

```

1  REM Author :
2  REM Date :
3  REM Objective : Chapter 6. Join
4  REM Environment : CentOS 6.5, SQLGate 2010 for Oracle, Oracle 11g Enterprise Edition
   11.2.0
5
6  REM 조인(JOIN)
7  --한개 이상의 테이블로부터 데이터를 조회하는 것
8  --주로 Primary-Key와 Foreign-Key의 관계를 가진 컬럼을 소유하고 있는 테이블을 통한
   검색 시 사용
9  --Oracle 9i에서는 EQUI Join, Non-Equi Join, Outer Join, Self Join을 변경하여
10 --Cross Join, Natural Join, Using 절 Join, ...이 새로 소개됐다.
11
12 REM Join의 종류
13
14 REM Cross Join
15 --Cartesian Join(카티시안 곱)
16 --조인 조건이 부적합하거나 조인조건을 완전히 생략한 경우 행의 모든 조합을 표시
17 --첫번째 테이블의 모든 행이 두번째 테이블의 모든 행에 조인됨
18 --너무 많은 행을 생성하기 때문에 결과를 사용할 수 없다.
19 --모든 행을 조합해야 하는 경우가 아니라면 반드시 WHERE절에 적합한 조인 조건을
   포함시켜야
20 --CROSS JOIN
21
22 --SELECT empno, ename, dname
23 --FROM emp, dept;
24
25 --SELECT empno, ename, dname
26 --FROM emp
27 --CROSS JOIN dept;
28
29
30 REM Natural Join
31 --EQUI Join (등가조인;단순조인;내부조인)
32 --2개 이상의 테이블이 공통되는 컬럼에 의해 논리적으로 결합되는 조인기법
33 --WHERE절에 사용된 공통된 컬럼들이 동등 연산자(=)에 의해 비교
34
35 --SELECT empno, ename, dname
36 --FROM emp, dept
37 --WHERE emp.deptno = dept.deptno;
38
39 --사원이름 KING 의 부서이름과 근무지를 출력하시오.
40 --SELECT empno, ename, dname
41 --FROM emp, dept
42 --WHERE emp.deptno = dept.deptno;
43 --AND ename = 'KING';
44
45 --테이블 별칭 사용
46   ---별칭의 Guideline
47   --열 이름을 지정하는 경우 시간이 많이 걸리며 테이블 이름이 길때는 더욱 오래 걸림
48   --테이블 이름 대신 테이블 별칭을 사용가능
49   --SQL 코드를 적게 작성해도 되므로 메모리 사용이 줄어듦
50
51 --지침
52 --테이블 명칭은 최대 30개까지 가능하지만 짧을수록 좋다
53 --FROM 절에서 특정 테이블이름에 대해 별칭을 사용했다면 SELECT 문에서도 테이블
   이름을 대신한다.
54 --의미있는 이름을 주자.
55 --별칭은 현재 SELECT에서만 유효하다.
56
57 --SELECT e.empno, e.ename, d.dname
58 --FROM emp e, dept d
59 --WHERE e.deptno = d.deptno;
60
61 --SELECT empno, ename, dname
62 --FROM emp
63 --NATURAL JOIN dept;
64
65 --SELECT empno, ename, job, dname, loc
66 --FROM emp
67 --NATURAL JOIN dept
68 --WHERE empno = 7900;
69

```

```

70 REM JOIN ~ USING
71 --USING을 사용하지 않았다면 Oracle Server는 각 테이블에 공통된 컬럼을 자동으로
  검색하여 비교
72 --USING을 사용하면 USING절에 정의된 컬럼을 기준으로 Natural 조인이 발생한다.
73
74 --SELECT empno, ename, dname
75 --FROM emp JOIN dept
76 --USING (deptno)
77 --WHERE empno = 7900;
78
79 REM NON-EQUI JOIN (비등가조인)
80 --emp table과 salgrade table간의 관계
81 --두개의 테이블사이엔 직접 대응하는 열이 없다.
82 --이 관계는 =을 제외한 연산자를 사용하여 형성
83
84 --SELECT e.ename, e.sal, s.grade
85 --FROM emp e, salgrade s
86 --WHERE e.sal BETWEEN s.losal AND s.hisal;
87
88 REM OUTER JOIN
89 --포괄조인
90 --일치하는 항목이 없는 레코드도 질의할 수 있다.
91 --SELECT e.ename, e.deptno, d.dname
92 --FROM emp e, dept d
93 --WHERE e.deptno = d.deptno;
94 --부서 OPERATIONS는 그 부서에서 일하는 사원이 없다. 그래서 표시되지 않는다.
95
96 --포괄조인 연산자 '+'를 사용하여 누락된 행을 반환할 수 있다.
97 --정보가 부족한 조인 옆에 넣는다.
98 --하나이상의 널 행을 생성하여 완전한 테이블의 하나 이상의 행과 조인할 수 있다.
99 --IN 연산자는 사용할 수 없으며 OR 연산자를 사용해 다른 조건에 링크할 수 없다.
100
101 --SELECT e.ename, e.deptno, d.dname
102 --FROM emp e, dept d
103 --WHERE e.deptno(+) = d.deptno;
104
105
106 REM JOIN ~ ON
107 --SELF JOIN(자체조인)
108 --하나의 테이블이 2번 이상 반복적으로 사용되고 참조해야 할 컬럼이 자신의 테이블에
  있을 때
109 --각 사원의 관리자의 이름을 찾을 때
110 --예를 들어, Blake의 관리자 이름을 찾으려면 다음을 수행할 것이다.
111 --EMP 테이블의 ename 열에서 BLAKE를 찾는다.
112 --mgr 열에서 BLAKE의 관리자 번호를 찾는다. BLAKE의 관리자 번호는 7839이다.
113 --ename 열에서 empno 7839인 관리자 이름을 찾는다. King의 사원 번호가 7839이므로
  King이 Blake의 관리자이다.
114 --이 프로세스에서는 테이블을 두 번 검색한다. 첫 번째는 테이블의 ename 열에서
  BLAKE와 mgr 값 7839를 찾고 두 번째는 empno 열에서 7839를 찾고 ename 열에서 KING을
  찾는다.
115
116 --SELECT worker.ename, manager.ename
117 --FROM emp worker, emp manager
118 --WHERE worker.mgr = manager.empno;
119
120 --1.사원 이름 및 사원 번호를 해당 관리자 이름 및 관리자 번호와 함께 표시하고 열
  머리글을 각각 "사원이름", "사원번호", "관리자이름", "관리자번호"로 표시하시오.
121 SELECT employee.ename AS "사원이름",
122        employee.empno AS "사원번호",
123        employer.ename AS "관리자이름",
124        employer.empno AS "관리자번호"
125 FROM emp employee, emp employer
126 WHERE employee.mgr = employer.empno AND
127        employee.deptno = 10;
128
129 --2. emp table에서 self join하여 관리자를 출력하되, 아래의 형식에 맞게 출력하시오.
130 --BLAKE의 관리자는 KING이다.
131
132 --3.사원테이블에서 그들의 관리자보다 먼저 입사한 사원에 대해 이름, 입사일, 관리자
  이름, 관리자 입사일을 출력하시오.
133

```