

목차

- I. 데이터베이스 이해
- II. 데이터베이스를 구성하는 객체 이해
- III. SQL 기본
- IV. SQL 함수
- V. 그룹 쿼리와 집합 연산자
- VI. 조인(Join)과 서브쿼리(SubQuery)
- VII.PL/SQL

목차

VI. 조인(Join)과 서브 쿼리 (SubQuery)

1. 조인의 종류
2. 내부 조인과 외부 조인
3. ANSI 조인
4. 서브 쿼리

VI.조인과 서브쿼리

1. 조인의 종류

- ✓ 관계형 데이터베이스에서 SQL을 이용해 ‘관계’를 맺는 방법이 바로 조인이다.
- ✓ 관계형 데이터베이스는 최소한의 데이터를 테이블에 담고 있는데, 원하는 정보를 테이블에서 추출하려면 대부분 한 개 이상의 테이블이나 뷰에서 데이터를 읽어 와야 한다.
이때 테이블 간의 연결고리로 관계를 조인을 통해 맺고 데이터를 추출한다.
- 조인 연산자에 따른 구분 : 동등 조인, 안티 조인
- 조인 대상에 따른 구분 : 셀프 조인
- 조인 조건에 따른 구분 : 내부 조인, 외부 조인, 세미 조인, 카타시안 조인
- 기타 : ANSI 조인
- ✓ 조인의 종류가 많지만 각 명칭을 암기할 필요는 없다 (각각의 명칭이 다른 뿐)
- ✓ 일반적으로 개발자 관점에서 내부 조인과 외부 조인만 구분할 수 있으면 충분하다.

VI.조인과 서브쿼리

2. 내부 조인과 외부 조인

-동등 조인 (EQUI-JOIN)

- ✓ 가장 기본이 되며 일반적인 조인 방법이 바로 동등 조인(EQUI-JOIN) 이다.
- ✓ 동등 조인은 WHERE 절에서 등호 (=) 연산자를 이용해 2개 이상의 테이블이나 뷰를 연결한 조인.
- ✓ 등호 연산자를 사용한 WHERE절 조건에 만족하는 데이터를 추출하는 조인.

EQUI JOIN

A와 B테이블의 두 테이블에서 공통된 값을 가진 컬럼을 등호 연산자로 연결해 주는 조인
조인 조건 결과가 참(TRUE)에 해당하는 두 컬럼의 값이 같은 행을 추출.

```
SELECT *  
FROM 테이블 A  
    , 테이블 B  
WHERE A.column1 = B.column1;
```

VI.조인과 서브쿼리

2. 내부 조인과 외부 조인

-세미 조인 (SEMI-JOIN)

- ✓ 서브 쿼리를 사용해 서브 쿼리에 존재하는 데이터만 메인 쿼리에서 추출하는 조인방법
- ✓ IN과 EXISTS 연산자를 사용한 조인이다.

EXISTS 사용

서브 쿼리에 있는 테이블을 B, 메인 쿼리에 사용된 테이블을 A라고 한다면
세미조인은 B 테이블에 존재하는 A 테이블의 데이터를 추출하는 조인

```
SELECT *  
FROM 테이블 A  
WHERE EXISTS(SELECT *  
              FROM 테이블 B  
              WHERE a.column1 = b.column1  
              AND b.column2 > 3000)
```

IN 사용

IN 연산자는 OR 조건으로 변환 할 수 있는데,
즉, '이것이거나 저것이거나' 로 할 수 있다.

```
SELECT column1  
      , column2  
FROM 테이블 a  
WHERE a.column1 IN (SELECT b.column1  
                   FROM 테이블 b  
                   WHERE b.column2 > 3000)
```

VI.조인과 서브쿼리

2. 내부 조인과 외부 조인

-안티 조인 (ANTI-JOIN)

- ✓ 안티란 말에서 유추할 수 있듯이 안티 조인 (ANTI-JOIN)은 서브 쿼리의 B 테이블에 없는 메인쿼리의 A 테이블의 데이터만 추출하는 조인방법.
- ✓ NOT IN 이나 NOT EXISTS 연산자를 사용한다. 즉 세미조인과 반대 개념이다.

NOT IN 사용

서브쿼리 B 테이블에 없는 메인쿼리 A 테이블의 데이터만 추출하는 조인 방법

```
SELECT *  
FROM 테이블 A  
      , 테이블 B  
WHERE a.column1 = b.column1  
      AND a.column1 NOT IN(SELECT column1  
                           FROM 테이블  
                           WHERE column2 IS NOT NULL)
```

NOT EXISTS사용

서브쿼리 B 테이블에 없는 메인쿼리 A 테이블의 데이터만 추출하는 조인 방법

```
SELECT count(*)  
FROM 테이블 A  
WHERE NOT EXISTS ( SELECT 1  
                   FROM 테이블 C  
                   WHERE a.column1 = c.column1  
                   AND c. column1 IS NOT NULL);
```

VI.조인과 서브쿼리

2. 내부 조인과 외부 조인

-외부조인(OUTER JOIN)

- ✓ 조인 조건을 만족하는 데이터 뿐만 아니라, **어느 한 쪽 테이블에** 조인 조건에 명시된 컬럼에 값이 없거나 (NULL 이더라도) 해당 로우가 아예 없더라도 데이터를 모두 추출한다.

OUTER JOIN

조인 조건에서 데이터가 없는 테이블의 컬럼에 (+)기호

```
SELECT *  
FROM 테이블 A  
    , 테이블 B  
WHERE a.column1 = b.column1(+)
```

OUTER JOIN

외부 조인의 조인 조건이 여러 개일 때 모든 조건에 (+)를 붙인다.

```
SELECT *  
FROM 테이블 A  
    , 테이블 B  
WHERE a.column1 = b.column1(+)  
      AND a.column2 = b.column2(+)
```

- ✓ 한번에 한 테이블에만 외부조인을 할 수 있다.
- ✓ (+)연산자가 붙은 조건과 OR, IN 연산자를 같이 사용할 수 없다.(단 IN 절에 포함되는 값이 1면 가능)

VI.조인과 서브쿼리

2. 내부 조인과 외부 조인

-카타시안 조인(CATASIAN PRODUCT)

- ✓ WHERE 절에 조인 조건이 없는 조인을 말하며
- ✓ 조인조건이 없으므로 그 결과는 두 테이블 건수의 곱이다. 즉 A테이블이 N1 건 고 B테이블이 N2 건이라면 결과는 $N1 * N2$ 건 수가 된다.

CATASIAN PRODUCT

```
SELECT A.column1, A.column2  
      , B.column2, B.column2  
FROM 테이블 A  
      , 테이블 B
```


VI.조인과 서브쿼리

3. ANSI 조인

- ✓ 안시(ANSI)란 American National Standards Institute의 줄임말이다. 미국국립표준협회라는 뜻이다. 이곳에서 정한 개량된(enganced) 표준입력 및 표준출력에 관한 규정을 흔히 안시(ANSI)라고 말한다.
- ✓ ANSI SQL 문법을 사용한 조인을 말하며 지금까지 설명한 조인을 모두 ANSI SQL 을 사용해 변환이 가능하다.
- ✓ 기존 문법과 ANSI 조인의 차이점은 조인 조건이 WHERE 절이 아닌 FROM 절에 들어간다는 점이다.

-ANSI 내부조인

ANSI INNER JOIN	일반 SQL 내부조인의 동등조인과 동등한 결과
<pre>SELECT A.column1 , B.column1 FROM 테이블 A INNER JOIN 테이블 B ON (A.column1 = b.column1) WHERE 조건</pre>	

- ✓ 만약 조인 조건 컬럼이 두 테이블 모두 동일 하다면 ON 대신 USING 절을 사용 할 수 있다.

VI.조인과 서브쿼리

3. ANSI 조인

-ANSI 외부 조인

- ✓ 기존 문법에서는 기준 테이블과 대상 테이블(데이터가 없는 테이블)에서 대상 테이블쪽 조인 조건에 (+)를 붙였지만, ANSI 외부 조인은 FROM 절에 명시된 테이블 순서에 입각해 먼저 명시된 테이블을 기준으로 LEFT 또는 RIGHT를 붙인다.

ANSI OUTER JOIN	일반 SQL 외부조인의 동등조인과 동등한 결과
<pre>SELECT A.column1 , B.column1 FROM 테이블 A LEFT(RIGHT) [OUTER] JOIN 테이블 B ON (A.column1 = b.column1) WHERE 조건</pre>	✓ OUTER라는 키워드는 생략 가능하다.
ANSI OUTER JOIN	LEFT, RIGHT 조인을 합친 결과
<pre>SELECT A.column1 , B.column1 FROM 테이블 A FULL [OUTER] JOIN 테이블 B ON (A.column1 = b.column1) WHERE 조건</pre>	

VI.조인과 서브쿼리

3. ANSI 조인

-CROSS 조인

✓ WHERE 절에 조인 조건을 명시하지 않은 카타시안 조인이 있는데 ANSI조인은 CROSS 조인이라고 한다.

ANSI CROSS JOIN	일반 카타시안 조인과 같은결과
<pre>SELECT A.column1 , B.column1 FROM 테이블 A CROSS JOIN 테이블 B</pre>	

감사합니다!