1. Query all columns for all American cities in the ****CITY**** table with populations larger than 100000. The ****CountryCode**** for America is USA.The ****CITY**** table is described as



**SELECT \***

**FROM CITY**

**WHERE CountryCode = 'USA'**

**AND Population > 100000;**

1. Query the ****NAME**** field for all American cities in the ****CITY**** table with populations larger than 120000. The CountryCode for America is USA.

The ****CITY**** table is described as follows:  


**Select name from city where population >120000 and CountryCode=’USA’;**

1. Query all columns for a city in ****CITY**** with the ID 1661.The ****CITY**** table is described as follows:  
   

**Select \* from city where id=1661;**

1. Query all attributes of every Japanese city in the ****CITY**** table. The ****COUNTRYCODE**** for Japan is JPN.The ****CITY**** table is described as follows:  
   

**Select \* from city where COUNTRYCODE=’JPN’;**

5.Query the names of all the Japanese cities in the ****CITY**** table. The ****COUNTRYCODE**** for Japan is JPN.  
The ****CITY**** table is described as follows



**Select name from city where COUNTRYCODE=’JPN’;**

6.Query a list of ****CITY**** and ****STATE**** from the ****STATION**** table.  
The ****STATION**** table is described as follows:  


where ****LAT\_N**** is the northern latitude and ****LONG\_W**** is the western longitude.

**Select city,state from station;**

7.Query the following two values from the ****STATION**** table:

1. The sum of all values in LAT\_N rounded to a scale of  decimal places.
2. The sum of all values in LONG\_W rounded to a scale of  decimal places.

****Input Format****

The ****STATION**** table is described as follows:



where LAT\_N is the northern latitude and LONG\_W is the western longitude.

****Output Format****

Your results must be in the form:

Lat lon

where  is the sum of all values in LAT\_N and  is the sum of all values in LONG\_W. Both results must be rounded to a scale of  decimal places.

**Select round(sum(LAT\_N),2),round(sum(LONG\_W),2) from station;**

8.Query a list of ****CITY**** names from ****STATION**** for cities that have an even ****ID**** number. Print the results in any order, but exclude duplicates from the answer.  
The ****STATION**** table is described as follows:



where ****LAT\_N**** is the northern latitude and ****LONG\_W**** is the western longitude.

**Select distinct city from station where mod(id,2)=0;**

9.Find the difference between the total number of ****CITY**** entries in the table and the number of distinct ****CITY**** entries in the table.  
The ****STATION**** table is described as follows:



where ****LAT\_N**** is the northern latitude and ****LONG\_W**** is the western longitude.

For example, if there are three records in the table with ****CITY**** values 'New York', 'New York', 'Bengalaru', there are 2 different city names: 'New York' and 'Bengalaru'. The query returns , because total number of records-number of unique city names =3-2=1

**Select count(city)-count(distinct city) from station;**

10.Query the two cities in ****STATION**** with the shortest and longest CITY names, as well as their respective lengths (i.e.: number of characters in the name). If there is more than one smallest or largest city, choose the one that comes first when ordered alphabetically.  
The ****STATION**** table is described as follows:



where ****LAT\_N**** is the northern latitude and ****LONG\_W**** is the western longitude.

****Sample Input****

For example, ****CITY**** has four entries: ****DEF, ABC, PQRS**** and ****WXY****.

****Sample Output****

****ABC 3****

****PQRS 4****

****Explanation****

When ordered alphabetically, the ****CITY**** names are listed as ****ABC, DEF, PQRS,**** and ****WXY****, with lengths  and . The longest name is ****PQRS****, but there are  options for shortest named city. Choose ****ABC****, because it comes first alphabetically.

****Note****  
You can write two separate queries to get the desired output. It need not be a single query.

**SELECT city, LENGTH(city)**

**FROM station ORDER BY LENGTH(city), city LIMIT 1;**

**SELECT city, LENGTH(city) FROM station ORDER BY LENGTH(city) DESC, city LIMIT 1;**

1. Query the list of CITY names starting with vowels (i.e., a, e, i, o, or u)

from ****STATION****. Your result cannot contain duplicates.

****Input Format****

The ****STATION**** table is described as follows:



where LAT\_N is the northern latitude and LONG\_W is the western longitude.

**SELECT DISTINCT city FROM station WHERE( city LIKE 'A%' OR city**

**LIKE 'E%' OR city LIKE 'I%' OR city LIKE 'O%' OR city LIKE 'U%');**

1. Query the list of CITY names ending with vowels (a, e, i, o, u) from ****STATION****. Your result cannot contain duplicates.

****Input Format**** The ****STATION**** table is described as follows:



where LAT\_N is the northern latitude and LONG\_W is the western longitude.

**SELECT DISTINCT city FROM station WHERE( city LIKE 'A%' OR city**

**LIKE 'E%' OR city LIKE 'I%' OR city LIKE 'O%' OR city LIKE 'U%');**

**13.Query the** list of CITY names from ****STATION**** which have vowels (i.e., a, e, i, o, and u) as both their first and last characters. Your result cannot contain duplicates.

****Input Format****

The ****STATION**** table is described as follows:



where LAT\_N is the northern latitude and LONG\_W is the western longitude.

**SELECT DISTINCT city**

**FROM station**

**WHERE (city LIKE 'A%' OR city LIKE 'E%' OR city LIKE 'I%' OR city LIKE 'O%' OR city LIKE 'U%')**

**AND (city LIKE '%A' OR city LIKE '%E' OR city LIKE '%I' OR city LIKE '%O' OR city LIKE '%U');**

14.Query the list of CITY names from ****STATION**** that do not start with vowels. Your result cannot contain duplicates.

****Input Format****

The ****STATION**** table is described as follows:



where LAT\_N is the northern latitude and LONG\_W is the western longitude.

**SELECT DISTINCT city FROM station WHERE city NOT LIKE '%A'AND city NOT LIKE '%E' AND city NOT LIKE '%I' AND city NOT LIKE '%O' AND city NOT LIKE '%U';**

15.Query the list of CITY names from ****STATION**** that do not end with vowels. Your result cannot contain duplicates.

****Input Format****

The ****STATION**** table is described as follows:



where LAT\_N is the northern latitude and LONG\_W is the western longitude.

**SELECT DISTINCT city FROM station WHERE city NOT LIKE '%A' AND city NOT LIKE '%E' AND city NOT LIKE '%I' AND city NOT LIKE '%O' AND city NOT LIKE '%U';**