과목 II

[SQL 기본]

제5장 GROUP BY, HAVING 절

[GROUP BY 절]
- 각 행을 특정 조건에 따라 그룹으로 분리하여 계산하도록 하는 구문식
- 그룹에 대한 조건은 WHERE 절에서 사용할 수 없음
- GROUP BY 절을 사용하면 데이터가 요약되므로 요약되기 전 데이터와 함께 출력할 수 없음
- GROUP BY 절에 명시되지 않은 컬럼 전달 불가

[HAVING 절]
- 그룹 함수 결과를 조건으로 사용할 때 사용하는 절
- WHERE 절을 사용하여 그룹을 제안할 수 없으므로 HAVING 절에 전달
- 내부적 연산 순서가 SELECT 절보다 먼저이므로 SELECT 절에서 선언된 Alias 사용 불가
- 순서상 WHERE 절을 먼저 수행, 원하는 데이터만 필터링 한 후 GROUP BY에 의해 그룹 연산을 수행한 뒤 HAVING 절에서 만족하는 데이터만 선택하여 출력

제6장 ORDER BY 절

[ORDER BY 절]
- 출력되는 행의 순서를 사용자가 변경하고자 할 때 ORDER BY 절을 사용
- ORDER BY 뒤에 명시된 컬럼 순서대로 정렬
- 오름차순(ASC - 생략 가능), 내림차순(DESC)로 전달
- 유일하게 SELECT 절에서 정의한 컬럼 별칭 사용 가능

복합 정렬
- 먼저 정렬한 값의 동일한 결과가 있을 경우 추가적으로 정렬 가능

```
ORDER BY SALARY DESC, HIRE_DATE ASC;
```

제7장 조인 절

[JOIN(조인)]

- 여러 테이블의 데이터를 사용하여 동시 출력하거나 참조할 경우 사용
- FROM 절에 조인할 테이블 나열
- 동일한 열의 이름이 여러 테이블에 존재할 경우 열 이름 앞에 테이블 이름이나 테이블 Alias 붙임
- ORACLE 표준은 테이블 나열 순서 중요 X, ANSI 표준은 OUTER JOIN 시 순서 중요
- WHERE 절에서 조인 조건을 작성 (ORACLE 표준)
- N개의 테이블을 조인하려면 최소 N-1 개의 조인 조건이 필요

[조인 종류]

1. 조건의 형태에 따라서
   1) EQUI JOIN : JOIN 조건이 동등 조건인 경우
   2) NON EQUI JOIN : JOIN 조건이 동등 조건이 아닌 경우

2. 조인 결과에 따라
   1) INNER JOIN : JOIN 조건에 성립하는 데이터만 출력하는 경우
   2) OUTER JOIN : JOIN 조건에 성립하지 않는 데이터도 출력하는 경우
      (LEFT/RIGHT/FULL OUTER JOIN 으로 나뉨)
3. NATURAL JOIN : 조인 조건 생략 시 두 테이블에 같은 이름으로 자연 연결되는 조인
4. CROSS JOIN : 조인 조건 생략 시 두 테이블의 발생 가능한 모든 행을 출력하는 조인
5. SELF JOIN : 하나의 테이블을 두 번 이상 참조하여 연결하는 조인

[EQUI JOIN]

- 조인 조건이 '=' 비교를 통해 같은 값을 가지는 행을 연결하여 결과를 얻는 조인 방법
- FROM 절에서 명시하는 테이블은 테이블 별칭(Alias) 사용 가능)
- WHERE 절에 두 테이블의 공통 컬럼에 대한 조인 조건을 나열

문법(ORACLE 기준)

```
SELECT 테이블1.컬럼, 테이블2.컬럼
FROM 테이블1, 테이블2
WHERE 테이블1.컬럼 = 테이블2.컬럼
```

[NON-EQUI JOIN]

- 테이블을 연결짓는 조인 컬럼에 대한 비교 조건이 '<', BETWEEN A AND B와 같이 '='조건이 아닌 연산자를 사용하는 경우의 조인 조건

[세 테이블 이상의 조인]

- 관계를 잘 파악하여 모든 테이블이 연결되도록 조인 조건 명시
- N개 테이블의 경우 최소 N-1 개의 조인 조건 필요

[SELF JOIN]
- 한 테이블 내 각 행 끼리 관계를 갖는 경우 조인 기법
- 한 테이블을 참조할 때 마다 명시해야 함
- 테이블 명이 중복되므로 반드시 테이블 별칭 사용

```
SELECT E1.EMPNO, E1.ENAME, E1.SAL
          E2.EMPNO, E2.ENAME, E2.SAL
FROM EMP E1, EMP E2
WHERE E1.MGR, E2.EMPNO
AND E1.SAL > E2.SAL;
```

제8장 표준 조인 절

표준 조인
-ANSI 표준으로 작성되는 INNER JOIN, CROSS JOIN, NATURAL JOIN, OUTER JOIN을 말함

INNER JOIN
- 내부 조인이라고 하며, 조인 조건이 일치하는 행만 추출(ORACLE 조인 기본)
- USING 이나 ON 조건절을 필수적으로 사용

ON절
- ON 조건의 괄호는 옵션(생략 가능)
- 컬럼명이 같을 경우 테이블 이름이나 별칭을 사용하여 명확하게 지정(테이블 출처 명확히)
- ON 조건절에서 조인조건 명시, WHERE 절에서는 일반 조건 명시

```
SELECT 테이블1.컬럼명, 테이블2.컬럼명
FROM 테이블1 INNER JOIN 테이블2
ON 테이블1.조인컬럼 = 테이블2.조인컬럼
```

- ORACLE 표준은 FROM 절에 테이블을 컴마로 구분, WHERE 절에 조인 조건 나열
- ORACLE은 INNER JOIN이 기본 조인 연산 이므로 별도 문법 존재 안함

USING 조건절
- 조인할 컬럼명이 같을 경우 사용
- Alias 나 테이블 이름 같은 접두사 붙이기 불가
- 괄호 필수

```
SELECT 테이블1.컬럼명, 테이블2.컬럼명
FROM 테이블1 INNER JOIN 테이블2
USING (동일컬럼명)
```

NATURAL JOIN
- 두 테이블 간의 동일한 이름을 가지는 모든 컬럼에 대해 EQUI JOIN을 수행
- USING, ON, WHERE 절에서 조건 정의 불가(에러 발생)
- JOIN에 사용된 컬럼들은 데이터 유형이 동일해야 하며 접두사를 사용 불가

```
SELECT 테이블1.컬럼명, 테이블2.컬럼명
FROM 테이블1 NATURAL JOIN 테이블2
```

주의점
- NATURAL JOIN은 동일한 이름의 모든 컬럼을 조인 컬럼으로 사용하므로 조인 컬럼의 값이 모두 같을 때만 결과가 리턴됨

CROSS JOIN
- 테이블 간 JOIN 조건이 없는 경우 생성 가능한 모든 데이터들의 조합
(Cartesian Product(카티시안 곱)) 출력
- 양쪽 테이블 행의 수의 곱한 수의 데이터 조합 발생

```
SELECT 테이블1.컬럼명, 테이블2.컬럼명
FROM 테이블1 CROSS JOIN 테이블2
```

OUTER JOIN
- INNER JOIN과 대비되는 조인 방식
- JOIN 조건에서 동일한 값이 없는 행도 반환할 때 사용
- 테이블 기준 방향에 따라 LEFT OUTER JOIN, RIGHT OUTER JOIN, FULL OUTER JOIN으로 구분

OUTER JOIN 종류
1) LEFT OUTER JOIN
- 왼쪽 테이블이 기준이 되며, 오른쪽 데이터를 채우는 방식
- 우측 값에서 같은 값이 없는 경우 NULL 값으로 출력

ORACLE 표준

```
SELECT *
FROM STUDENT S, PROFESSOR P
WHERE S.PROFNO = P.PROFNO(+)
AND S.GRADE IN (1, 4);
```

ANSI 표준

```
SELECT *
FROM STUDENT S LEFT OUTER JOIN PROFESSOR P
ON S.PROFNO = P.PROFNO
AND S.GRADE IN (1, 4);
```

2) RIGHT OUTER JOIN
- 오른쪽 테이블이 기준이 되며, 왼쪽 데이터를 채우는 방식
- FROM 절에 테이블 순서를 변경하며 LEFT OUTER JOIN으로 수행 가능

3) FULL OUTER JOIN
- 두 테이블 전체 기준으로 결과를 생성하여 중복 데이터는 삭제 후 리턴
- LEFT OUTER JOIN 과 RIGHT OUTER JOIN 결과의 UNION 연산 리턴과 동일함

ORACLE 문법

```
SELECT *
FROM STUDENT S, PROFESSOR P
WHERE S.PROFNO = P.PROFNO(+)
AND S.GRADE IN (1, 4);
UNION
SELECT *
FROM STUDENT S, PROFESSOR P
WHERE S.PROFNO(+) = P.PROFNO
AND S.GRADE IN (1, 4);
```