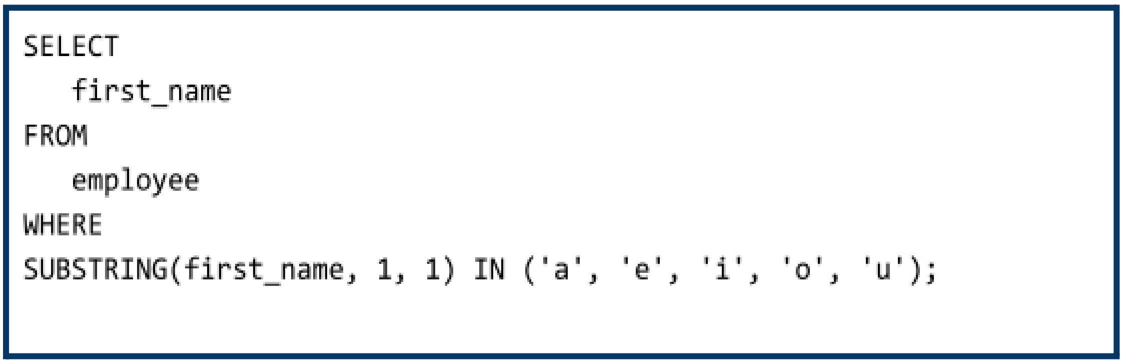
LEFT(),RIGHT(),LPAD(),RPAD(),SOUNDEX() FUNCTION

1. **Query the first name of all employees whose first name begins with vowel**



```
SELECT
   first_name
FROM
   employee
WHERE
SUBSTRING(first_name, 1, 1) IN ('a', 'e', 'i', 'o', 'u');
```

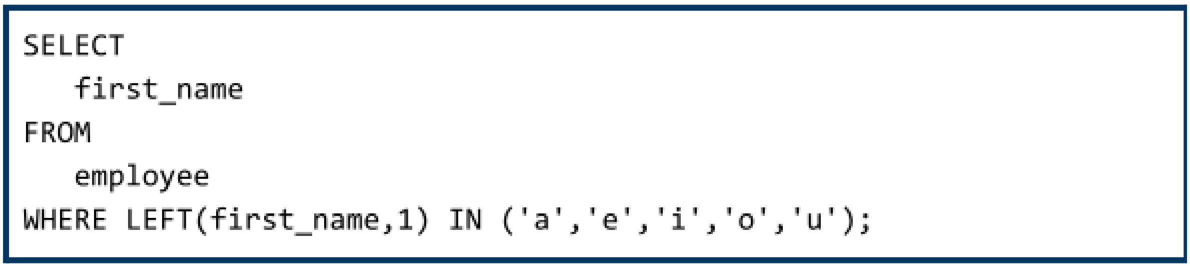
Output:

first_name
andy
anjel

Here we have used substring() to extract the first character from the string, instead of this we can make use of **LEFT()** to extract the first character. The syntax looks as shown below

Syntax:

LEFT(string, number_of_chars)



```
SELECT
   first_name
FROM
   employee
WHERE LEFT(first_name,1) IN ('a','e','i','o','u');
```

1. **Display 'MYSQL' from 'MYSQL IS EASY'**

```
SELECT
LEFT('MYSQL IS EASY',5) as LT;
```

Output:

LT
MYSQL

2. **Query the first name of all the employees whose first name ends with vowel**

```
SELECT
  first_name
FROM
  employee
WHERE SUBSTRING(first_name, -1) IN ('a','e','i','o','u');
```

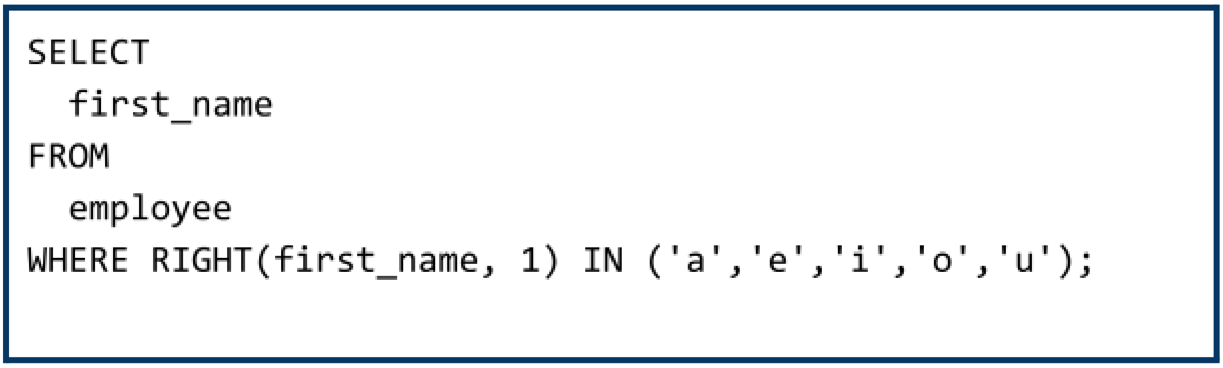
Output:

first_name
mike

Instead of making use of substring() to fetch the last characters in the given string you can make use of RIGHT(). The syntax of RIGHT() function is shown below

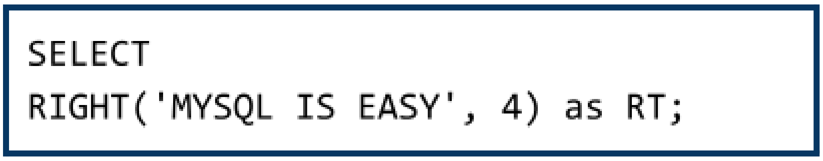
Syntax:

RIGHT(string, number_of_chars)



```
SELECT
  first_name
FROM
  employee
WHERE RIGHT(first_name, 1) IN ('a','e','i','o','u');
```

3. **Display ‘EASY’ from ‘MYSQL IS EASY’**



```
SELECT
RIGHT('MYSQL IS EASY', 4) as RT;
```

Output:

RT
EASY

LPAD() Function:

LPAD() function in MySQL is used to pad or add a string to the left side of the original string.

Syntax:

LPAD(str, len, padstr)

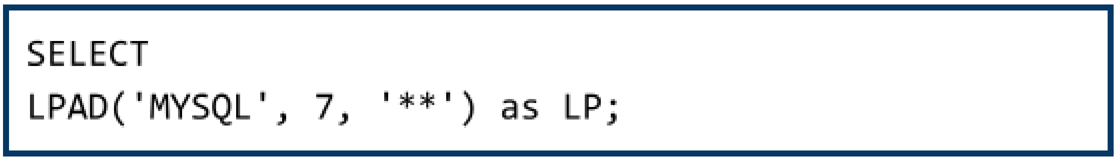
- **str** –
  The actual string which is to be padded. If the length of the original string is larger than the len parameter, this function removes the overfloating characters from the string.

- **len** –
  This is the length of a final string after the left padding.

- **padstr** –
  String that to be added to the left side of the Original Str.

Returns : It returns a new string of length len after padding.

4. **Query the string 'MYSQL' by adding '**' at the beginning of the string**

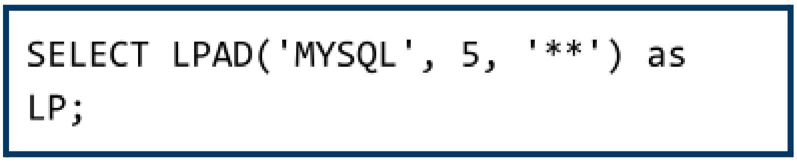


```
SELECT
LPAD('MYSQL', 7, '**') as LP;
```

Output:

LP
**MYSQL

Now let's see if the length is equal to or less than the given string

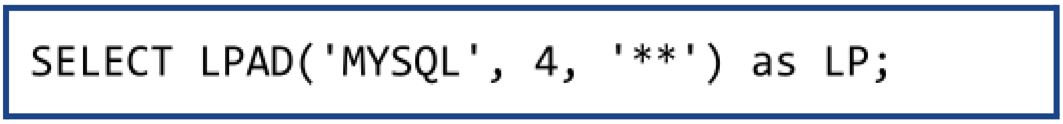


```
SELECT LPAD('MYSQL', 5, '**') as
LP;
```

Output:

LP
MYSQL

As you can see from the above output here length of the string is same given string so there is no padding happens as shown above



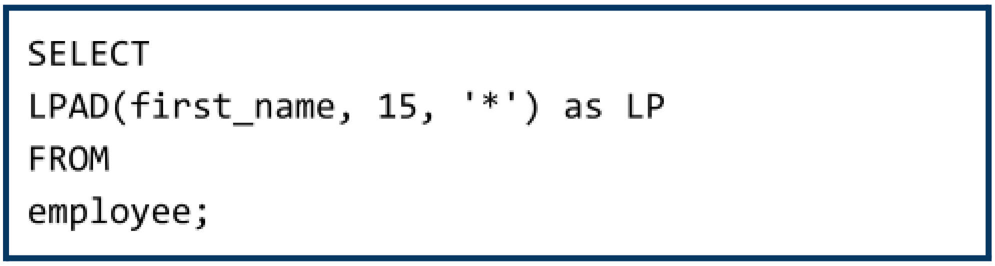
```
SELECT LPAD('MYSQL', 4, '**') as LP;
```

Output:

LP
MYSQ

Here the length of the original string is larger than the len parameter in that case, string length will be reduced.

5. **Query the first name of all the employees by adding '*' at the beginning with 15 as total length**



```
SELECT
LPAD(first_name, 15, '*') as LP
FROM
employee;
```

Output:

LP
**********kelly
************Tom
***********mike
***********andy
**********anjel
************ram
**********rohan
***********john

As you can see from the above output for all first name in the employee table * is added at the beginning till the length of the string becomes 15.

RPAD() Function:

RPAD() function in MySQL is used to pad or add a string to the right side of the original string.

Syntax:
RPAD(str, len, padstr)

- **str** –
The actual string which is to be padded. If the length of the original string is larger than the len parameter, this function removes the

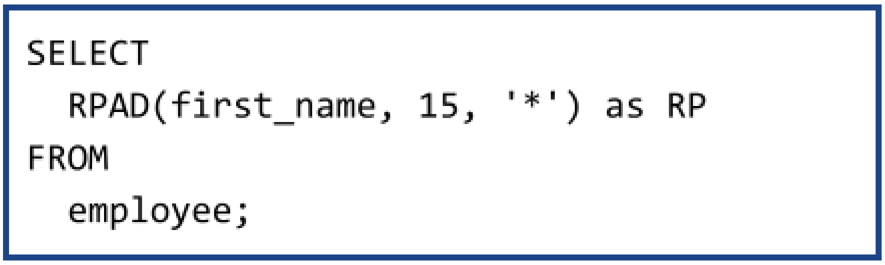
overflowing characters from the string.

- **len** –
  This is the length of a final string after the right padding.

- **padstr** –
  String that to be added to the right side of the Original Str.

Returns : It returns a new string of length len after padding.

6. **Query the first name of all the employees by adding '*' at the ending with 15 as total length**



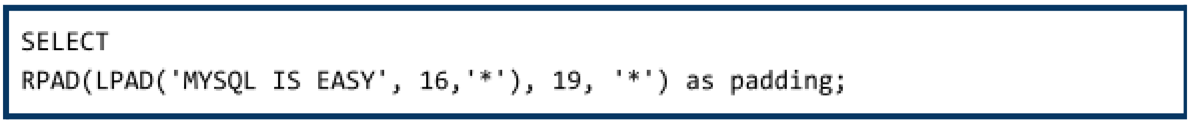
```
SELECT
  RPAD(first_name, 15, '*') as RP
FROM
  employee;
```

Output:

RP
kelly**********
tom************
mike***********
andy***********
anjel**********
ram************

rohan**********
john***********

7. Display the string 'MYSQL IS EASY' as '**MYSQL IS EASY**'



```
SELECT
RPAD(LPAD('MYSQL IS EASY', 16,'*'), 19, '*') as padding;
```

Output:

padding
MYSQL IS EASY

Here left padding is done on the string and making a string of length 16, for that string right padding is done making the string length 19.

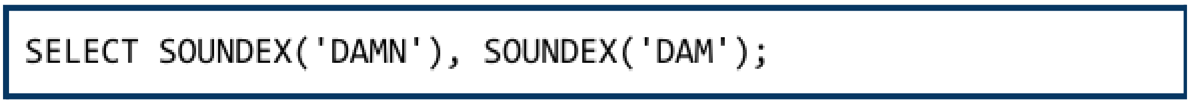
SOUNDEX() Function:

The SOUNDEX() function accepts a string and converts it to a four-character code based on how the string sounds when it is spoken.

The following shows the syntax of the SOUNDEX() function:

SOUNDEX(input_string);

8. Query to check 'DAMN' and 'DAM' sounds same



```
SELECT SOUNDEX('DAMN'), SOUNDEX('DAM');
```

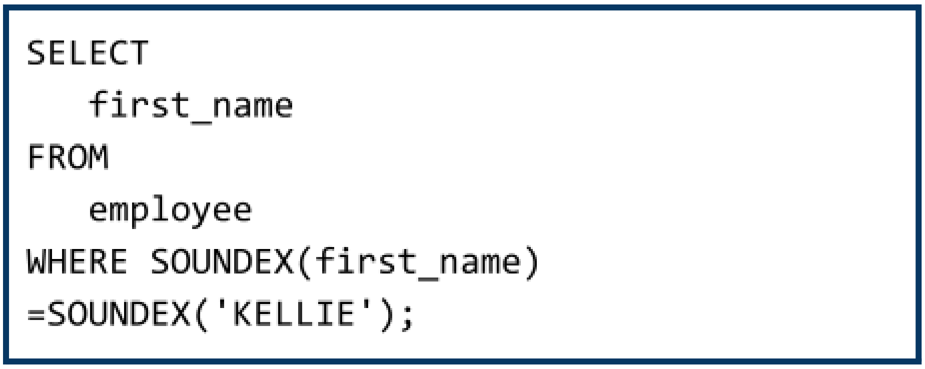
Output:

SOUNDEX('DAMN')	SOUNDEX('DAM')

D500	D500

As you can see from the above output, both string is returning the same value it clearly indicates that both strings sounds the same

9. **Query the first name of all employees that are homophones of ‘KELLIE’**



```
SELECT
   first_name
FROM
   employee
WHERE SOUNDEX(first_name)
=SOUNDEX('KELLIE');
```

Output:

first_name
kelly

As you can observe from the above it will match whether the first name sounds exactly the same as ‘KELLIE’ . Here it matches with one string kelly which sounds exactly same as ‘KELLIE’

HACKERANK QUERIES

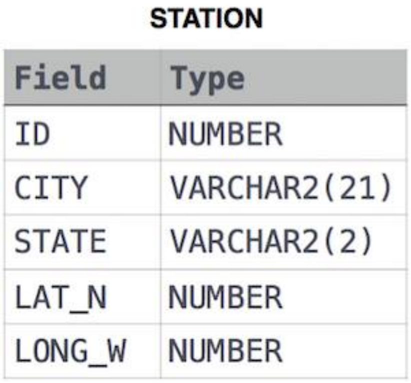
1. **Weather Observation station 12**
   Link:
   https://www.hackerrank.com/challenges/weather-observation-station-12/problem?isFullScreen=true

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.

Input Format

The STATION table is described as follows:

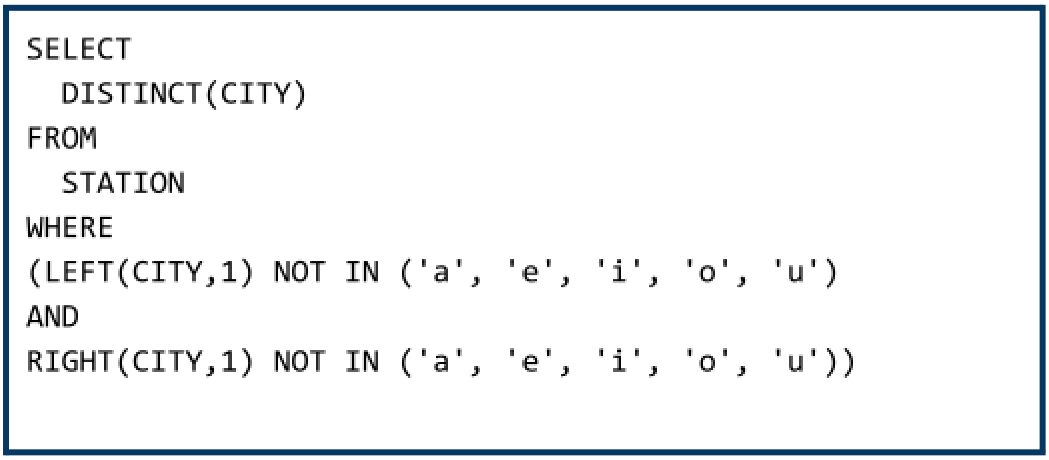


STATION

Field	Type
ID	NUMBER
CITY	VARCHAR2(21)
STATE	VARCHAR2(2)
LAT_N	NUMBER
LONG_W	NUMBER

where LAT_N is the northern latitude and LONG_W is the western longitude.

Solution:



```
SELECT
  DISTINCT(CITY)
FROM
  STATION
WHERE
(LEFT(CITY,1) NOT IN ('a', 'e', 'i', 'o', 'u')
AND
RIGHT(CITY,1) NOT IN ('a', 'e', 'i', 'o', 'u'))
```

2. **Weather Observation Station 7**

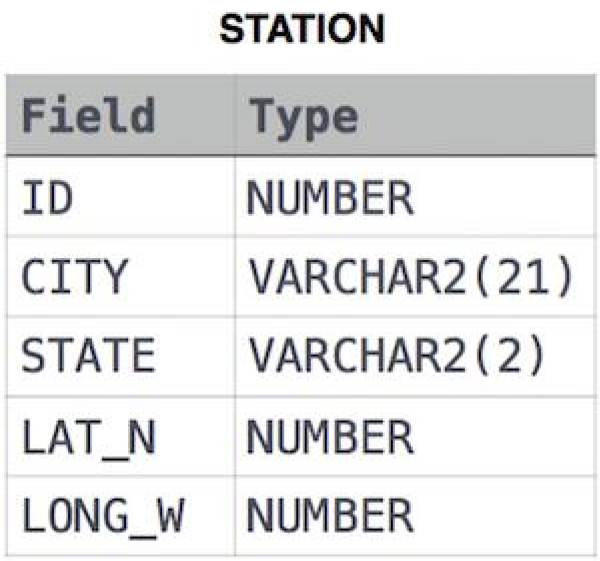
Link:
https://www.hackerrank.com/challenges/weather-observation-station-7/problem

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:



STATION

Field	Type
ID	NUMBER
CITY	VARCHAR2(21)
STATE	VARCHAR2(2)
LAT_N	NUMBER
LONG_W	NUMBER

where LAT_N is the northern latitude and LONG_W is the western longitude.

Solution:

```
SELECT
   DISTINCT(CITY)
FROM
   STATION
WHERE (RIGHT(CITY,1) IN ('a', 'e', 'i',
'o', 'u'))
```