* **SQL Query:**
* **Task 01:**

1. Write a query that returns all the orders where the standard\_qty is over 1000, the poster\_qty is 0, and the gloss\_qty is 0.

Query:

SELECT \* FROM orders WHERE standard\_qty > 1000

AND poster\_qty = 0

AND gloss\_qty = 0;

1. Using the accounts table find all the companies whose names do not start with 'C' and end with 's'.

Query:

SELECT \* FROM accounts

WHERE name NOT LIKE 'C%''s%';

1. Use the web\_events table to find all information regarding individuals who were contacted via organic or adwords and started their account at any point in 2016 sorted from newest to oldest.

Query:

SELECT \* FROM web\_events

WHERE channel IN ('adwords','organic') AND occurred\_at >= '2016-01-01'

ORDER BY occurred\_at DESC;

* **Task 02:**

1. Provide a table that provides the **region** for each **sales\_rep** along with their associated **accounts**. This time only for the Midwest region. Your final table should include three columns: the region **name**, the sales rep **name**, and the account **name**. Sort the accounts alphabetically (A-Z) according to account name.

Query:

SELECT r.name region, s.name rep, a.name account

FROM sales\_reps s

JOIN region r

ON s.region\_id = r.id

JOIN accounts a

ON a.sales\_rep\_id = s.id

WHERE r.name = 'Midwest'

ORDER BY a.name;

1. Provide a table that provides the **region** for each **sales\_rep** along with their associated **accounts**. This time only for accounts where the sales rep has a first name starting with S and in the Midwest region. Your final table should include three columns: the region **name**, the sales rep **name**, and the account **name**. Sort the accounts alphabetically (A-Z) according to account name.

Query:

SELECT r.name region, s.name rep, a.name account

FROM sales\_reps s

JOIN region r

ON s.region\_id = r.id

JOIN accounts a

ON a.sales\_rep\_id = s.id

WHERE r.name = 'Midwest' AND s.name LIKE 'S%'

ORDER BY a.name;

1. Provide a table that provides the **region** for each **sales\_rep** along with their associated **accounts**. This time only for accounts where the sales rep has a **last** name starting with K and in the Midwest region. Your final table should include three columns: the region **name**, the sales rep **name**, and the account **name**. Sort the accounts alphabetically (A-Z) according to account name.

Query:

SELECT r.name region, s.name rep, a.name account

FROM sales\_reps s

JOIN region r

ON s.region\_id = r.id

JOIN accounts a

ON a.sales\_rep\_id = s.id

WHERE r.name = 'Midwest' AND s.name LIKE '% K%'

ORDER BY a.name;

* **Task 03:**

1. Find the total amount spent on standard\_amt\_usd and gloss\_amt\_usd paper for each order in the orders table. This should give a dollar amount for each order in the table.

Query:

SELECT standard\_amt\_usd + gloss\_amt\_usd total\_amount

FROM orders;

1. Find the mean (AVERAGE) amount spent per order on each paper type, as well as the mean amount of each paper type purchased per order. Your final answer should have 6 values - one for each paper type for the average number of sales, as well as the average amount.

Query:

SELECT AVG(standard\_qty) standard\_qty,

AVG(gloss\_qty) gloss\_qty,

AVG(poster\_qty) poster\_qty,

AVG(standard\_amt\_usd) standard\_amt\_usd,

AVG(gloss\_amt\_usd) gloss\_amt\_usd,

AVG(poster\_amt\_usd) poster\_amt\_usd

FROM orders;

1. Which account used facebook most as a channel?

Query:

SELECT account\_id, channel\_name, COUNT(\*) AS usage\_count

FROM channel\_activities

WHERE channel\_name = 'Facebook'

GROUP BY account\_id, channel\_name

ORDER BY usage\_count DESC

LIMIT 1;

* **Task 04:**

1. Use DATE\_TRUNC or EXTRACT to pull month level information about the first order ever placed in the orders table.

Query:

SELECT channel,

AVG(event\_count)

FROM (SELECT

DATE\_TRUNC('day', occurred\_at) AS day,

COUNT(\\*) AS event\_count,

Channel

* **Task 05:**

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](https://iwantmyname.com/domains/domain-name-registration-list-of-extensions). Pull these extensions and provide how many of each website type exist in the accounts table.

Query:

SELECT

RIGHT(website, 3) AS extension,

COUNT(\*) AS count

FROM accounts

GROUP BY extension

ORDER BY count DESC;