

-- LEFT and RIGHT Quizzes

1. 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. A list of extensions (and pricing) is provided here. Pull these extensions and provide how many of each website type exist in the accounts table.

```
SELECT COUNT(*),  
       RIGHT(website,3)  
FROM accounts  
GROUP BY 2 DESC
```

2. 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(name,1) initial,  
       COUNT(*)  
FROM accounts  
GROUP BY 1  
ORDER BY 2 DESC
```

3. 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 CASE WHEN LEFT(name,1) IN  
('0','1','2','3','4','5','6','7','8','9') THEN 'number'  
          ELSE 'letter' END AS initial,  
       COUNT(*) count  
FROM accounts  
GROUP BY 1
```

-- $350/351 * 100 = 0.29$

4. 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?

```
WITH vowel AS (  
    SELECT COUNT(*) vowel  
    FROM accounts  
    WHERE LEFT(UPPER(name),1) IN  
    ('A','E','I','O','U')  
)  
SELECT ROUND(ROUND(vowel * 100,2) / (SELECT COUNT(*) FROM  
accounts),2) vowel_percentage  
FROM vowel
```

-- POSITION & STRPOS Quiz

5. Use the accounts table to create first and last name columns that hold the first and last names for the primary_poc.

```
select primary_poc,  
length(primary_poc) length_poc,  
position(' ' in primary_poc) position_space,  
strpos(primary_poc, ' ') strpos_space,  
left(primary_poc,position(' ' in primary_poc)-1)first_name,  
right(primary_poc,-strpos(primary_poc, ' '))last_name  
from accounts
```

6. Now see if you can do the same thing for every rep name in the sales_reps table. Again provide first and last name columns.

```
with t1 as (  
select left(name,strpos(name, ' ')-1) first_name,  
       right(name,-strpos(name, ' ')) last_name  
from sales_reps)  
  
select first_name,  
       length(first_name) len,  
       last_name,  
       length(last_name) len_last  
from t1
```

6. 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 . last name primary_poc @ company name .com.

```
select primary_poc,  
       left(lower(primary_poc), strpos(primary_poc, ' ')-1)  
|| right(lower(primary_poc),-strpos(primary_poc, ' '))  
|| '@' || REPLACE(LOWER(name), ' ', '') || '.com' as email  
from accounts
```

7. You may have noticed that in the previous solution some of the company names include spaces, which will certainly not work in an email address. See if you can create an email address that will work by removing all of the spaces in the account name, but otherwise your solution should be just as in question 1. Some helpful documentation is here.

same as 3.

8. 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 primary_poc,  
       left(lower(primary_poc),strpos(primary_poc, ' ')-1)  
first_name,  
       right(lower(primary_poc),-strpos(primary_poc, ' '))  
last_name,  
       lower(replace(name, ' ', '')) company_name      from  
accounts )
```

```
select
```

```

        primary_poc, left(first_name,1) || right(first_name,1)
        || left(last_name,1) || right(last_name,1) ||
        length(first_name) || length(last_name) ||
        replace(company_name, '.', '') as password from t1

```

-- CAST

9. Write a query to change the date into correct SQL format. You will need to use at least SUBSTR and CONCAT.

```

with t1 as (select trim(left(date, strpos(date, ' ')))
wrong_date from sf_crime_data
limit 10 ),
t2 as (

    select wrong_date,
           substring(wrong_date,1,2) clean_month,
           substring(wrong_date,4,2) clean_day,
           substring(wrong_date,7,4) clean_year
    from t1
)
select wrong_date, cast(clean_year || '-' || clean_month || '-' ||
clean_day as date) from t2

```

-- COALESCE

10. Use COALESCE to fill in the missing values.

```

SELECT *,
        COALESCE(a.id,a.id) filled_accounts_id,
        COALESCE(o.account_id,a.id)
filled_orders_account_id,
        COALESCE(standard_qty,0) filled_std_qty,
        COALESCE(gloss_qty,0) filled_gloss_qty,
        COALESCE(poster_qty,0) filled_poster_qty,
        COALESCE(standard_amt_usd,0)
filled_standard_amt_usd,
        COALESCE(gloss_amt_usd,0) filled_gloss_amt_usd,
        COALESCE(poster_amt_usd,0) filled_poster_amt_usd
FROM accounts a
LEFT JOIN orders o ON a.id = o.account_id

```