**조인(join)**

테이블과 테이블을 연결

테이블 간 연결을 위해서는 연결고리 역할을 하는 컬럼이 필요

컬럼 이름이 같을 필요는 없지만, 같게 만드는 것이 좋긴 함

조인 컬럼은 한 개 이상으로 구성될 수 있음.

방식에 따라, 내부조인 / 외부조인

**내부조인( INNER JOIN )**

가장 기본적인 방식

테이블 간 조인 컬럼이 같은 건을 가져옴

WHERE 절에 JOIN조건 명시

일반적으로 동등연산자(=) 사용

구성)

**SELECT** a.employee\_id, a.first\_name, a.department\_id, b.department\_name

**FROM** employees a

, departments b

**WHERE** a.department\_id = b.department\_id

**ORDER BY** a.department\_id;

FROM절에 조인 참여할 테이블 모두 적음

테이블에 alias를 주는 게 편함

Alias명.컬럼명 형태로 사용

조인 조건을 만족하는 데이터만 조회 (WHERE절에 있는 조건이 모두 만족해야 조회됨)

두 테이블에 모두 컬럼이 있을 땐, 꼭 Alias 가 붙어야 어느 테이블인지 알 수있다.(오류남)

여러 테이블 조인 예시 )

**SELECT** a.employee\_id, a.first\_name || ' ' || a.last\_name emp\_names

, b.job\_title

, c.department\_name

, d.location\_id, d.street\_address, d.city, d.state\_province

**FROM** employees a

, jobs b

, departments c

, locations d

**WHERE** a.job\_id = b.job\_id

**AND** a.department\_id = c.department\_id

**AND** c.location\_id = d.location\_id

**ORDER BY** 1;

**외부조인 ( OUTER JOIN )**

조인 조건을 만족하지 않는 데이터까지 포함해 조회

A, B 테이블 조인 조건에 부합하지 않는 테이블 데이터도 조회됨.

조인 조건에 부합하지 않는 데이터를 뽑으려는 테이블의 반대쪽에 **(+)**를 붙임

조인 조건을 만족하지 않는 A테이블 데이터 조회 시,

A.TABLE = B.TABLE**(+)**

조인 조건을 만족하지 않는 B테이블 데이터 조회 시,

A.TABLE**(+)** = B.TABLE

양쪽 다 (+)를 붙일 수 없음 (오라클문법에서는)

외부 조인 시, 조인조건에 (+)를 붙이는 것은 오라클 전용 문법(다른곳에서는 오류남)

**SELECT** a.employee\_id emp\_id

, a.department\_id a\_dept\_id

, b.department\_id b\_dept\_id

, b.department\_name dept\_name

**FROM** employees a, departments b

**WHERE** a.department\_id = b.department\_id **(+)**

**ORDER BY** a.department\_id;

**ANSI 조인 ( ANSI JOIN )**

ANSI 표준 문법으로 작성한 조인 방법

내부조인. 외부조인을 모두 ANSI문법에 맞게 작성

내부 조인 : INNER JOIN

외부 조인 : LEFT OUTER JOIN , RIGHT OUTER JOIN , FULL OUTER JOIN

INNER, OUTER는 생략가능( 근데 일단쓰자)

FULL OUTER JOIN은 ANSI문법으로만 구현 가능 ( 오라클에서 (+)을 양쪽에 붙이는 거)

FROM 절에 어떤 조인을 쓸건지 기술하고, 조인 조건은 ON 다음에 기술

조인 조건 이외에 다른 조건은 WHERE 절에 기술

LEFT, RIGHT 구분?

FROM **테이블1** LEFT OUTER JOIN 테이블2

FROM 테이블1 RIGHT OUTER JOIN **테이블2**

조인 기준으로 왼쪽/오른쪽 테이블의 데이터를 조건에 맞지 않아도 가져온다.

ANSI INNER JOIN 예시\_)

**SELECT** a.employee\_id emp\_id

, a.department\_id a\_dept\_id

, b.department\_id b\_dept\_id

, b.department\_name dept\_name

**FROM** employees a

**INNER JOIN** departments b

**ON** a.department\_id = b.department\_id

**ORDER BY** a.department\_id;

ANSI OUTER JOIN 예시\_)

**SELECT** A. EMPLOYEE\_ID

, A.FIRST\_NAME||’ ‘||A.LAST\_NAME

, B.DEPARTMENT\_ID

, B.DEPARTMENT\_NAME

**FROM** EMPLOYEES A

**LEFT OUTER JOIN** DEPARTMENTS B

**ON** A.DEPARTMENT\_ID = B.DEPARTMENT\_ID

**ORDER BY** A.EMPLOYEE\_ID;

ANSI JOIN 여러 개 예시\_)

**SELECT** a.employee\_id,

a.first\_name || ' ' || a.last\_name emp\_names,

b.job\_title,

c.department\_id,

c.department\_name

**FROM** employees a

**INNER JOIN** jobs b

**ON** a.job\_id = b.job\_id

**INNER JOIN** departments c

**ON** a.department\_id = c.department\_id

**ORDER BY** 1;

ANSI JOIN 여러 개 + WHERE 절 예시\_)

**SELECT** a.employee\_id

, a.first\_name || ' ' || a.last\_name emp\_names, a.job\_id

, b.job\_id, b.job\_title

, c.department\_id, c.department\_name

**FROM** employees a

**INNER JOIN** jobs b

**ON** a.job\_id = b.job\_id

**INNER JOIN** departments c

**ON** a.department\_id = c.department\_id

**WHERE** b.job\_id = 'SH\_CLERK'

**ORDER BY** 1;

**Cartesian Product**

조인 조건이 없는 조인

조인 참여 테이블을 FROM절에 기술하고, WHERE절에 조인조건 기술X

두테이블 기준 모든 조합이 조회됨

A테이블(5건), B테이블(6건) >> 5\*6= 30건

이거 보면 **조인 조건이 누락됐다** 정도 깨달으면 됨.

**셀프 조인 ( SELF JOIN )**

**SELECT** a.employee\_id ,a.first\_name || ' ' || a.last\_name emp\_name

,a.manager\_id

,b.first\_name || ' ' || b.last\_name manager\_name

**FROM** employees a

**INNER JOIN** employees b

**ON** a.manager\_id = b.employee\_id

**ORDER BY** 1;