Week-4: Assignment

Q1). **Weather Observation Station 8**

Ans). SELECT DISTINCT CITY FROM STATION WHERE CITY REGEXP '^[aeiouAEIOU].\*[aeiouAEIOU]$'

Q2). **Population Density Difernece**

Ans). SELECT MAX(POPULATION)-MIN(POPULATION)   
FROM CITY;

Q3). **Weather Observation Station 19**

Ans). SELECT ROUND(  
 SQRT(POWER(MAX(LAT\_N) - MIN(LAT\_N), 2) +   
 POWER(MAX(LONG\_W) - MIN(LONG\_W), 2)), 4)  
FROM STATION;

Q4). **Weather Observation Station 20**

Ans). WITH Ordered AS (  
 SELECT LAT\_N,  
 ROW\_NUMBER() OVER (ORDER BY LAT\_N) AS rn,  
 COUNT(\*) OVER () AS total\_rows  
 FROM STATION  
),  
MedianCalc AS (  
 SELECT LAT\_N  
 FROM Ordered  
 WHERE rn = (total\_rows + 1) / 2  
 OR rn = (total\_rows + 2) / 2  
)  
SELECT ROUND(AVG(LAT\_N), 4) AS median  
FROM MedianCalc;

Q5). **African Cities**

Ans). SELECT C1.name  
FROM CITY as C1  
JOIN COUNTRY as C2  
ON C1.COUNTRYCODE = C2.Code  
WHERE C2.CONTINENT='Africa';

Q6). **Report: Write a SQL Query to create a report**

Ans). SELECT  
 CASE   
 WHEN G.Grade < 8 THEN 'NULL'  
 ELSE S.Name  
 END AS Name,  
 G.Grade,  
 S.Marks  
FROM Students S  
JOIN Grades G  
 ON S.Marks BETWEEN G.Min\_Mark AND G.Max\_Mark  
ORDER BY  
 G.Grade DESC,  
 CASE   
 WHEN G.Grade >= 8 THEN S.Name  
 ELSE NULL  
 END,  
 CASE   
 WHEN G.Grade < 8 THEN S.Marks  
 ELSE NULL  
 END;

Q7). **Top Competitors**

Ans). SELECT  
 s.hacker\_id,  
 h.name  
FROM Submissions s  
JOIN Challenges c ON s.challenge\_id = c.challenge\_id  
JOIN Difficulty d ON c.difficulty\_level = d.difficulty\_level  
JOIN Hackers h ON s.hacker\_id = h.hacker\_id  
WHERE s.score = d.score  
GROUP BY s.hacker\_id, h.name  
HAVING COUNT(DISTINCT s.challenge\_id) > 1  
ORDER BY COUNT(DISTINCT s.challenge\_id) DESC, s.hacker\_id;

Q8). **Ollivander's Inventory**

Ans). SELECT w.id, wp.age, w.coins\_needed, w.power  
FROM Wands w  
JOIN Wands\_Property wp ON w.code = wp.code  
JOIN (  
 SELECT w2.power, wp2.age, MIN(w2.coins\_needed) AS min\_coins  
 FROM Wands w2  
 JOIN Wands\_Property wp2 ON w2.code = wp2.code  
 WHERE wp2.is\_evil = 0  
 GROUP BY w2.power, wp2.age  
) AS best  
 ON w.power = best.power   
 AND wp.age = best.age   
 AND w.coins\_needed = best.min\_coins  
WHERE wp.is\_evil = 0  
ORDER BY w.power DESC, wp.age DESC;

Q9). **Contest Leaderboard: Question based on joins**

Ans). SELECT  
 h.hacker\_id,  
 h.name,  
 SUM(max\_scores.max\_score) AS total\_score  
FROM Hackers h  
JOIN (  
 SELECT  
 s.hacker\_id,  
 s.challenge\_id,  
 MAX(s.score) AS max\_score  
 FROM Submissions s  
 GROUP BY s.hacker\_id, s.challenge\_id  
) AS max\_scores  
 ON h.hacker\_id = max\_scores.hacker\_id  
GROUP BY h.hacker\_id, h.name  
HAVING SUM(max\_scores.max\_score) > 0  
ORDER BY total\_score DESC, h.hacker\_id ASC;