

3. JOIN : 2개 이상의 테이블을 연결하여 데이터를 검색하는 방법

(1) EQUI JOIN: 가장 많이 사용하는 방법으로, 연결하는 테이블의 값이 일치되는 조건만 JOIN

<ex- 근무지(LOC)가 CHICAGO인 사람의 이름, 업무, 급여, 부서명, 근무지 필드 출력하기>

```
SELECT ENAME 이름, JOB, SAL 급여, DNAME 부서명, LOC FROM EMP E, DEPT D -- EMP와 DEPT 테이블 JOIN
WHERE E.DEPTNO = D.DEPTNO AND SAL >= 2000;
```

-- DEPTNO(부서 코드) 필드는 EMP와 DEPT 모두에 존재하므로,
각 테이블에 있는 부서코드 필드가 같다는 조인 조건을 걸어준다.

(2) NON-EQUI JOIN: 조인 조건이 특정 범위 내에 있는지 보기 위해

'=' 이외의 비교 연산자를 사용한다.

<그림1- SALGRADE 테이블>

<ex- 사원 SCOTT의 사원명, 급여, 급여 등급을

SALGRADE 테이블과 JOIN하여 불러오기>

	GRADE	LOSAL	HISAL
1	1	700	1200
2	2	1201	1400
3	3	1401	2000
4	4	2001	3000
5	5	3001	9999

```
SELECT ENAME, SAL, GRADE FROM EMP, SALGRADE
```

```
WHERE ENAME = 'SCOTT' AND SAL BETWEEN LOSAL AND HISAL;
```

-- SALGRADE 테이블에 제시된 각 급여등급에 따른 최저급여(LOSAL)와 최고급여(HISAL)를 SCOTT의 급여 등급 조건으로 제시 (그림1 참고)

(3) SELF JOIN: 동일한 두 테이블에 서로 다른 별명을 붙여 JOIN하는 것

<ex- 사원명과 각 사원들의 상사명 출력하기>

```
SELECT W.ENAME, M.ENAME FROM EMP W, EMP M -- 사원 EMP테이블(W)과, 상사 EMP테이블(M) JOIN
```

```
WHERE W.MGR = M.EMPNO; -- W.EMP테이블의 상사사번(MGR)과 M.EMP테이블의 사번을 동일하도록 설정
```

(4) OUTER JOIN: JOIN중 NULL값이라서 배제된 행을 결과에 포함시키도록 설정하는 JOIN

<ex = 사원명, 사원들의 상사명을 출력하되, 상사가 없는 사람들도 함께 출력하기>

```
SELECT W.ENAME, NVL(M.ENAME, '-CEO-') FROM EMP W, EMP M
```

-- NULL값(상사가 없는 경우)에 대체할 값(-CEO-)을 설정

```
WHERE W.MGR = M.EMPNO (+); -- NULL값이 있을 필드에 (+)를 입력하면 그 자리도 포함하여 출력된다.
```