

Case Study #7 - Balanced Tree

A. High Level Sales Analysis

1. What was the total quantity sold for all products?

```
SELECT
    product_name,
    SUM(qty) AS total_quantity
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY product_name
```

2. What is the total generated revenue for all products before discounts?

```
SELECT
    product_name,
    SUM(qty * s.price) AS total_revenue
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY product_name
```

3. What was the total discount amount for all products?

```
SELECT
    product_name,
    SUM(s.price * qty * discount/100) AS total_discount
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY product_name
```

B. Transaction Analysis

1. How many unique transactions were there?

```
SELECT
    COUNT(DISTINCT txn_id) AS trans_count
FROM sales
```

2. What is the average unique products purchased in each transaction?

```
SELECT
    ROUND(SUM(qty) / (SELECT COUNT(DISTINCT txn_id) FROM sales))
FROM sales AS s
```

3. What are the 25th, 50th and 75th percentile values for the revenue per transaction?

```
WITH revenue_quartiles AS (
    SELECT
        SUM(price*qty) AS revenue,
        NTILE(4) OVER(ORDER BY SUM(price*qty)) AS quartile
    FROM sales
    GROUP BY txn_id
)
SELECT
    MAX(CASE WHEN quartile = 1 THEN revenue END) AS percentile_25,
    MAX(CASE WHEN quartile = 2 THEN revenue END) AS percentile_50,
    MAX(CASE WHEN quartile = 3 THEN revenue END) AS percentile_75
FROM revenue_quartiles;
```

4. What is the average discount value per transaction?

```
SELECT
    ROUND(SUM(price * qty * discount/100)/(SELECT COUNT(DISTINCT txn_id)
FROM sales
```

5. What is the percentage split of all transactions for members vs non-members?

```
SELECT
    ROUND(SUM(IF(member,1,0)) / (SELECT COUNT(txn_id) FROM sales)) *
    100 - ROUND(SUM(IF(member,1,0)) / (SELECT COUNT(txn_id) FROM sales)) *
    100
FROM sales
```

6. What is the average revenue for member transactions and non-member transactions?

```
SELECT
    ROUND(SUM(IF(member,qty*price,0))/(SELECT COUNT(DISTINCT txn_id) FROM sales)) *
    100 - ROUND(SUM(IF(!member,qty*price,0))/(SELECT COUNT(DISTINCT txn_id) FROM sales)) *
    100
FROM sales
```

C. Product Analysis

1. What are the top 3 products by total revenue before discount?

```
SELECT
    product_name,
    SUM(qty * s.price) AS total_revenue
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY product_name
ORDER BY total_revenue DESC
LIMIT 3
```

2. What is the total quantity, revenue and discount for each segment?

```
SELECT
    segment_name,
    SUM(qty) as total_quantity,
    SUM(qty * s.price) AS total_revenue,
    SUM(qty * s.price * discount/100) AS total_discount
FROM sales AS s
```

```
JOIN product_details AS pd ON pd.product_id = s.prod_id  
GROUP BY segment_name;
```

3. What is the top selling product for each segment?

```
SELECT  
    segment_name,  
    product_name,  
    total_quantity  
FROM  
(SELECT  
    product_name,  
    segment_name,  
    SUM(qty) AS total_quantity,  
    DENSE_RANK() OVER (PARTITION BY segment_name ORDER BY SUM(qty) DESC) AS ranking  
FROM sales AS s  
JOIN product_details AS pd ON pd.product_id = s.prod_id  
GROUP BY product_name, segment_name) as x  
WHERE ranking = 1
```

4. What is the total quantity, revenue and discount for each category?

```
SELECT  
    category_name,  
    SUM(qty) as total_quantity,  
    SUM(qty * s.price) AS total_revenue,  
    SUM(qty * s.price * discount/100) AS total_discount  
FROM sales AS s  
JOIN product_details AS pd ON pd.product_id = s.prod_id  
GROUP BY category_name;
```

5. What is the top selling product for each category?

```
SELECT  
    category_name,  
    product_name,
```

```

        total_quantity
FROM
(SELECT
    product_name,
    category_name,
    SUM(qty) AS total_quantity,
    DENSE_RANK() OVER (PARTITION BY category_name ORDER BY SUM(qty)) AS ranking
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY product_name,category_name) as x
WHERE ranking = 1

```

6. What is the percentage split of revenue by product for each segment?

```

WITH total_revenue_segment AS
(SELECT
    segment_name,
    SUM(qty * s.price) AS total_revenue
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY segment_name)
SELECT
    pd.segment_name,
    product_name,
    SUM(qty*s.price) / total_revenue * 100 as revenue_percentage
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
JOIN total_revenue_segment AS trs ON trs.segment_name = pd.segment_name
GROUP BY pd.segment_name,product_name
ORDER BY pd.segment_name

```

7. What is the percentage split of revenue by segment for each category?

```

WITH total_revenue_category AS
(SELECT
    category_name,

```

```

        SUM(qty * s.price) AS total_revenue
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY category_name)
SELECT
    pd.category_name,
    product_name,
    SUM(qty*s.price) / total_revenue * 100 as revenue_percentage
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
JOIN total_revenue_category AS trc ON trc.category_name = pd.category_name
GROUP BY pd.category_name, product_name
ORDER BY pd.category_name

```

8. What is the percentage split of total revenue by category?

```

SELECT
    category_name,
    SUM(qty * s.price)/(SELECT SUM(qty*price) OVER() FROM sales) as percentage
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY category_name

```

9. What is the total transaction “penetration” for each product? (hint: penetration = number of transactions where at least 1 quantity of a product was purchased divided by total number of transactions)

```

WITH count_prod_trans AS
(
SELECT
    product_name,
    txn_id,
    ROW_NUMBER() OVER(PARTITION BY product_name) as ranking
FROM sales AS s
JOIN product_details AS pd ON pd.product_id = s.prod_id
GROUP BY product_name,txn_id)

```

```
SELECT
    DISTINCT product_name,
    LAST_VALUE(ranking) OVER(PARTITION BY product_name) / (SELECT
FROM count_prod_trans
```

10. What is the most common combination of at least 1 quantity of any 3 products in a 1 single transaction?