

Chapter 10. Basic Join

Basic Join

🎯 11. Average Population of Each Continent

- JOIN을 위해 매칭되는 컬럼 찾기
- 집계할 컬럼 찾기

```
1 • SELECT COUNTRY.Continent
2       , floor(avg(CITY.Population))
3 FROM COUNTRY
4 INNER JOIN CITY
5
6 ON COUNTRY.Code = CITY.CountryCode
7
8 GROUP BY COUNTRY.Continent
```

🎯 12. The Report

- JOIN을 통해 점수마다 정확한 등급을 매핑시키기
- 우선순위에 따라 ORDER BY 절 사용

```
1 • SELECT case when B.Grade < 8 then NULL
2           else A.Name
3           end as Name
4           , B.Grade
5           , A.Marks
6 FROM Students A
7 INNER JOIN Grades B
8 ON A.Marks >= B.Min_Mark
9 AND A.Marks <= B.Max_Mark
10
11 ORDER BY Grade DESC, Name ASC, Marks ASC
```

🎯 13. Contest Leaderboard

- Challenge마다 Submission의 MAX Score 구하기
- JOIN과 GROUP BY를 사용하여 Total Score 구하기
- HAVING 조건 사용하기

```
1 • SELECT A.hacker_id
2       , A.name
3       , sum(score) as total_score
4 FROM Hackers A
5 INNER JOIN (SELECT hacker_id
6                , challenge_id
7                , max(score) as score
8              FROM Submissions
9              GROUP BY hacker_id, challenge_id) B
10
11 ON A.hacker_id = B.hacker_id
12 GROUP BY A.hacker_id, A.name
13 HAVING sum(score) != 0
14 ORDER BY total_score desc, hacker_id asc
```

🎯 14. Challenges

- Hacker ID 별로 제출한 Challenge 수 계산하기
- WHERE 절에 서브쿼리 사용하여 조건에 부합한 Challenge 수로 필터하기
- 서브쿼리시 Table alias 사용하기

```
1 • SELECT main.*
2 FROM
3   (select a.hacker_id
4      , a.name
5      , count(b.challenge_id) as num
6
7   from hackers a
8   inner join challenges b
9   on a.hacker_id = b.hacker_id
10
11  group by a.hacker_id, a.name) main
12
13 WHERE main.num = (select max(t1.cnt_challenges)
14                  from (select hacker_id
15                        , count(challenge_id) as cnt_challenges
16                        from challenges
17                        group by hacker_id) t1)
18 OR main.num in (select t2.cnt_challenges
19                from (select hacker_id
20                      , count(challenge_id) as cnt_challenges
21                      from challenges
22                      group by hacker_id) t2
23                group by t2.cnt_challenges
24                having count(t2.hacker_id) < 2)
25 ORDER BY main.num desc, main.hacker_id
```