



MAVEN COFFEE SALES ANALYSIS USING SQL



TABLE

```
CREATE TABLE transactions (
    transaction_id INT,
    transaction_date DATE,
    transaction_time TIME,
    transaction_qty INT,
    store_id INT,
    store_location VARCHAR(100),
    product_id INT,
    unit_price DECIMAL(10, 2),
    product_category VARCHAR(50),
    product_type VARCHAR(100),
    product_detail VARCHAR(100)
)
```



-- Find the top 5 dates with the highest total sales:

```
SELECT  
TRANSACTION_DATE,  
SUM(TRANSACTION_QTY*UNIT_PRICE) AS TOTAL_SALES  
FROM TRANSACTIONS  
GROUP BY 1  
ORDER BY SUM(TRANSACTION_QTY*UNIT_PRICE) DESC  
LIMIT 5;
```

Result Grid | Filter Rows:

	TRANSACTION_DATE	TOTAL_SALES
1	2023-06-19	6403.91
2	2023-06-13	6189.36
3	2023-06-08	6151.59
4	2023-06-17	6117.60
5	2023-06-18	6026.09

-- Determine peak transaction Time (by quantity):

```
SELECT  
CASE  
    WHEN EXTRACT(HOUR FROM TRANSACTION_TIME) BETWEEN 6 AND 10 THEN 'MORNING'  
    WHEN EXTRACT(HOUR FROM TRANSACTION_TIME) BETWEEN 11 AND 16 THEN 'AFTERNOON'  
    WHEN EXTRACT(HOUR FROM TRANSACTION_TIME) BETWEEN 17 AND 20 THEN 'EVENING'  
    ELSE 'NIGHT'  
END AS TIME,  
SUM(TRANSACTION_QTY) AS TOTAL_QTY  
FROM TRANSACTIONS  
GROUP BY TIME  
ORDER BY SUM(TRANSACTION_QTY) DESC;  
-- LIMIT 1;
```



TIME	TOTAL_QTY
MORNING	103594
AFTERNOON	77875
EVENING	33001

-- Get all transactions where unit_price is higher than the average price of its category:

```
SELECT *
FROM TRANSACTIONS
WHERE UNIT_PRICE >
    (SELECT
        AVG(UNIT_PRICE) AS AVG
    FROM TRANSACTIONS);
```

store_id	store_location	product_id	unit_price	product_category
5	Lower Manhattan	59	4.50	Drinking Chocolate
5	Lower Manhattan	39	4.25	Coffee
5	Lower Manhattan	58	3.50	Drinking Chocolate
5	Lower Manhattan	33	3.50	Coffee

-- Rank product types by sales within each category:

```
SELECT  
    PRODUCT_CATEGORY,  
    PRODUCT_TYPE,  
    SUM(TRANSACTION_QTY*UNIT_PRICE) AS REVENUE,  
    RANK() OVER(PARTITION BY PRODUCT_CATEGORY ORDER BY SUM(TRANSACTION_QTY*UNIT_PRICE) DESC) AS SALES_RANK  
FROM TRANSACTIONS  
GROUP BY 1,2;
```



PRODUCT_CATEGORY	PRODUCT_TYPE	REVENUE	SALES_RANK
Bakery	Scone	36866.12	1
Bakery	Pastry	25655.99	2
Bakery	Biscotti	19793.53	3
Branded	Housewares	7444.00	1
Branded	Clothing	6163.00	2

-- Find the top-selling product per store by revenue:

```
WITH CTE AS(
  SELECT
    STORE_ID,
    PRODUCT_ID,
    SUM(UNIT_PRICE * TRANSACTION_QTY) AS REVENUE,
    RANK() OVER(PARTITION BY STORE_ID ORDER BY SUM(UNIT_PRICE * TRANSACTION_QTY) DESC)
  FROM transactions
  GROUP BY STORE_ID, PRODUCT_ID)

SELECT *
FROM CTE
WHERE RANK_REV =1;
```



STORE_ID	PRODUCT_ID	REVENUE	RANK_REV
3	59	7897.50	1
5	59	6574.50	1
8	8	7380.00	1

-- Monthly sales trend by product category:

```
SELECT  
DATE_FORMAT(TRANSACTION_DATE, '%Y-%m') AS MONTH,  
PRODUCT_CATEGORY,  
SUM(UNIT_PRICE * TRANSACTION_QTY) AS TOTAL_SALES  
FROM TRANSACTIONS  
GROUP BY MONTH, PRODUCT_CATEGORY  
ORDER BY MONTH, PRODUCT_CATEGORY DESC ;
```



MONTH	PRODUCT_CATEGORY	TOTAL_SALES
2023-01	Tea	22621.95
2023-01	Packaged Chocolate	521.94
2023-01	Loose Tea	1293.85
2023-01	Flavours	974.40
2023-01	Drinking Chocolate	8337.75
2023-01	Coffee beans	5245.25

-- List dates where sales increased more than 30% compared to the previous day:

```
WITH DAILY_SALES AS(
  SELECT
    TRANSACTION_DATE,
    SUM(UNIT_PRICE * TRANSACTION_QTY) AS REVENUE
  FROM TRANSACTIONS
  GROUP BY TRANSACTION_DATE),  
  
SALES_WITH_LAG AS(
  SELECT
    TRANSACTION_DATE,
    REVENUE,
    LAG(REVENUE) OVER(ORDER BY TRANSACTION_DATE) AS PEREVIOUS_DAY_SALES
  FROM DAILY_SALES)  
  
SELECT *
  FROM SALES_WITH_LAG
  WHERE REVENUE > 1.3*PEREVIOUS_DAY_SALES;
```

TRANSACTION_DATE	REVENUE	PEREVIOUS_DAY_SALES
2023-01-13	3033.60	2327.70
2023-03-01	3040.25	2311.10
2023-05-01	4731.45	3552.33



-- Find the slowest-selling product types (sold at least 5 times but with lowest revenue:

```
SELECT  
PRODUCT_TYPE,  
SUM(TRANSACTION_QTY) AS TOTAL_QTY,  
SUM(UNIT_PRICE * TRANSACTION_QTY) AS REVENUE  
FROM TRANSACTIONS  
GROUP BY PRODUCT_TYPE  
HAVING SUM(TRANSACTION_QTY) >=5  
ORDER BY SUM(UNIT_PRICE * TRANSACTION_QTY) ASC  
LIMIT 5;
```



PRODUCT_TYPE	TOTAL_QTY	REVENUE
Green beans	134	1340.00
Green tea	159	1470.75
Organic Chocolate	221	1679.60
Sugar free syrup	2905	2324.00
Black tea	303	2711.85



Thank's For Watching



RESOURCE PAGE

