/\*In the accounts table, there is a column holding the website for each company.

The last three digits specify what type of web address they are using.

Pull these extensions and provide how many of each website type exist in the accounts table.\*/

SELECT RIGHT (website, 3) AS domain, COUNT(\*) num\_companies

FROM accounts

GROUP BY 1

ORDER BY 2 DESC;

/\*There is much debate about how much the name (or even the first letter of a company name) matters.

Use the accounts table to pull the first letter of each company name to

see the distribution of company names that begin with each letter (or number)\*/

SELECT LEFT(UPPER(name), 1) AS first\_letter, COUNT(\*) num\_companies

FROM accounts

GROUP BY 1

ORDER BY 2 DESC;

/\* Use the accounts table and a CASE statement to create two groups:

one group of company names that start with a number and a second group of those company names that start with a letter.

What proportion of company names start with a letter?\*/

SELECT SUM(num) nums, SUM(letter) letters

FROM (SELECT name, CASE WHEN LEFT(UPPER(name), 1) IN ('0','1','2','3','4','5','6','7','8','9')

THEN 1 ELSE 0 END AS num,

CASE WHEN LEFT(UPPER(name), 1) IN ('0','1','2','3','4','5','6','7','8','9')

THEN 0 ELSE 1 END AS letter

FROM accounts) t1;

/\* Consider vowels as a, e, i, o, and u. What proportion of company names start with a vowel,

and what percent start with anything else?\*/

SELECT SUM(vowels) vowels, SUM(other) other

FROM (SELECT name, CASE WHEN LEFT(UPPER(name), 1) IN ('A','E','I','O','U')

THEN 1 ELSE 0 END AS vowels,

CASE WHEN LEFT(UPPER(name), 1) IN ('A','E','I','O','U')

THEN 0 ELSE 1 END AS other

FROM accounts) t1;

/\* use acounts table to create first & last name column that hold the first & last name for the Primary\_POC\*/

SELECT primary\_poc, LEFT(primary\_poc, STRPOS(accounts.primary\_poc, ' ')-1),

RIGHT(primary\_poc, LENGTH(primary\_poc)- STRPOS(accounts.primary\_poc, ' '))

FROM accounts

/\* now see if u can do same thing for every rep name in the sales\_reps table\*/

SELECT name full\_name, LEFT(name, STRPOS(s.name, ' ')-1) first\_name,

RIGHT(name, LENGTH(name)- STRPOS(s.name, ' ')) last\_name

FROM sales\_reps s

/\*Each company in the accounts table wants to create an email address for each primary\_poc.

the email address should be the first name of the primary\_poc . primary\_poc lastname @ company name.com\*/

WITH t1 As

(SELECT LEFT(primary\_poc, STRPOS(primary\_poc, ' ')-1) first\_name,

RIGHT(primary\_poc, LENGTH(primary\_poc)- STRPOS(primary\_poc, ' ')) last\_name, name

FROM accounts)

SELECT first\_name, last\_name, CONCAT(first\_name, '.', last\_name, '@', name, '.com')

FROM t1

/\*remove space in the company name part of the email\*/

WITH t1 As

(SELECT LEFT(primary\_poc, STRPOS(primary\_poc, ' ')-1) first\_name,

RIGHT(primary\_poc, LENGTH(primary\_poc)- STRPOS(primary\_poc, ' ')) last\_name, name

FROM accounts)

SELECT first\_name, last\_name, CONCAT(first\_name, '.', last\_name, '@', REPLACE(name, ' ', ''),'.com')

FROM t1

/\*We would also like to create an initial password,

which they will change after their first log in.

The first password will be the first letter of the primary\_poc’s first name (lowercase),

then the last letter of their first name (lowercase), the first letter of their last name (lowercase),

the last letter of their last name (lowercase), the number of letters in their first name,

the number of letters in their last name, and then the name of the company they are working with,

all capitalized with no spaces.\*/

WITH t1 AS (

SELECT LEFT(primary\_poc, STRPOS(primary\_poc, ' ') -1 ) first\_name,

RIGHT(primary\_poc,

LENGTH(primary\_poc)-STRPOS(primary\_poc, ' ')) last\_name,

name

FROM accounts)

SELECT first\_name, last\_name, CONCAT(first\_name, '.', last\_name, '@', name, '.com'),

LEFT(LOWER(first\_name), 1) || RIGHT(LOWER(first\_name), 1) || LEFT(LOWER(last\_name), 1) || RIGHT(LOWER(last\_name), 1)

|| LENGTH(first\_name) || LENGTH(last\_name) || REPLACE(UPPER(name), ' ', '')

FROM t1

/\*Each company in the sales\_rep table wants to create an email address for each sales\_rep.

the email address should be the first name of the sales\_reps . sales\_reps lastname @ region\_name .com\*/

SELECT first\_name, last\_name, CONCAT(first\_name, '.', last\_name, '@', region\_name, '.com')

FROM (

SELECT LEFT(sales\_rep\_name, STRPOS(sales\_rep\_name, ' ')-1) first\_name,

RIGHT(sales\_rep\_name, LENGTH(sales\_rep\_name) - STRPOS(sales\_rep\_name, ' ')) last\_name, region\_name

FROM (SELECT s.name sales\_rep\_name, r.name region\_name

FROM sales\_reps s

JOIN region r

ON s.region\_id = r.id))

/\*We would also like to create an initial password,

which they will change after their first log in.

The first password will be the first letter of the first name (lowercase),

then the last letter of their first name (lowercase), the first letter of their last name (lowercase),

the last letter of their last name (lowercase), the number of letters in their first name,

the number of letters in their last name, and then the name of the region they are working with,

all capitalized with no spaces.\*/

--CTE mtd

WITH t2 AS (

SELECT LEFT(sales\_rep\_name, STRPOS(sales\_rep\_name, ' ')-1) first\_name,

RIGHT(sales\_rep\_name, LENGTH(sales\_rep\_name) - STRPOS(sales\_rep\_name, ' ')) last\_name, region\_name

FROM (SELECT s.name sales\_rep\_name, r.name region\_name

FROM sales\_reps s

JOIN region r

ON s.region\_id = r.id))

SELECT first\_name, last\_name, CONCAT(first\_name, '.', last\_name, '@', region\_name, '.com') email,

LEFT(LOWER(first\_name),1) || RIGHT(LOWER(first\_name),1) || LEFT(LOWER(last\_name),1) || RIGHT(LOWER(last\_name),1) ||

LENGTH(first\_name) || LENGTH(last\_name) || UPPER(region\_name) AS password

FROM t2

--subquery mtd

SELECT first\_name, last\_name, CONCAT(first\_name, '.', last\_name, '@', region\_name, '.com') email,

LEFT(LOWER(first\_name),1) || RIGHT(LOWER(first\_name),1) || LEFT(LOWER(last\_name),1) || RIGHT(LOWER(last\_name),1) ||

LENGTH(first\_name) || LENGTH(last\_name) || UPPER(region\_name) AS password

FROM (SELECT LEFT(sales\_rep\_name, STRPOS(sales\_rep\_name, ' ')-1) first\_name,

RIGHT(sales\_rep\_name, LENGTH(sales\_rep\_name) - STRPOS(sales\_rep\_name, ' ')) last\_name, region\_name

FROM (SELECT s.name sales\_rep\_name, r.name region\_name

FROM sales\_reps s

JOIN region r

ON s.region\_id = r.id))