

6장 Subquery

목적

- 서브쿼리가 해결 할 수 있는 문제의 유형을 기술
- 서브쿼리를 정의
- 서브쿼리의 유형을 나열
- 단일 행 서브쿼리와 다중 행 서브쿼리를 작성

문제 해결을 위한 서브쿼리 사용

“Jones 보다 많은 급여를 받는 사람은?”

Main query



“Jones의 급여보다 많은 급여를
받는 종업원은 누구인가?”



Subquery



Jones의 급여는 얼마인가?

내부 질의 또는 서브쿼리는 외부질의 또는 메인 쿼리에 의해 사용되는 값을 리턴


서브쿼리

```
SELECT      select_list
FROM        table
WHERE       expr operator
           (SELECT select_list FROM table);
```

- 서브쿼리(내부질의)는 메인 쿼리 이전에 한번만 실행됨
- 서브쿼리의 결과는 메인 쿼리(외부질의)에 의해 사용됨

서브쿼리 사용

```
SQL> SELECT      ename
  2  FROM      emp
  3  WHERE      sal >
  4      (SELECT sal FROM emp WHERE empno= 7566);
```



ENAME

KING

FORD

SCOTT

서브쿼리 사용지침

- 서브쿼리는 괄호로 둘러싸여야 함
- 서브쿼리는 비교 연산자의 오른쪽에 위치 하여야 함
- 서브쿼리에 **ORDER BY** 절을 포함하지 말 것
 - **SELECT** 문장에는 오직 하나의 **ORDER BY** 절이 올 수 있으며 문장의 끝에 위치하여야 함
- 단일 행 서브쿼리에는 단일 행 연산자를 사용
- 다중 행 서브쿼리에는 다중 행 연산자를 사용

서브쿼리의 유형

- 단일 행 서브쿼리 : 내부 SELECT 문장에서 오직 하나의 행만을 리턴하는 질의



- 다중 행 서브쿼리 : 내부 SELECT 문장에서 하나 이상의 행을 리턴하는 질의



- 다중 열 서브쿼리 : 내부 SELECT 문장에서 하나 이상의 열을 리턴하는 질의



단일 행 서브쿼리

- 오직 하나의 행만을 리턴
- 단일 행 비교 연산자를 사용

연산자	의미
=	같다
>	보다 크다
>=	보다 크거나 같다
<	보다 작다
<=	보다 작거나 같다
<>	같지 않다

단일 행 서브쿼리 실행

```
SQL> SELECT      ename, job
  2  FROM          emp
  3  WHERE         job=      ← CLERK
  4                (SELECT job FROM emp WHERE empno= 7369);
  5  AND          sal >     ← 1100
  8                (SELECT sal FROM emp WHERE empno= 7876);
```

ENAME	JOB
-----	-----
MILLER	CLERK

서브쿼리에서 그룹함수 사용

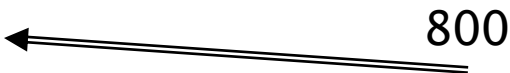
```
SQL> SELECT      ename, job, sal
  2  FROM        emp
  3  WHERE       sal = ← 800
  4              (SELECT MIN(sal) FROM emp );
```

ENAME	JOB	SAL
-----	-----	-----
SMITH	CLERK	800

서브쿼리를 가진 **HAVING** 절

- 오라클 서버는 먼저 서브쿼리를 실행
- 오라클 서버는 결과를 메인 쿼리의 **HAVING** 절에 리턴 함

```
SQL> SELECT      deptno, MIN(sal)
  2  FROM          emp
  3  GROUP BY      deptno
  4  HAVING         MIN(sal) > 800
  5              (SELECT MIN(sal) FROM emp WHERE deptno = 20);
```

A diagram consisting of two parallel arrows pointing from the number '800' to the 'MIN(sal)' expression in the HAVING clause, illustrating that the subquery result is used for comparison.

이 문장에서 잘못된 점은?

```
SQL> SELECT          empno, ename
  2  FROM             emp
  3  WHERE            sal =
  4                (SELECT MIN(sal) FROM emp GROUP BY deptno);
```

ERROR:

ORA-01427 : single-row subquery returns more
than one row

No rows selected

이 문장은 실행될까?

```
SQL> SELECT          ename, job  
  2  FROM            emp  
  3  WHERE            job =  
  4      (SELECT job  FROM emp WHERE ename='SMYTH');
```

No rows selected

다중 행 서브쿼리

- 하나 이상의 행을 리턴
- 다중 행 비교 연산자를 사용

연산자	의미
IN	목록의 어떤 구성원과 같다
ANY	값을 서브쿼리에 의해 리턴된 각각의 값과 비교한다
ALL	값을 서브쿼리에 의해 리턴되는 모든 값과 비교한다

다중 행 서브쿼리에서 **ANY** 연산자 사용

```
SQL> SELECT      empno, ename, job
  2  FROM          emp
  3  WHERE         sal > ANY
  4                (SELECT sal FROM emp WHERE job='CLERK')
  5  AND           job <> 'CLERK';
```

EMPNO	ENAME	JOB
7654	MARTIN	SALESMAN
7521	WARD	SALESMAN

다중 행 서브쿼리에서 **ALL** 연산자 사용

```
SQL> SELECT          empno, ename, job
  2   FROM            emp
  3   WHERE            sal > ALL
  4                   (SELECT avg(sal) FROM emp GROUP BY deptno);
```

EMPNO	ENAME	JOB
-----	-----	-----
7839	KING	PRESIDENT
7566	JONES	MANAGER
7902	FORD	ANALYST
7902	SCOTT	ANALYST

요약

- 서브쿼리는 다른 **SQL** 문장의 절에 내장된 **SELECT**문으로 미지정된 값을 근거로 할 때 유용함
 - 데이터의 한 행을 **=, <>, >, >=, <, <=** 같은 단일 행 연산자를 포함하는 메인 문장으로 전달할 수 있음
 - 데이터의 다중 행을 **IN** 과 같은 다중 행 연산자를 포함하는 메인문장으로 전달할 수 있음
 - 서브쿼리는 오라클 서버에 의해서 먼저 처리된 다음에 **WHERE** 또는 **HAVING** 절이 그 결과를 사용
 - 그룹함수를 포함할 수 있음