

# COFFEE SHOP SALES ANALYSIS



## My SQL Project

- **Objective:**  
Analyze coffee sales data to derive insights and improve business strategies.
- **Goals:**  
Understand transaction details such as IDs, quantities, and prices.  
Segment data by store locations to identify performance variations.  
Calculate metrics like total sales, average prices, and popular products for informed decision-making.

## Question Modes

- **Basic** - Queries include: SELECT, GROUP BY, ORDER BY, LIMIT, DESC.
- **Moderate** - Queries Include: JOINS, GROUP BY, ORDER BY, LIMIT.
- **Advance** - Queries Include: CTE (COMMON TABLE EXPRESSION).



pngtree.com

By Anand Sahu

# COFFEE SHOP SALES ANALYSIS



## Level -Basic

Q1 fetch all unique product category

Input:

```
SELECT DISTINCT product_category FROM coffeesales;
```

Output:

product_category	
Coffee	
Tea	
Drinking Chocolate	
Bakery	
Flavours	
Loose Tea	
Coffee beans	
Packaged Chocolate	
Branded	

Q2 fetch total transition in every area by descending order of transition

Input:

```
SELECT store_location,COUNT(*)AS total_transition FROM coffeesales  
GROUP BY store_location ORDER BY total_transition DESC;
```

Output:

store_location	total_transition
Astoria	6924
Hell's Kitchen	6823
Lower Manhattan	6429

Q3 total no of sales on date '1/1/2023'

Input:

```
SELECT COUNT(*)AS totalSales FROM coffeesales  
WHERE transaction_date ='1/1/2023'
```

Output :

totalSales
550

By Anand Sahu

# COFFEE SHOP SALES ANALYSIS



## Level -Basic

Q4 Determine the total sales amount for each product type in each store location.

Input:

```
SELECT product_type,store_location,ROUND(SUM(transaction_qty*unit_price),2)AS totalsales
FROM coffeesales GROUP BY product_type,store_location;
```

Output:

product_type	store_location	totalsales
Barista Espresso	Astoria	3808.25
Barista Espresso	Hell's Kitchen	4228.35
Barista Espresso	Lower Manhattan	4221.1
Biscotti	Astoria	906.5
Biscotti	Hell's Kitchen	896
Biscotti	Lower Manhattan	894.02
Black tea	Astoria	80.55
Black tea	Hell's Kitchen	187.95
Black tea	Lower Manhattan	107.4
Brewed Black tea	Astoria	2240.5
Brewed Black tea	Hell's Kitchen	2192.5

Q5 Find the difference in total sales amount between Coffee and Tea products

Input:

```
SELECT *FROM coffeesales limit 100;
SELECT product_type,store_location,ROUND(SUM(transaction_qty*unit_price),2)AS totalsales
FROM coffeesales GROUP BY product_type,store_location;
-- Find the difference in total sales amount between Coffee and Tea products
WITH cte AS (
SELECT product_category ,ROUND(SUM(transaction_qty*unit_price),4)AS total_sale
FROM coffeesales WHERE product_category IN('Coffee','Tea')
GROUP BY product_category)
SELECT
(SELECT total_sale FROM cte WHERE product_category='Coffee')-
(SELECT total_sale FROM cte WHERE product_category='Tea') AS diff
FROM cte LIMIT 1;
```

Output:

diff
9775.05

By Anand Sahu

# COFFEE SHOP SALES ANALYSIS



## Level Moderate

**Q1. Find the top 3 most sold products by total quantity.**

**Input:**

```
SELECT product_type,SUM(transaction_qty)AS total_quantity  
FROM coffeesales GROUP BY product_type  
ORDER BY total_quantity DESC LIMIT 3;
```

**Output**

store_location	total_transition
Astoria	6924
Hell's Kitchen	6823
Lower Manhattan	6429

**Q2 Calculate the total sales for each store and each product category.**

**Input:**

```
SELECT store_id,product_category,ROUND(SUM(transaction_qty*unit_price),2)AS total_sales  
FROM coffeesales GROUP BY store_id,product_category;
```

**Output:**

product_category	total_qunatity
Bakery	3113
Branded	100
Coffee	12152
Coffee beans	234
Drinking Chocolate	2378
Flavours	1218
Loose Tea	141
Packaged Chocolate	57
Tea	9572

**By Anand Sahu**

# COFFEE SHOP SALES ANALYSIS



## Level Moderate

Q3 List the store locations where the average transaction amount is greater than 4.

```
SELECT store_location,ROUND(AVG(transaction_qty*unit_price),2)AS tran_amount  
FROM coffeesales GROUP BY store_location HAVING tran_amount>4;
```

Output:

store_location	tran_amount
Astoria	4.58
Hell's Kitchen	4.7
Lower Manhattan	4.75

Q4.List the transactions where the transaction\_time is the same but different transaction\_id.

```
SELECT a.transaction_id ,a.transaction_time,b.transaction_id,b.transaction_time  
FROM coffeesales a  
JOIN coffeesales b ON a.transaction_time=b.transaction_time  
WHERE a.transaction_id!=b.transaction_id;
```

Output:

transaction_id	transaction_time	transaction_id	transaction_time
6	7:22:41	5	7:22:41
9207	7:22:41	5	7:22:41
5	7:22:41	6	7:22:41
9207	7:22:41	6	7:22:41
4448	7:33:34	8	7:33:34
9755	7:33:34	8	7:33:34
17	7:59:58	16	7:59:58
16	7:59:58	17	7:59:58
8644	8:00:18	18	8:00:18
12636	8:11:45	20	8:11:45
12637	8:11:45	20	8:11:45

By Anand Sahu

# COFFEE SHOP SALES ANALYSIS



## Level Moderate

Q5 Fetch total revenue percent of each product category Round 2 decimal

Input:

```
SELECT product_category,  
ROUND(SUM(transaction_qty*unit_price),2)/(SELECT  
SUM(ROUND(transaction_qty*unit_price,2)) FROM coffeesales)*100 revenue  
FROM coffeesales GROUP BY product_category;
```

Output:

product_category	revenue
Bakery	11.69222
Branded	2.003972
Coffee	38.85681
Coffee beans	5.561553
Drinking Chocolate	10.4347
Flavours	1.033159
Loose Tea	1.371873
Packaged Chocolate	0.553414
Tea	28.4923

Q6. Fetch the contribution Percentage of tea and coffee in total revenue .

Input:

```
WITH total_revenue_cte AS (  
    SELECT SUM(transaction_qty * unit_price) AS total_revenue  
    FROM coffeesales  
)  
cte2 AS (SELECT product_category,ROUND(total_revenue,2)AS golobal_total_rev,  
    ROUND(SUM(transaction_qty * unit_price), 2) AS total_revenue,  
    ROUND((SUM(transaction_qty * unit_price) / total_revenue_cte.total_revenue) * 100, 2)  
    AS '% contri.'  
FROM coffeesales, total_revenue_cte  
GROUP BY product_category, total_revenue_cte.total_revenue)  
SELECT *FROM cte2 WHERE product_category='Coffee' OR product_category='Tea';
```

Output:

product_category	golobal_total_rev	total_revenue	% contri
Coffee	94312.69	36646.9	38.86
Tea	94312.69	26871.85	28.49

By Anand Sahu

# COFFEE SHOP SALES ANALYSIS



## Level -Advance

Q1. Find the product with the highest average sales amount per date.

Input:

```
SELECT*FROM coffeesales ORDER BY transaction_date DESC LIMIT 100;
WITH cte AS (
SELECT product_type,transaction_date,ROUND(AVG(transaction_qty*unit_price),2) AS
avg_sale
FROM coffeesales GROUP BY transaction_date,product_type),
ranks AS(
SELECT product_type,transaction_date,avg_sale,RANK()OVER(PARTITION BY transaction_date
ORDER BY avg_sale DESC)AS rnk
FROM cte )
SELECT product_type,transaction_date,avg_sale
FROM ranks
WHERE rnk=1;
```

Output:

product_type	transaction_date	avg_sale
Barista Espresso	1/1/2023	5.97
Premium Beans	1/10/2023	32.38
Premium Beans	1/11/2023	45
Gourmet Beans	1/12/2023	21
Clothing	1/13/2023	28
Premium Beans	1/14/2023	32.38
Premium Beans	1/15/2023	36.58
Premium Beans	1/16/2023	45
Premium Beans	1/17/2023	202.5
Premium Beans	1/18/2023	45
Premium Beans	1/19/2023	45

By Anand Sahu

# COFFEE SHOP SALES ANALYSIS



## Level -Advance

Q2 Identify the top 3 store locations with the highest total sales amount, and for each of these locations, list the top-selling product category.

Input:

```
WITH cte AS (  
    SELECT store_location, product_category, transaction_qty * unit_price AS totalsale  
    FROM coffeesales  
)  
, store_totals AS (  
    SELECT store_location, SUM(totalsale) AS store_total_sale  
    FROM cte GROUP BY store_location  
)  
, ranked_stores AS (  
    SELECT store_location, store_total_sale, RANK() OVER (ORDER BY store_total_sale DESC) AS  
store_rank  
    FROM store_totals  
)  
, top_stores AS (  
    SELECT store_location, store_total_sale  
    FROM ranked_stores  
    WHERE store_rank <= 3  
)  
, category_totals AS (  
    SELECT store_location, product_category, SUM(totalsale) AS category_total_sale  
    FROM cte GROUP BY store_location, product_category  
)  
, ranked_categories AS (  
    SELECT store_location, product_category, category_total_sale,  
    RANK() OVER (PARTITION BY store_location ORDER BY category_total_sale DESC) AS  
category_rank  
    FROM category_totals)  
SELECT rc.store_location,  
    rc.product_category,  
    ROUND(rc.category_total_sale, 2) AS category_total_sale  
FROM ranked_categories rc  
JOIN top_stores ts ON rc.store_location = ts.store_location  
WHERE rc.category_rank = 1;
```

Output:

store_location	product_category	category_total_sale
Hell's Kitchen	Coffee	12307.25
Astoria	Coffee	12255.75
Lower Manhattan	Coffee	12083.9

By Anand Sahu



# COFFEE SHOP SALES ANALYSIS



## Level -Advance

Q3 Calculate the cumulative total sales amount for each product category over time, sorted by transaction date and time.

Input:

```
SELECT product_category,transaction_date,transaction_time
,SUM(transaction_qty*unit_price)As totalsales,
SUM(SUM(transaction_qty*unit_price))
OVER(PARTITION BY product_category ORDER BY transaction_date,transaction_time DESC
ROWS BETWEEN unbounded preceding AND CURRENT ROW )AS cummulative_sum
FROM
coffeesales
GROUP BY
product_category,
transaction_date
ORDER BY product_category, cummulative_sum;
```

Output:

product_category	transaction_date	transaction_time	totalsales	cummulative_sum
Bakery	1/1/2023	7:22:41	241.5	241.5
Bakery	1/10/2023	6:02:43	350.75	592.25
Bakery	1/11/2023	6:01:29	305	897.25
Bakery	1/12/2023	6:10:09	348	1245.25
Bakery	1/13/2023	7:01:27	358	1603.25
Bakery	1/14/2023	7:16:09	383.75	1987
Bakery	1/15/2023	6:14:18	352.5	2339.5
Bakery	1/16/2023	6:24:05	333.75	2673.25
Coffee	1/25/2023	6:04:25	1165.85	18371.65
Coffee	1/26/2023	6:02:06	1217.1	19588.75
Coffee	1/27/2023	7:01:52	1065.2	20653.95
Coffee	1/28/2023	7:02:31	973.25	21627.2
Coffee	1/29/2023	7:11:17	894.95	22522.15
Tea	1/10/2023	6:04:55	771.95	1583.05
Tea	1/11/2023	6:01:29	689.4	2272.45
Tea	1/12/2023	6:10:09	646.25	2918.7
Tea	1/13/2023	7:02:24	868.75	3787.45
Tea	1/14/2023	7:02:42	527.55	4315
Tea	1/15/2023	6:05:48	865.55	5180.55
Tea	1/16/2023	6:00:35	693.1	5873.65

By Anand Sahu

# COFFEE SHOP SALE ANALYSIS



## My SQL Project

Dataset Link:

<https://github.com/Jarvisggits47/SQL-coffee-sales-data-analysis/blob/main/Coffee-Shop-Sales.csv>

SQL Database Link:

[https://github.com/Jarvisggits47/SQL-coffee-sales-data-analysis/blob/main/coffee\\_sale\\_analysis.sql](https://github.com/Jarvisggits47/SQL-coffee-sales-data-analysis/blob/main/coffee_sale_analysis.sql)

GitHub Project Link:

<https://github.com/Jarvisggits47/SQL-coffee-sales-data-analysis>

Profile Links:

LinkedIn Profile: <https://www.linkedin.com/in/anand-sahu-49a337225/>

GitHub Profile: <https://github.com/Jarvisggits47>

Mail Id: [jarvisanad47@gmail.com](mailto:jarvisanad47@gmail.com)

