, ...) : v가 (v1, v2, v3, ...)의 모든 값과 값이 다르다면 true를 return한다.
unqualified attribute가 참조하는 table은 해당 attribute가 사용된 query를 포함하여 그 query의 바깥 쪽으로 존재하는 모든 queries 중에 해당 attribute이름을 가지는 가장 가까이에 있는 table을 참조한다.
```

database-6

```
id가 7 혹은 12인 임직원이 참여한 프로젝트 id와 이름을 알고 싶다.

SELECT P.id, P.name
FROM project P
WHERE EXISTS(
SELECT * FROM works_on W WHERE W.proj_id = AND W.empl_id IN (7,12)
);

정리

correlated query : subquery가 바깥쪽 query의 attribute를 참조할 때, correlated subquery라 부름

EXISTS : subquery의 결과가 최소 하나의 row 라도 있다면 true를 반환

NOT EXISTS : subquery의 결과가 단 하나의 row도 없다면 true를 반환
```

database-6