Task-1

Solution

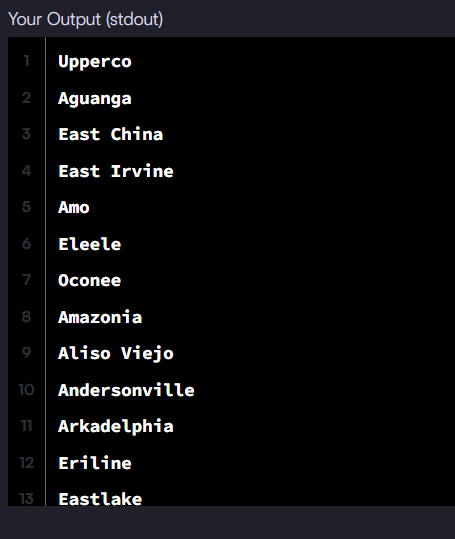
SELECT DISTINCT CITY

FROM STATION

WHERE LOWER(SUBSTRING(CITY, 1, 1)) IN ('a', 'e', 'i', 'o', 'u')

AND LOWER(SUBSTRING(CITY, LENGTH(CITY), 1)) IN ('a', 'e', 'i', 'o', 'u');

Output:



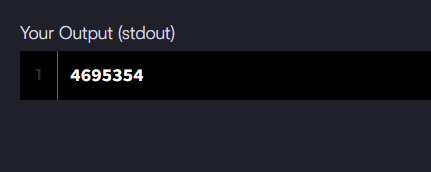
Task-2

Solution:

SELECT MAX(POPULATION) - MIN(POPULATION) AS PopulationDifference

FROM CITY;

Output:



Task-3

Solution:

SELECT

ROUND(

SQRT(

POWER(MAX(LAT\_N) - MIN(LAT\_N), 2) +

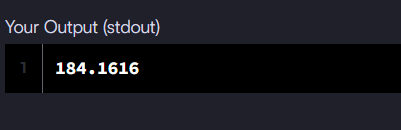
POWER(MAX(LONG\_W) - MIN(LONG\_W), 2)

), 4

) AS EuclideanDistance

FROM STATION;

Output:



Task-4

Soution:

SELECT

ROUND(

AVG(LAT\_N), 4

) AS Median\_LAT\_N

FROM (

SELECT

LAT\_N,

ROW\_NUMBER() OVER (ORDER BY LAT\_N) AS RowAsc,

ROW\_NUMBER() OVER (ORDER BY LAT\_N DESC) AS RowDesc

FROM STATION

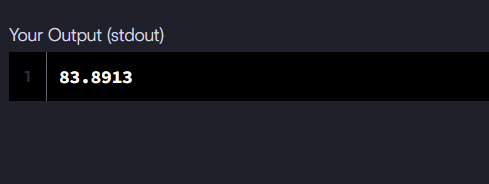
) AS sub

WHERE RowAsc = RowDesc

OR RowAsc + 1 = RowDesc

OR RowAsc = RowDesc + 1;

Output:



Task-5

Solution:

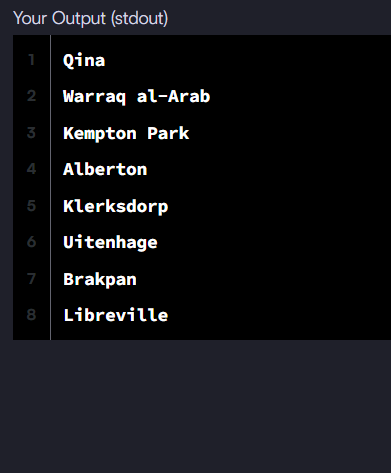
SELECT CITY.NAME

FROM CITY

JOIN COUNTRY ON CITY.CountryCode = COUNTRY.Code

WHERE COUNTRY.Continent = 'Africa';

Output:



Task-6 is same as task -5

Task-7

Solution:

SELECT

CASE

WHEN G.Grade >= 8 THEN S.Name

ELSE NULL

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 ASC,

CASE

WHEN G.Grade < 8 THEN S.Marks

ELSE NULL

END ASC;

Output :



Task-8

Solution:

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 ASC;

Output:



Task-9:

Solution:

SELECT

w.id,

wp.age,

w.coins\_needed,

w.power

FROM

Wands w

JOIN

Wands\_Property wp

ON w.code = wp.code

WHERE

wp.is\_evil = 0

AND w.coins\_needed = (

SELECT MIN(w2.coins\_needed)

FROM Wands w2

JOIN Wands\_Property wp2 ON w2.code = wp2.code

WHERE wp2.is\_evil = 0

AND wp2.age = wp.age

AND w2.power = w.power

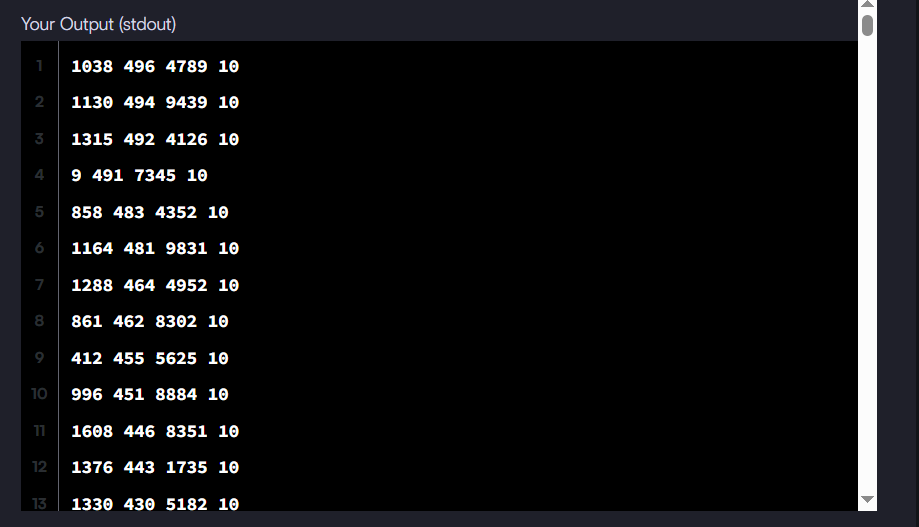
)

ORDER BY

w.power DESC,

wp.age DESC;

Output:



Task-10:

Solution:

SELECT

h.hacker\_id,

h.name,

SUM(max\_scores.max\_score) AS total\_score

FROM

Hackers h

JOIN (

SELECT

hacker\_id,

challenge\_id,

MAX(score) AS max\_score

FROM

Submissions

GROUP BY

hacker\_id, challenge\_id

) AS max\_scores

ON h.hacker\_id = max\_scores.hacker\_id

GROUP BY

h.hacker\_id, h.name

HAVING

total\_score > 0

ORDER BY

total\_score DESC,

h.hacker\_id ASC;

Output:

