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

select \* from city where population > 100000 and countrycode ='USA';

Query a list of CITY names from **STATION** with even ID numbers only. You may print the results in any order, but must exclude duplicates from your answer.

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

Let  N be the number of CITY entries in **STATION**, and let N’  be the number of distinct CITY names in **STATION**; query the value of  from **STATION**. In other words, find the difference between the total number of CITYentries in the table and the number of distinct CITY entries in the table.

select (count(city) - count(distinct(city))) as difference from station ;

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.

select \* from ( select city , length(city) from station order by length(city), city ) where rownum = 1 union

select \* from ( select city , length(city) from station order by length(city) desc, city) where rownum = 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.

select distinct(city)from station where regexp\_like(city,'^[AaEeIiOoUu].');

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

select distinct(city) from station where regexp\_like (lower(city),'[aeiou]$');

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.

SELECT DISTINCT CITY FROM STATION WHERE LOWER(SUBSTR(CITY,LENGTH(CITY),1)) IN ('a','e','i','o','u') and LOWER(SUBSTR(CITY,0,1)) IN ('a','e','i','o','u');

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

select distinct(city) from station where substr(lower(city),0,1) not in ('a','e','i','o','u');

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

select distinct(city) from station where substr(lower(city),length(city),1) not in ('a','e','i','o','u') or substr(lower(city),0,1) not in ('a','e','i','o','u');

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

select distinct(city) from station where substr(lower(city),0,1) not in ('a','e','i','o','u') and substr(lower(city),length(city),1) not in ('a','e','i','o','u');

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.

SELECT (NAME) FROM STUDENTS WHERE MARKS>75 ORDER BY SUBSTR(NAME,-3,3), ID ASC;

Write a query that prints a list of employee names (i.e.: the name attribute) from the **Employee** table in alphabetical order. 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 their monthly salary.

select name from employee order by name;

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.

select name from employee where salary > 2000 and months < 10 order by employee\_id;