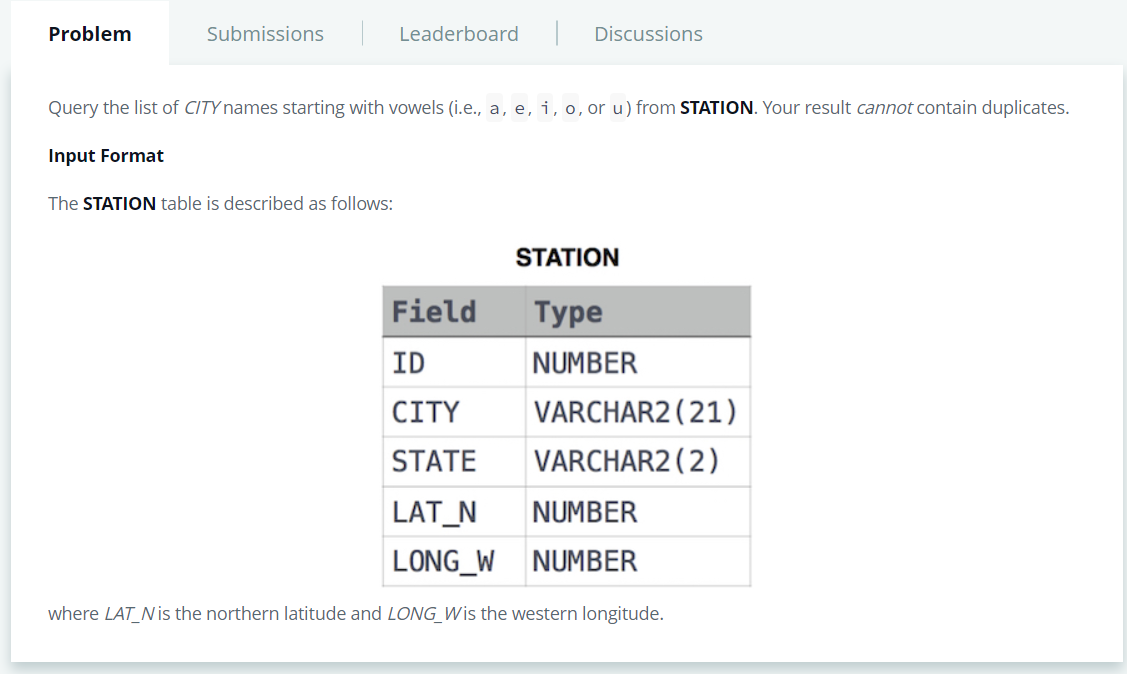
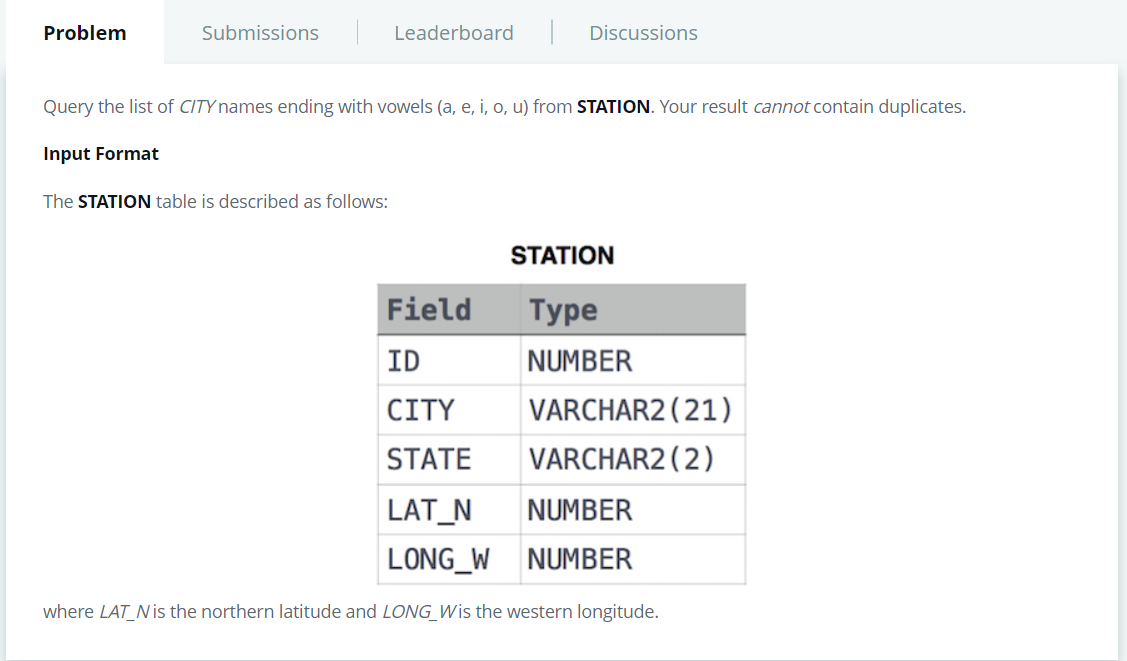
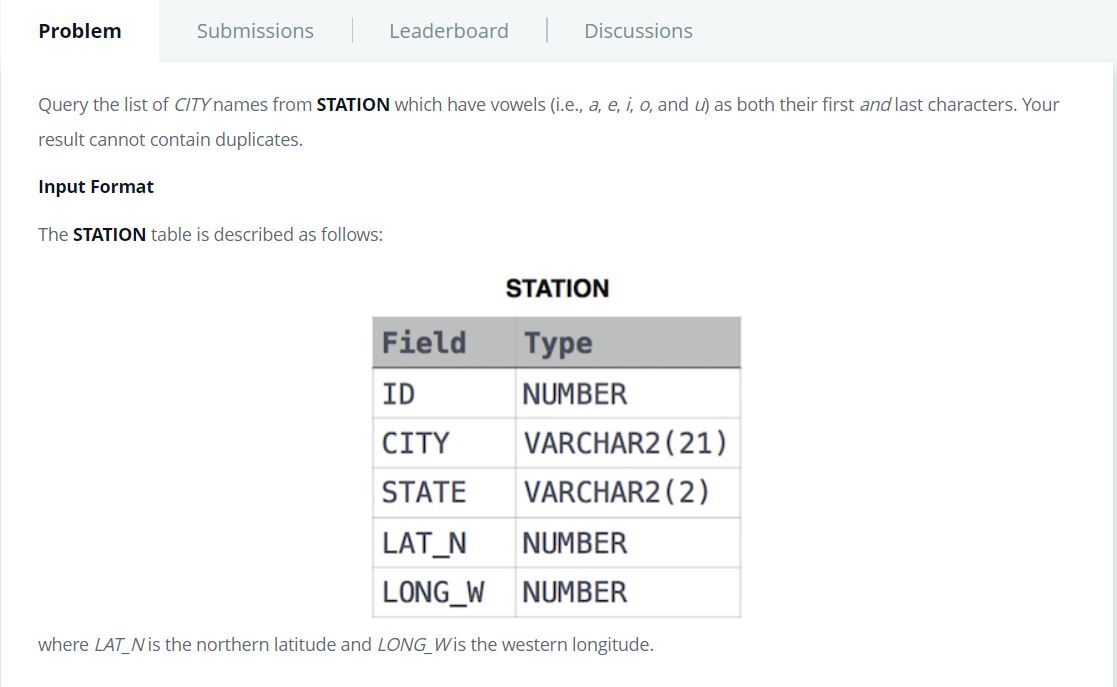
1.



SELECT DISTINCT CITY FROM STATION WHERE CITY LIKE '[aeiou]%';

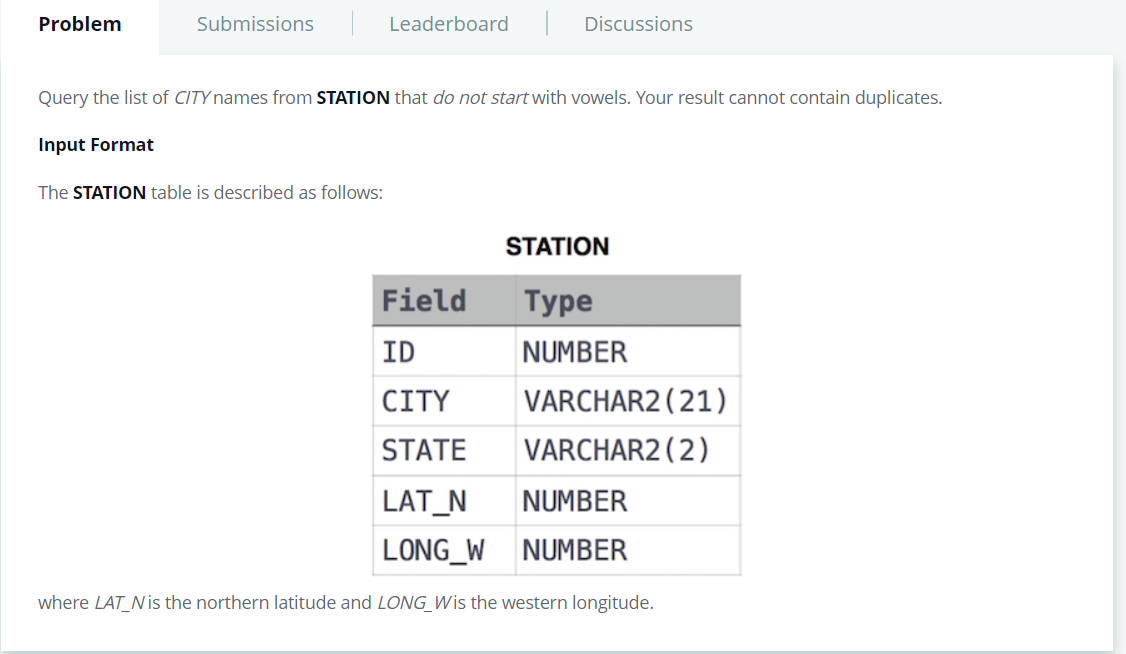
2. 

SELECT DISTINCT CITY FROM STATION WHERE CITY LIKE '%[aeiou]'

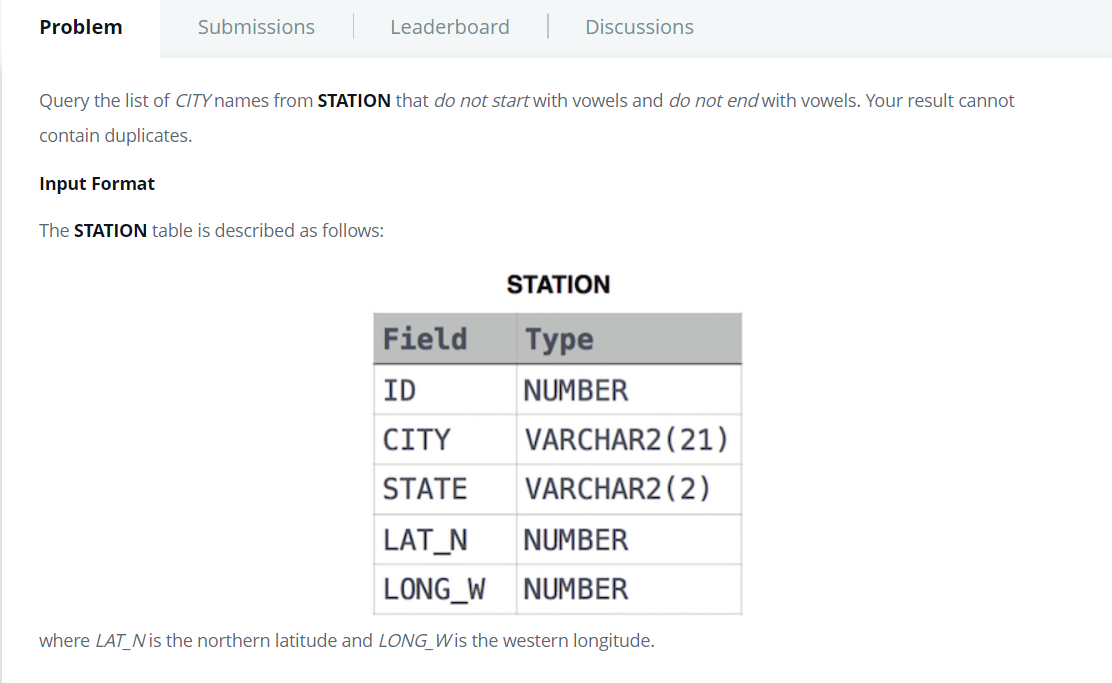
3. 

SELECT DISTINCT CITY FROM STATION WHERE CITY LIKE '[aeiou]%[aeiou]'

4.



SELECT DISTINCT CITY FROM STATION WHERE CITY NOT LIKE '[aeiou]%'

5. 

SELECT DISTINCT CITY

FROM STATION

WHERE

CITY NOT LIKE '%[aeiou]' AND

CITY NOT LIKE'[aeiou]%'

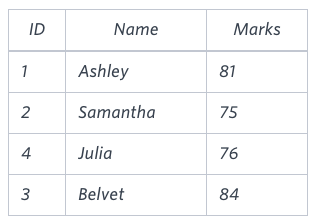
6.

Query the *Name* of any student in **STUDENTS** who scored higher than  *Marks*. Order your output by the *last three characters*of each name. If two or more students both have names ending in the same last three characters (i.e.: Bobby, Robby, etc.), secondary sort them by ascending *ID*.

**Input Format**

The **STUDENTS** table is described as follows:The *Name* column only contains uppercase (A-Z) and lowercase (a-z) letters.

**Sample Input**



**Sample Output**

Ashley

Julia

Belvet

**Explanation**

Only Ashley, Julia, and Belvet have *Marks* > . If you look at the last three characters of each of their names, there are no duplicates and 'ley' < 'lia' < 'vet'.

Ans:-

SELECT Name FROM STUDENTS

WHERE Marks > 75

ORDER BY RIGHT (Name,3) ASC,ID ASC

7.

Write a query that prints a list of employee names (i.e.: the name attribute) for employees in **Employee** having a salary greater than  per month who have been employees for less than  months. Sort your result by ascending employee\_id.

**Input Format**

The **Employee** table containing employee data for a company is described as follows:



where employee\_id is an employee's ID number, name is their name, months is the total number of months they've been working for the company, and salary is the their monthly salary.

**Sample Input**



**Sample Output**

Angela

Michael

Todd

Joe

**Explanation**

Angela has been an employee for  month and earns  per month.

Michael has been an employee for  months and earns  per month.

Todd has been an employee for  months and earns  per month.

Joe has been an employee for  months and earns  per month.

We order our output by ascending employee\_id.

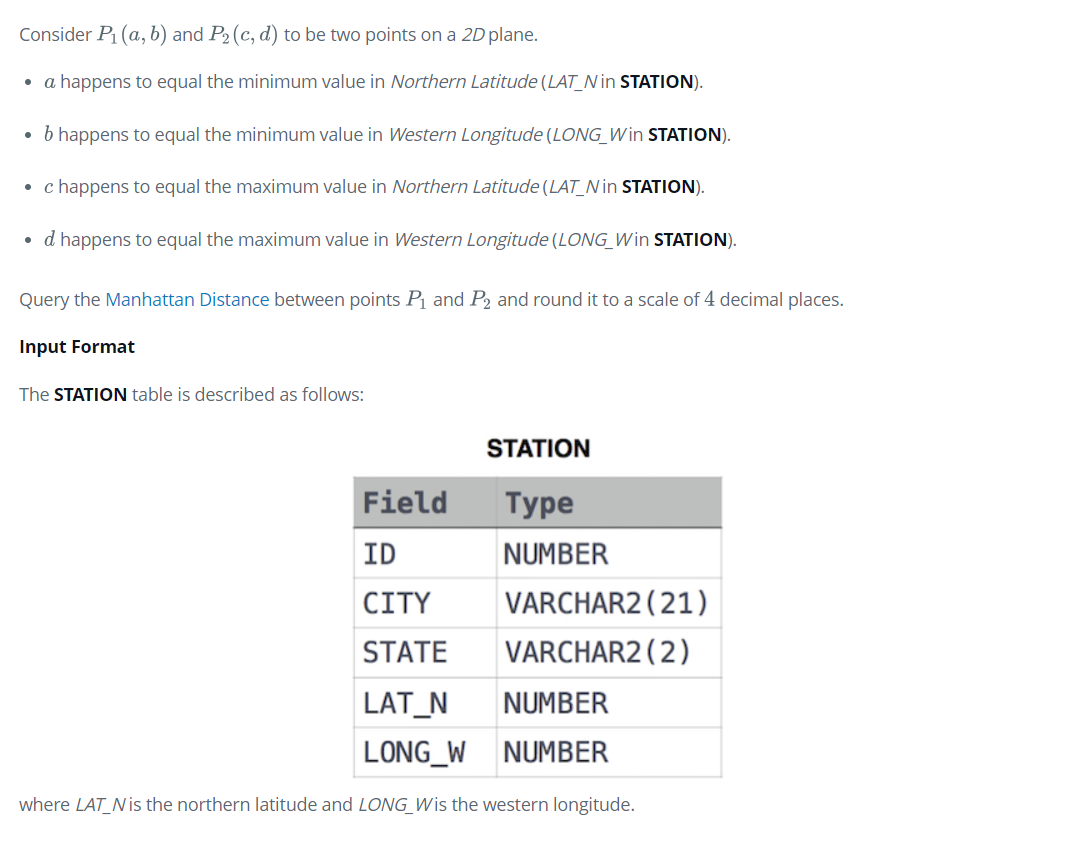
Ans:-

select name from Employee

where salary > 2000 and

months < 10

order by employee\_id ASC

8. 

Ans

select round( abs(MIN(LAT\_N) - MAX(LAT\_N) ) + abs( MIN(LONG\_W) - MAX(LONG\_W)) ,4) from STATION