

```
1  USE PRACTICE;
2
3  SELECT * FROM product_spend;
4
5
6  --BUSINESS QUESTION
7  /** Question 1 **/
8  --Is Revenue(Spend) increasing or decreasing over time?(per Category per month)
9  --1. Monthly Revenue Trend(FOUNDATION)
10
11
12  --MONTH OVER MONTH GROWTH
13  /**Month Over Month Growth**/
14  SELECT
15      DATEFROMPARTS(YEAR(transaction_date), MONTH(transaction_date), 1) AS date, -- full date for ordering
16      FORMAT(transaction_date, 'MMM') AS month, -- short month name (Jan, Feb...)
17      category,
18      SUM(spend) AS monthly_revenue
19  FROM product_spend
20  GROUP BY
21      DATEFROMPARTS(YEAR(transaction_date), MONTH(transaction_date), 1),
22      FORMAT(transaction_date, 'MMM'),
23      category
24  ORDER BY date;
25
26  /**Month Over Month Trends Per Category **/
27  WITH monthly_revenue AS (
28      SELECT
29          FORMAT(transaction_date, 'MMMM') AS month, YEAR(transaction_date) AS year, category, SUM(spend) AS revenue
30      FROM product_spend
31      GROUP BY YEAR(transaction_date), category, FORMAT(transaction_date, 'MMMM')
32  )
33  SELECT category, year, month, revenue, COALESCE(revenue - LAG(revenue) OVER (PARTITION BY category ORDER BY year), 0) AS M_O_M_Growth
34  FROM monthly_revenue;
35
36
37
38  ---YEAR OVER YEAR GROWTH (YoY)
39  --Question 2: Is Revenue(Spend) increasing or decreasing over time?(per Category per month)
40
41  /** Year Over Year Growth **/
```

```
42 ---Yearly Revenue Trend(FOUNDATION)
43 SELECT YEAR(transaction_date) AS year,
44 category,
45 SUM(spend) AS revenue, COALESCE(LAG(spend) OVER (PARTITION BY category
46 ORDER BY YEAR(transaction_date)), 0) AS Previous_revenue
47 FROM product_spend
48 GROUP BY YEAR(transaction_date), category, spend
49 ;
50 --Y_O_Y_Growth(Per Category)
51 WITH yearly_revenue AS(
52 SELECT
53     YEAR(transaction_date) AS year,
54     category,
55     SUM(spend) AS revenue
56 FROM product_spend
57 GROUP BY YEAR(transaction_date), category
58 )
59 --SELECT * from yearly_revenue;
60 SELECT category , year, revenue, COALESCE (revenue - LAG(revenue) OVER
61 (PARTITION BY category ORDER BY year), 0) AS Y_O_Y_Growth
62 FROM yearly_revenue
63 ;
64 --Y_O_Y_Growth(Per Product)
65 WITH yearly_revenue AS(
66 SELECT
67     YEAR(transaction_date) AS year,
68     product,
69     SUM(spend) AS revenue
70 FROM product_spend
71 GROUP BY YEAR(transaction_date), product
72 )
73 --SELECT * from yearly_revenue;
74 SELECT product , year, revenue, COALESCE (revenue - LAG(revenue) OVER
75 (PARTITION BY product ORDER BY year), 0) AS Y_O_Y_Growth
76 FROM yearly_revenue
77 ;
78
79 --QUESTION 3: (Slow Moving Products)
80 --Identify Slow moving products
81 SELECT
82     product,
83     COUNT(*) AS total_sales,
84     SUM(spend) AS total_revenue
85 FROM product_spend
86 GROUP BY product
87 ORDER BY total_revenue ASC, total_sales ASC;
```

```
88
89
90  --QUESTION 5(Cummulative Revenue)
91  --Generate the Cummulative revenue per Product
92
93  SELECT
94      transaction_date,
95      product,
96      spend,
97      SUM(spend) OVER (
98          PARTITION BY product
99          ORDER BY transaction_date
100      ) AS cumulative_revenue
101  FROM product_spend
102  ;
103
104
105  --Genreate the Cummulative Revenue per Category
106  SELECT
107      transaction_date,
108      category,
109      spend,
110      SUM(spend) OVER (
111          PARTITION BY category
112          ORDER BY transaction_date
113      ) AS cumulative_revenue
114  FROM product_spend;
```