Chapter 10. Advanced Select

Advanced Select

🎯 6. Type of Triangle

- CASE WHEN 문 사용 하기

```sql
SELECT CASE WHEN ((A = B) and (B = C) and (A = C)) THEN 'Equilateral'
    WHEN ((A+B <= C) or (A+C <= B) or (C+B <= A)) THEN 'Not A Triangle'
    WHEN ((A = B) or (B = C) or (A = C)) THEN 'Isosceles'
    ELSE 'Scalene' END AS Output
FROM TRIANGLES
```

🎯 7. The PADS

- 문자형 함수 CONCAT, SUBSTR 사용하기

```sql
SELECT concat(NAME,'(', upper(substr(OCCUPATION,1,1)),')')
FROM OCCUPATIONS
ORDER BY NAME;

SELECT concat('There are a total of ', count(OCCUPATION), ' ', lower(OCCUPATION),'s.')
FROM OCCUPATIONS
GROUP BY OCCUPATION
ORDER BY count(OCCUPATION), OCCUPATION;
```

🎯 8. Occupations

- RANK 함수 이용하여 그룹핑 할 컬럼 생성
- CASE WHEN문을 사용하여 SQL의 피봇하는 법 학습

```sql
SELECT min(case when A.Occupation = 'Doctor' then A.Name end) as Doctor
     , min(case when A.Occupation = 'Professor' then A.Name end) as Professor
     , min(case when A.Occupation = 'Singer' then A.Name end) as Singer
     , min(case when A.Occupation = 'Actor' then A.Name end )as Actor

FROM (SELECT Name
           , Occupation
           , rank() over(partition by Occupation order by Name asc) as name_order
      FROM OCCUPATIONS) A

GROUP BY A.name_order
```

🎯 9. Binary Tree Nodes

- IN 조건에 서브쿼리 사용
- CASE WHEN 문 사용

```sql
SELECT case when P is null then concat(N, " Root")
            when N in (select distinct P from BST) then concat(N, " Inner")
            else concat(N, " Leaf")
       end
FROM BST
ORDER BY N ASC
```

🎯 10. New Companies

- JOIN 없이 쿼리문 작성
- 메인 쿼리와 조건을 따지는 스칼라 서브쿼리 사용

```
SELECT A.company_code
     , A.founder
     , (select count(distinct lead_manager_code)
        from Lead_Manager
        where company_code = A.company_code)
     , (select count(distinct senior_manager_code)
        from Senior_Manager
        where company_code = A.company_code)
     , (select count(distinct manager_code)
        from Manager
        where company_code = A.company_code)
     , (select count(distinct employee_code)
        from Employee
        where company_code = A.company_code)
FROM Company A
ORDER BY company_code
```