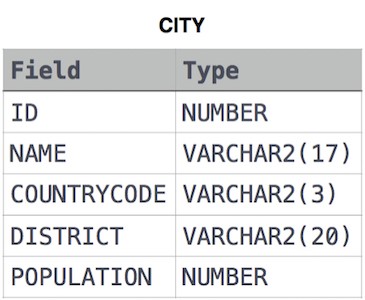
**Revising the Select Query I**

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



**Input Format**

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



Answer:

Select \* from CITY where

COUNTRYCODE='USA' and POPULATION > 100000;

2. Query the names of all American cities in **CITY** with populations larger than 120000. The CountryCode for America is USA.

A2.: SELECT NAME FROM CITY WHERE

COUNTRYCODE='USA' AND POPULATION>120000;

3.Query all columns (attributes) for every row in the **CITY** table.

A3.: SELECT \* FROM CITY;

4. Query all columns for a city in **CITY** with the ID 1661.

A4.: SELECT \* FROM CITY WHERE ID=1661;

5. Query all attributes of every Japanese city in the **CITY** table. The COUNTRYCODE for Japan is JPN.

A5.: SELECT \* FROM CITY WHERE COUNTRYCODE='JPN';

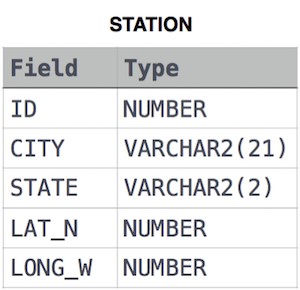
**Weather Observation Station 1**

Query a list of CITY and STATE from the **STATION** table.

**Input Format**

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





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

Answer: SELECT CITY,STATE FROM STATION;

2. 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.

A2: SELECT DISTINCT CITY FROM STATION WHERE

ID % 2 = 0;

3. 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.

A3: select CITY,LENGTH(CITY) from STATION order by Length(CITY) asc, CITY limit 1;

select CITY,LENGTH(CITY) from STATION order by Length(CITY) desc, CITY limit 1;

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

A4: Select distinct CITY from STATION WHERE

CITY like '[A,E,I,O,U]%';

**Higher Than 75 Marks**

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.

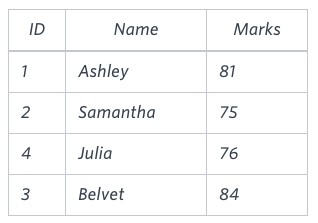
**Input Format**

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



The Name column only contains uppercase (A- Z) and lowercase (a- z) letters.

**Sample Input**



Answer: SELECT NAME FROM STUDENTS WHERE

MARKS>75

ORDER BY RIGHT(NAME,3), ID ASC;

**THE SECRET QUESTION:**

SELECT IF(A+B>C AND A+C>B AND B+C>A, IF(A=B AND B=C, 'Equilateral', IF(A=B OR B=C OR A=C, 'Isosceles', 'Scalene')), 'Not A Triangle') FROM TRIANGLES;