**20SW035**

**SECTION – 01**

**LAB#06**

**Task#01 – Basic SQL -Run the following queries:**

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.**

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'.**

SELECT \*

FROM accounts

WHERE NOT (company\_name LIKE 'C%' AND company\_name LIKE '%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**

SELECT \*

FROM web\_events

WHERE channel IN ('organic', 'adwords')

  AND DATE\_PART ('year', occurred\_at) = 2016

ORDER BY occurred\_at DESC;

**Task#02 – Joins -Run the following queries:**

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 (AZ) according to account name.**

SELECT r.region\_name, s.sales\_rep\_name, a.account\_name

FROM sales\_reps s

JOIN accounts a ON s.sales\_rep\_id = a.sales\_rep\_id

JOIN regions r ON s.region\_id = r.region\_id

WHERE r.region\_name = 'Midwest'

ORDER BY a.account\_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 17 rep name, and the account name. Sort the accounts alphabetically (AZ) according to account name.**

SELECT r.region\_name, s.sales\_rep\_name, a.account\_name

FROM sales\_reps s

JOIN accounts a ON s.sales\_rep\_id = a.sales\_rep\_id

JOIN regions r ON s.region\_id = r.region\_id

WHERE r.region\_name = 'Midwest'

  AND s.sales\_rep\_name LIKE 'S%'

ORDER BY a.account\_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 (AZ) according to account name**

SELECT r.region\_name, s.sales\_rep\_name, a.account\_name

FROM sales\_reps s

JOIN accounts a ON s.sales\_rep\_id = a.sales\_rep\_id

JOIN regions r ON s.region\_id = r.region\_id

WHERE r.region\_name = 'Midwest'

  AND s.sales\_rep\_name LIKE '% K%'

ORDER BY a.account\_name;

**Task#03 – SQL Aggregations -Run the following queries:**

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.**

SELECT order\_id,

       SUM(standard\_amt\_usd) AS total\_standard\_amount,

       SUM(gloss\_amt\_usd) AS total\_gloss\_amount

FROM orders

GROUP BY order\_id;

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.**

SELECT

    AVG(standard\_amt\_usd) AS avg\_standard\_amount\_per\_order,

    AVG(gloss\_amt\_usd) AS avg\_gloss\_amount\_per\_order,

    AVG(standard\_amt\_usd + gloss\_amt\_usd) AS avg\_total\_amount\_per\_order

FROM orders;

1. **Which account used facebook most as a channel**

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

FROM web\_events

WHERE channel = 'Facebook'

GROUP BY account\_id, channel

ORDER BY channel\_count DESC

LIMIT 1;

**Task#04 –SQL Subqueries-Run the following queries:**

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

SELECT DATE\_TRUNC('month', order\_date) AS month\_of\_first\_order

FROM orders

ORDER BY order\_date

LIMIT 1;

**Task#05 –SQL Data Cleaning -Run the following queries:**

**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

    RIGHT(website, 3) AS website\_extension,

    COUNT(\*) AS extension\_count

FROM accounts

GROUP BY website\_extension

ORDER BY extension\_count DESC;