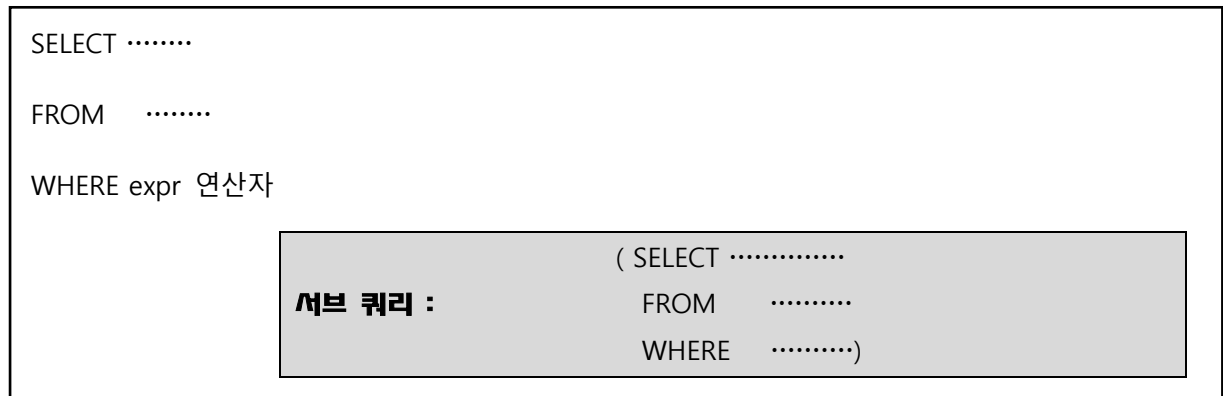


1. 서브 쿼리

부속 질의어로 SQL 명령문(SELECT문, INSERT문, UPDATE문, DELETE문, CREATE문 등) 내에 사용되는 SELECT문을 말한다. SQL 명령문 내에 서브 쿼리가 사용되면 서브 쿼리를 먼저 실행하여 그 결과 값을 메인 쿼리(main query)로 반환하고, 반환된 값을 메인 쿼리에서 사용한다.



☞ 서브 쿼리 사용시 주의사항

- 서브 쿼리는 괄호()안에서 사용한다.
- 서브 쿼리가 반환하는 행의 수는 expr 또는 연산자가 기대하는 값의 수와 일치해야 한다.
- 서브 쿼리가 반환하는 칼럼의 수는 expr 또는 연산자가 기대하는 칼럼의 수와 일치해야 한다.
- 서브 쿼리는 ORDER BY 절을 사용하지 않는다.
- 메인 쿼리의 FROM절에 있는 테이블의 칼럼명은 서브 쿼리 내에 사용될 수 있으나, 서브 쿼리의 FROM절에 있는 테이블의 칼럼명은 메인 쿼리에서 사용할 수 없다.
- 서브 쿼리는 최대 255개까지 사용할 수 있다.

※ 서브 쿼리의 반환 행에 따라 단일 행 서브 쿼리와 다중 행 서브 쿼리가 있다.

2. 단일 행 서브 쿼리

단일 행 서브 쿼리란 서브 쿼리의 결과 값이 단일 행이며, 단일 칼럼이나 다중 칼럼으로 반환된다.

예) 1. SG_Scores 테이블의 'L1031' 'SQL' 과목의 행들을 출력하고, 'L1031' 과목의 평균점수보다 높은 점수를 조회해보자.

2. SG_Scores 테이블의 'L1031' 'SQL' 과목에서 최고점을 받은 학생들의 학번, 과목코드, 성적, 성적취득일자를 조회해보자.

3. 다중 행 서브 쿼리

다중 행 서브 쿼리는 다중 행으로 반환되는 서브 쿼리이다. 다중 행 서브 쿼리는 다중 행 연산자와 함께 사용해야 한다. 다중 행 연산자로 IN, ANY, SOME, ALL, EXISTS 연산자가 있다.

연산자	의 미
IN	메인 쿼리의 비교 조건이 서브 쿼리의 결과 중에서 하나라도 일치하면 참이 되는 연산자
ANY, SOME	메인 쿼리의 비교 조건이 서브 쿼리의 결과 중에서 하나 이상 일치하면 참이 되는 연산자
ALL	메인 쿼리의 비교 조건이 서브 쿼리의 결과와 모든 값이 일치하면 참이 되는 연산자
EXISTS	메인 쿼리의 비교 조건이 서브 쿼리의 결과 중에서 하나라도 반환하면 참이 되는 연산자

3.1. IN 연산자

IN 연산자는 메인 쿼리의 비교 조건이 서브 쿼리의 결과와 비교하여 하나라도 일치하면 참이 되는 연산자이다.

예) SG_Scores 테이블의 년도별 최고점을 받은 학생들의 학번, 과목코드, 성적, 성적취득일자를 조회

3.2. ANY, SOME 연산자

메인 쿼리의 비교 조건이 서브 쿼리의 결과 중에서 하나 이상 일치하면 참이 되는 연산자이다.

예) SG_Scores 테이블의 'L1031' 'SQL' 과목에서 성적이 최하점보다 높은 점수의 학생에 대하여 학번, 과목코드, 성적, 성적취득일자를 조회

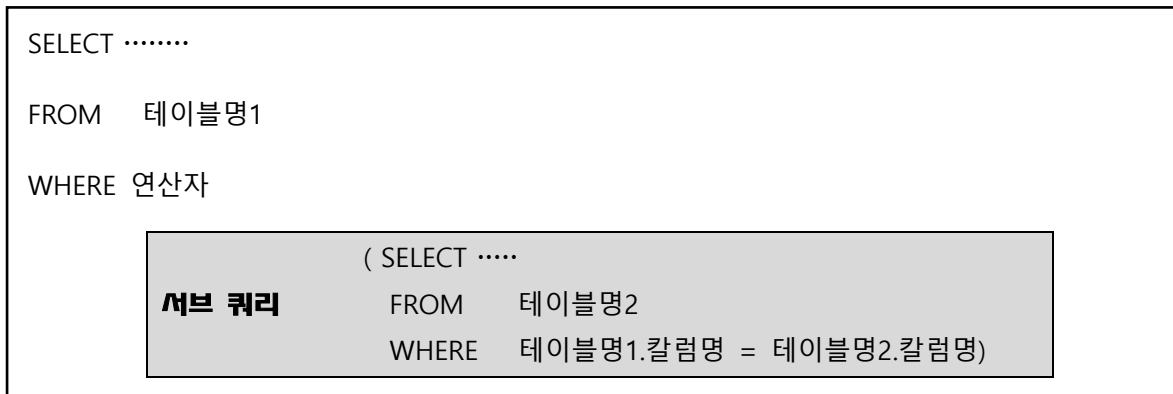
3.3. ALL 연산자

메인 쿼리의 비교조건이 서브 쿼리의 결과와 모든 값이 일치하면 참이 되는 연산자이다.

예) SG_Scores 테이블의 'L1031' 'SQL' 과목에서 년도별 최고점을 받은 학생들의 학번, 과목코드, 성적, 성적취득 일자를 조회

4. 상호관련 부속질의어

서브 쿼리에서 메인 쿼리의 값을 참조하고, 서브 쿼리의 결과 값을 다시 메인 쿼리가 참조하는 부속 질의어를 말한다. 상호 관련 부속 질의어는 WHERE절에 EXISTS, ANY, ALL 연산자를 이용하며, 서브 쿼리의 결과 행에 따라 참(true)이 되거나 거짓(false)이 된다.



☞ 상호관련 부속 질의어는

서브 쿼리에서 메인 쿼리(테이블1)의 칼럼을 참조할 수 있다. 그러나 서브 쿼리(테이블2)의 칼럼은 메인 쿼리에서 참조하지 못한다.

서브 쿼리의 WHERE절 검색 조건에 테이블1.칼럼명 = 테이블명2.칼럼명을 기술하면, 반환되는 값은

- 테이블1.칼럼명의 첫 번째 행과 테이블2.칼럼명의 첫 번째 행의 값부터 마지막 행이 될 때까지 순서대로 비교하여 동일한 값이면 반환한다.
- 테이블1.칼럼명의 두 번째 행과 테이블2.칼럼명의 값을 차례대로 비교하여 동일한 값이면 반환한다.
- 이런 방법으로 테이블1. 칼럼명의 마지막 행이 될 때까지 반복한다.

메인 쿼리의 EXISTS 연산자는 서브 쿼리에서 적어도 하나 이상의 행을 반환하면 참이 되고, 아무 행도 반환하지 않으면 거짓이 된다.

예) 1. Professor 테이블과 Course 테이블을 이용하여 강의를 담당하고 있는 교수의 명단 조회

2. Professor 테이블과 Course 테이블을 이용하여 강의 담당하고 있지 않은 교수의 명단 조회

5. 서브 쿼리를 이용한 테이블 생성

CREATE TABLE문에 서브 쿼리를 사용하면 테이블의 데이터를 전체 혹은 일부분을 복사하여 새로운 테이블을 생성할 수 있다.

문 법	CREATE TABLE 테이블명 [(칼럼명_1, 칼럼명_2, ...)] AS SELECT문;
----------------	--

※ 기술 방법

- CREATE TABLE문의 칼럼 수와 SELECT절의 칼럼 수가 동일해야 한다.
- SELECT절의 첫 번째 칼럼 값이 CREATE TABLE문의 첫 번째 컬럼 값으로 저장된다.
- CREATE TABLE문의 칼럼명은 생략하면, SELECT절의 칼럼명을 참조한다. 단, 수식이나, 함수명을 사용하면 생략할 수 없다.

예) 1. Student 테이블에서 '컴공'학과에 재학중인 학생들을 Computer_Student 테이블을 생성하고, 저장하시오.

2. 서브 쿼리를 이용한 빈 테이블 생성 : Course 테이블과 동일한 T_Course 빈 테이블을 만들어보자.