

Join의 필요성

특정부서번호에 대한 부서이름은 무엇인지 dept 테이블에 있고,
특정 사원에 대한 부서명을 알아내기 위해서는 부서 테이블에서 정보를 얻어와야 한다

Join을 하지 않는 경우 다음의 두 SQL문을 통해 알아내야 하는 번거로움이 있다.

```
SELECT ename, deptno  
FROM emp  
ORDER BY deptno;
```

```
SELECT deptno, dname  
FROM dept;
```

다행히도 SQL에서는 두 개 이상의 테이블을 결합해야만
원하는 결과를 얻을 수 있을 때 한번의 질의로 원하는 결과를 얻어내는

Join 기능을 제공.

Cross Join

Cross Join은 특별한 키워드 없이 다음과 같이
FROM절에 두 개 이상의 테이블을 기술하는 것.
SELECT * FROM EMP, DEPT;

Cross Join의 결과 얻어지는 컬럼의 수는 사원 테이블의 컬럼의 수(8)와
부서 테이블의 컬럼의 수(3)를 더한 것이므로 11이 됩니다.
로우 수는 사원 한 명에 대해서 DEPT 테이블의 4개의 로우와 결합되기에 56개(14*4).

Cross Join의 결과를 보면 사원 테이블에 부서에 대한 상세정보가 결합되긴 했지만, 조인 될 때
아무런 조건을 제시하지 않았기에 사원 한명에 대해서 DEPT 테이블의 4개의 로우와 결합된 형태
이기에 Cross Join의 결과는 아무런 의미를 갖지 못 한다.

```
SELECT * FROM EMP, DEPT WHERE ENAME='SCOTT';  
SELECT * FROM DEPT WHERE DEPTNO=20;
```

조인 결과가 의미를 갖으려면 조인할 때 조건을 지정. 조인 조건에 따라 조인의 종류가 결정.

Equi Join : 동일한 컬럼을 기준으로 조인

Non-Equi Join : 동일한 컬럼없이 다른 조건을 사용하여 조인

Outer Join : 조인 조건에 만족하지 않는 행도 나타나는 조인

Self Join : 한 테이블 내에서 조인.

Equi Join

EQUI JOIN은 가장 많이 사용하는 조인 방법으로서 조인 대상이 되는 두 테이블에서 공통적으로 존재하는 컬럼의 값이 일치되는 행을 연결하여 결과를 생성하는 조인 방법

다음은 사원 정보를 출력할 때 각 사원들이 소속된 부서의 상세 정보를 출력하기 위해서 두 개의 테이블을 조인한 예

```
SELECT *  
FROM emp, dept  
WHERE emp.deptno=dept.deptno;
```

사원(EMP) 테이블과 부서(DEPT) 테이블의 공통 컬럼인

DEPTNO의 값이 일치(=)되는 조건을 WHERE 절에 기술하여 사용.

테이블을 조인하려면 일치되는 공통 컬럼을 사용해야 하고

컬럼의 이름이 같게 되면 혼동이 오기 때문에

컬럼 이름 앞에 테이블 이름을 기술.

Non-Equi Join

Non-Equi Join은 조인 조건에 특정 범위 내에 있는지를 조사하기 위해서

WHERE 절에 조인 조건을 = 연산자 이외의 비교 연산자를 사용합니다.

Self Join

조인은 두 개 이상의 서로 다른 테이블을 서로 연결하는 것뿐만 아니라,

하나의 테이블 내에서 조인을 해야만 원하는 자료를 얻는 경우가 생깁니다.

Self Join이란 말 그대로 자기 자신과 조인을 맺는 것.

Quter Join

조인 조건에 만족하지 못하였더라도 해당 로우를 나타내고 싶을 때에 사용하는 것이

외부 조인(Outer Join)입니다.

외부 조인(Outer Join)은 NULL 값이기에 배제된 행을 결과에 포함시킬 수 있으며

다음과 같이 “(+)” 기호를 조인 조건에서 정보가 부족한 칼럼 이름 뒤에 덧붙입니다.

사원 번호(EMPNO)가 NULL인 사원은 없으므로

manager.empno 뒤에 “(+)” 기호를 덧붙입니다.

SELECT WORKER.ENAME, MANAGER.ENAME 상사

FROM EMP WORKER, EMP MANAGER

WHERE WORKER.MGR=MANAGER.EMPNO(+);