

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?
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?

Input

HISTORY ▾ MENU ▾

SCHEMA ↻

accounts ▾

orders ▾

region ▾

sales_reps ▾

web_events ▾

```
1 SELECT SUM(number) num,SUM(letter) let FROM
2 (SELECT a.name,CASE WHEN LEFT(a.name,1) IN
   ('0','1','2','3','4','5','6','7','8','9') THEN 1
   ELSE 0 END AS number,CASE WHEN LEFT (a.name,1) IN
   ('0','1','2','3','4','5','6','7','8','9') THEN 0
   ELSE 1 END AS letter
3 FROM accounts a
4 GROUP BY a.name)t1
```

Success!

EVALUATE

Output 1 results

Download CSV

num	let
1	350

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?

Input

HISTORY ▾

MENU ▾

SCHEMA

accounts ▾

orders ▾

region ▾

sales_reps ▾

web_events ▾

```
1 SELECT ((SUM(vowels))*100/COUNT(*)) vow,
  ((SUM(not_vowels))*100/COUNT(*)) not_vow FROM
  (SELECT CASE WHEN LEFT(a.name,1) IN
  ('A','E','I','O','U') THEN 1 ELSE 0 END AS
  vowels,CASE WHEN LEFT (a.name,1) IN
  ('A','E','I','O','U') THEN 0 ELSE 1 END AS
  not_vowels
  FROM accounts a)+1
```

Success!

EVALUATE

Output 1 results

Download CSV


vow	not_vow
22	77

1. Use the `accounts` table to create **first** and **last** name columns that hold the first and last names for the `primary_poc`.
2. 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.

Input

HISTORY ▾

MENU ▾

SCHEMA 

primary_poc

sales_rep_id

orders ▾

region ▾

sales_reps ▾

```
1 SELECT primary_poc,
   LEFT(primary_poc,STRPOS(primary_poc,' ')) AS
   first_name,
2  RIGHT(primary_poc,LENGTH(primary_poc)-
   STRPOS(primary_poc,' ')) last_name FROM accounts
```

Success!

EVALUATE

Output 351 results


primary_poc	first_name	last_name
Tamara Tuma	Tamara	Tuma
Sung Shields	Sung	Shields
Jodee Lupo	Jodee	Lupo
Serafina Banda	Serafina	Banda
Angeles Crusoe	Angeles	Crusoe

2. 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.

Input

HISTORY ▾

MENU ▾

SCHEMA 

sales_reps ^

id

name

region_id

web_events ▾

1 SELECT name, LEFT(name,STRPOS(name,' ')) AS first_name,

2 RIGHT(name,LENGTH(name)-STRPOS(name,' ')) last_name

FROM sales_reps

Success!

EVALUATE

Output 50 results

name	first_name	last_name
Samuel Racine	Samuel	Racine
Eugena Esser	Eugena	Esser
Michel Averette	Michel	Averette
Renetta Carew	Renetta	Carew
Cara Clarke	Cara	Clarke

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.

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](#).

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(LOWER(primary_poc),STRPOS(primary_poc,' ')-1) first_name,

RIGHT(LOWER(primary_poc),LENGTH(primary_poc)-STRPOS(primary_poc,' ')) Last_name,

LOWER(name) email

FROM accounts)

SELECT (first_name||'.'||Last_name||'@'||REPLACE(email,' ','')||'.com') email,

(LEFT(first_name,1)||RIGHT(first_name,1)||LEFT>Last_name,1)||RIGHT>Last_name,1)||LENGTH(first_name)||LENGTH>Last_name)||email)

FROM t1

Tasks to complete:



1. Write a query to look at the top 10 rows to understand the columns and the raw data in the dataset called `sf_crime_data`.



2. Remembering back to the lesson on dates, use the **Quiz Question** at the bottom of this page to make sure you remember the format that dates should use in SQL.



3. Look at the `date` column in the `sf_crime_data` table. Notice the date is not in the correct format.



4. Write a query to change the date into the correct SQL date format. You will need to use at least **SUBSTR** and **CONCAT** to perform this operation.



5. Once you have created a column in the correct format, use either `CAST` or `::` to convert this to a date.

Note: If the proper tables for this SF Crime Data database do not appear for some reason in the Schema below, you can fix this using the Menu in the lower left of the workspace. Make sure you have first saved any query you have entered, then click on Menu, then choose Reset Data, and type in "Reset Data" as directed. This will definitely bring up the proper schema tables for the SF Crime Data database.

Input

HISTORY ▾

MENU ▾

SCHEMA

sf_crime_data

incidnt_num

category

descript

day_of_week

▲

▼

1

2

SELECT * FROM sf_crime_data

LIMIT 10

Success!

EVALUATE

Output

10 results

Download CSV

ress	lon	lat	location
ock of GARRISON AV	-122.413623946206	37.709725805163	(37.709725805163, -122
Block of FONT BL	-122.47370623066	37.7154876086057	(37.7154876086057, -12
ock of CASTRO ST	-122.435718550322	37.7686887134351	(37.7686887134351, -12
FRSON ST / POWELL ST	-122.417577739682	37.8086250595467	(37.8086250595467, -12



- ✓ 3. Look at the `date` column in the `sf_crime_data` table. Notice the date is not in the correct format.
- ✓ 4. Write a query to change the date into the correct SQL date format. You will need to use at least **SUBSTR** and **CONCAT** to perform this operation.
- ✓ 5. Once you have created a column in the correct format, use either `CAST` or `::` to convert this to a date.

Note: If the proper tables for this SF Crime Data database do not appear for some reason in the Schema below, you can fix this using the Menu in the lower left of the workspace. Make sure you have first saved any query you have entered, then click on Menu, then choose Reset Data, and type in "Reset Data" as directed. This will definitely bring up the proper schema tables for the SF Crime Data database.

Input

HISTORY ▾

ME

SCHEMA

sf_crime_data

incidnt_num

category

descript

day_of_week

1

With t1 AS (SELECT SUBSTR(date,1,2) month1, SUBSTR(date,4,2) day1, SUBSTR(date,7,4) year1, sf_crime_data)

2

3

SELECT (year1 || '-' || month1 || '-' || day1)::date FROM t1

Success!

EVALUATE


Output

30400 results

Download

date1

2014-01-31T00:00:00.000Z

- 
- ✓ 1. Run the query entered below in the SQL workspace to notice the row with missing data.
 - ✓ 2. Use **COALESCE** to fill in the `accounts.id` column with the `account.id` for the NULL value for the table in 1.
 - ✓ 3. Use **COALESCE** to fill in the `orders.account_id` column with the `account.id` for the NULL value for the table in 1.
 - ✓ 4. Use **COALESCE** to fill in each of the `qty` and `usd` columns with 0 for the table in 3.
 - ✓ 5. Run the query in 1 with the **WHERE** removed and **COUNT** the number of `ids`.
 - ✓ 6. Run the query in 5, but with the **COALESCE** function used in questions 2 through 4.

```

WITH t1 AS (SELECT COALESCE(o.id,a.id) fill_id,
a.name a.website a.lat a.long a.primary_poc a.sales_rep_id, o.occurred_at,
COALESCE(o.account_id, a.id) orders_fill_id,
COALESCE(o.standard_qty,0) fill_std,
COALESCE(o.gloss_qty,0) fill_goss,
COALESCE(o.poster_qty,0) fill_poster,
COALESCE(o.total,0) fill_total,
COALESCE(o.standard_amt_usd,0) fill_std_amt,
COALESCE(o.gloss_amt_usd,0) fill_gloss_amt,
COALESCE(o.poster_amt_usd,0) fill_poster_amt,
COALESCE(o.total_amt_usd,0) fill_tot_amt
FROM accounts a
LEFT JOIN orders o
ON a.id = o.account_id)

```

```

select count(*) from t1

```


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).
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?
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?

Input

HISTORY ▾

MENU ▾

SCHEMA ↻

accounts ▾

orders ▾

region ▾

```
1 SELECT COUNT(*), LEFT(a.name,1) as com_name
2 FROM accounts a
3 GROUP BY 2
4 ORDER BY 1 DESC
```

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.
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).
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?
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?

Input

HISTORY ▾

MENU ▾

SCHEMA ↻

accounts ▾

orders ▾

region ▾

sales_reps ▾

web_events ▾

```
1 SELECT COUNT(*),RIGHT(a.website,3) as ext
2 FROM accounts a
3 GROUP BY ext
4 ORDER BY 1 DESC
```

Success!

EVALUATE

Output 3 results

Download CSV

count	ext
349	com
1	net
1	org