

SUPERMARKET SALES ANALYSIS



PRERNA PHULWAANI

Project Overview



This project focuses on analyzing supermarket sales data using SQL. It began with formulating specific queries to extract meaningful insights from a dataset,

The primary objective is to demonstrate how SQL can be utilized for data analysis and decision-making.

Each query is designed to answer practical business questions that can help a supermarket improve operations and maximize profit

Dataset Source: <https://www.kaggle.com/datasets/bravehart101/sample-supermarket-dataset>

LIST ALL RECORDS FROM CALIFORNIA WHERE THE CATEGORY IS 'FURNITURE'

```
SELECT *
FROM samplesuperstore
WHERE State = 'California'
AND Category = 'Furniture';
```

Result Grid													
	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit
▶	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Furniture	Furnishings	48.86	7	0	14.1694
	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Furniture	Tables	1706.184	9	0.2	85.3092
	Standard Class	Consumer	United States	Los Angeles	California	90004	West	Furniture	Furnishings	79.76	4	0	22.3328
	Second Class	Home Office	United States	Los Angeles	California	90004	West	Furniture	Chairs	81.424	2	0.2	-9.1602

GET ALL RECORDS FROM LOS ANGELES WHERE SEGMENT IS 'CONSUMER' AND CATEGORY IS 'TECHNOLOGY'.

```
SELECT *
FROM samplesuperstore
WHERE City = 'Los Angeles'
AND Segment = 'Consumer'
AND Category = 'Technology';
```

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit
▶	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Technology	Phones	907.152	6	0.2	90.7152
	Standard Class	Consumer	United States	Los Angeles	California	90032	West	Technology	Phones	911.424	4	0.2	68.3568
	Second Class	Consumer	United States	Los Angeles	California	90049	West	Technology	Accessories	90.57	3	0	11.7741
	Standard Class	Consumer	United States	Los Angeles	California	90004	West	Technology	Accessories	13.98	2	0	6.1512

FIND ALL ENTRIES FROM 'UNITED STATES' EXCLUDING CITIES 'LOS ANGELES', 'SAN FRANCISCO'.

```
SELECT *
FROM samplesuperstore
WHERE Country = 'United States'
AND City NOT IN ('Los Angeles', 'San Francisco');
```

Result Grid														
	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit	
▶	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.96	2	0	41.9136	
	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.94	3	0	219.582	
	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5	0.45	-383.031	
	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.368	2	0.2	2.5164	

DISPLAY THE TOTAL PROFIT FOR EACH SEGMENT IN THE "TECHNOLOGY" CATEGORY WHERE THE PROFIT EXCEEDS 5000.

```
SELECT  
    Segment,  
    SUM(Profit) AS total_profit  
FROM  
    samplesuperstore  
WHERE  
    Category = 'Technology'  
GROUP BY  
    Segment  
HAVING  
    total_profit > 5000;
```

Result Grid		
	Segment	total_profit
▶	Consumer	70797.80960000002
	Corporate	44166.99799999998
	Home Office	30490.14049999998



DISPLAY THE TOTAL SALES AND AVERAGE PROFIT FOR EACH CATEGORY WHERE THE QUANTITY SOLD IS GREATER THAN 3



SELECT

Category,

SUM(Sales) AS total_sales,

AVG(Profit) AS average_profit

FROM

samplesuperstore

GROUP BY

Category

HAVING

SUM(Quantity) > 3;

Result Grid | Filter Rows: Export:

	Category	total_sales	average_profit
▶	Furniture	741999.7952999998	8.699327109853842
	Office Supplies	719047.0320000029	20.327049585131117
	Technology	836154.0329999966	78.75200221981586

LIST THE STATES FROM THE 'SOUTH' REGION WHERE THE TOTAL DISCOUNT EXCEEDS 100 AND THE AVERAGE SALES IS BELOW 250

```
SELECT  
    State,  
    SUM(Discount) AS total_discount,  
    AVG(Sales) AS average_sales  
FROM  
    samplesuperstore  
WHERE  
    Region = 'South'  
GROUP BY  
    State  
HAVING  
    total_discount > 100 AND average_sales < 250;
```



A screenshot of a database query results window. The window has a title bar and a toolbar with buttons for 'Result Grid', 'Filter Rows', and 'Export'. The main area displays a table with three columns: 'State', 'total_discount', and 'average_sales'. There is one row of data: Florida, 114.65000000000074, and 233.6128146214099.

	State	total_discount	average_sales
▶	Florida	114.65000000000074	233.6128146214099

LIST THE TOP 3 POSTAL CODES BY TOTAL PROFIT IN THE 'EAST' REGION.

```
SELECT  
    `Postal Code`,  
    SUM(Profit) AS total_profit  
FROM  
    samplesuperstore  
WHERE  
    Region = 'East'  
GROUP BY  
    `Postal Code`  
ORDER BY  
    total_profit DESC  
LIMIT 3;
```

Result Grid | Filter Rows:

	Postal Code	total_profit
▶	10024	21653.724800000015
	10035	16533.866899999994
	10009	13697.0019

COMPARE AVERAGE PROFIT AND AVERAGE SALES BETWEEN FLORIDA AND KENTUCKY FOR THE 'FURNITURE' CATEGORY.

SELECT

State,
AVG(Profit) AS avg_profit,
AVG(Sales) AS avg_sales

FROM

samplesuperstore

WHERE

State **IN ('Florida', 'Kentucky')**
AND Category = 'Furniture'

GROUP BY

State;



A screenshot of a database query results window. The interface includes a toolbar with 'Result Grid', 'Filter Rows', and 'Export' buttons. The result grid displays the following data:

	State	avg_profit	avg_sales
▶	Kentucky	107.03310666666667	404.228
▶	Florida	-26.529184705882347	270.43574117647046

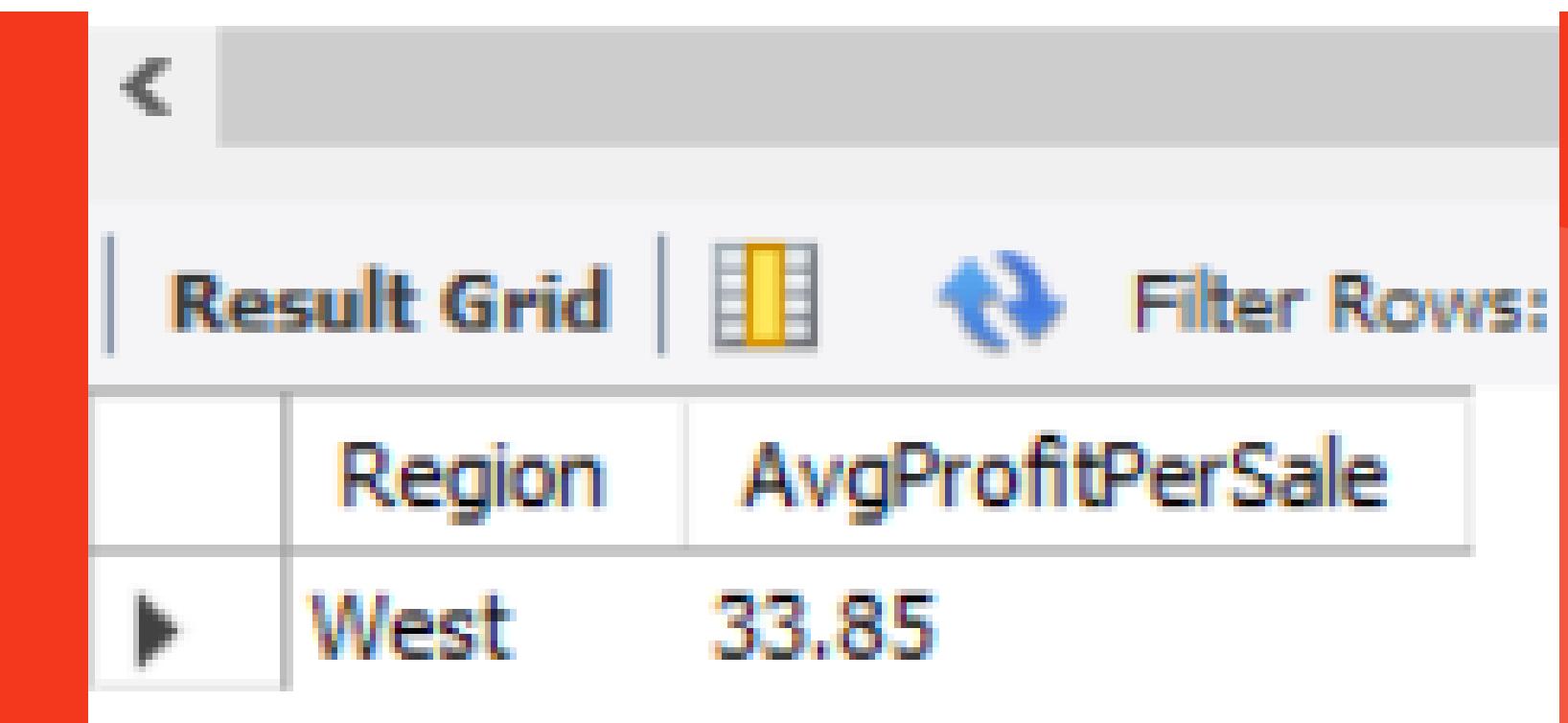
FOR EACH REGION AND CATEGORY, DISPLAY THE TOTAL SALES AND QUANTITY SOLD BUT ONLY WHERE TOTAL SALES EXCEED 1000

```
SELECT  
    Region,  
    Category,  
    SUM(Sales) AS total_sales,  
    SUM(Quantity) AS total_quantity  
FROM  
    samplesuperstore  
GROUP BY  
    Region, Category  
HAVING  
    total_sales > 1000;
```

	Region	Category	total_sales	total_quantity
▶	South	Furniture	117298.6840000001	1291
	West	Office Supplies	220853.24900000007	7235
	South	Office Supplies	125651.31299999992	3800
	West	Furniture	252612.7435000003	2696
	West	Technology	251991.8319999997	2335

FIND THE REGION WITH THE HIGHEST AVERAGE PROFIT PER SALE

```
SELECT Region, ROUND(SUM(Profit)/COUNT(*), 2) AS AvgProfitPerSale  
FROM samplesuperstore  
GROUP BY Region  
ORDER BY AvgProfitPerSale DESC  
LIMIT 1;
```



The screenshot shows a software interface for running SQL queries. At the top, there's a toolbar with a back arrow, a refresh icon, and a 'Result Grid' button. To the right of the grid button is a 'Filter Rows:' button with a magnifying glass icon. Below the toolbar is a table with two columns: 'Region' and 'AvgProfitPerSale'. The first row shows the header, and the second row shows the result for the West region.

Region	AvgProfitPerSale
West	33.85

COMPARE AVERAGE DISCOUNT AND PROFIT ACROSS SEGMENTS AND REGIONS

```
SELECT Segment, Region,  
       ROUND(AVG(Discount), 2) AS AvgDiscount,  
       ROUND(AVG(Profit), 2) AS AvgProfit  
FROM samplesuperstore  
GROUP BY Segment, Region  
ORDER BY Segment, AvgProfit DESC;
```

Result Grid | Filter Rows:

	Segment	Region	AvgDiscount	AvgProfit
▶	Consumer	West	0.11	34.36
	Consumer	South	0.14	32.12
	Consumer	East	0.15	28.04
	Consumer	Central	0.25	7.07
	Corporate	West	0.11	35.87

COMPARE TOTAL SALES BETWEEN CITIES IN CALIFORNIA AND FLORIDA, DISPLAYING CITY, STATE, AND TOTAL SALES.

```
SELECT  
    City,  
    State,  
    SUM(Sales) AS total_sales  
FROM  
    samplesuperstore  
WHERE  
    State IN ('California', 'Florida')  
GROUP BY  
    City, State  
ORDER BY  
    total_sales DESC;
```



Result Grid			
	City	State	total_sales
▶	Los Angeles	California	175851.341
	San Francisco	California	112669.09199999992
	San Diego	California	47521.02899999995
	Jacksonville	Florida	39133.32799999994
	Miami	Florida	8673.07449999995

CREATE A NEW COLUMN THAT CATEGORIZES ORDERS AS 'HIGH PROFIT', 'MODERATE PROFIT', OR 'LOSS'.

```
SELECT  
    City,  
    Profit,  
    CASE  
        WHEN Profit > 100 THEN 'High Profit'  
        WHEN Profit BETWEEN 0 AND 100 THEN 'Moderate Profit'  
        ELSE 'Loss'  
    END AS Profit_Category  
FROM samplesuperstore;
```

Result Grid | Filter Rows:

	City	Profit	Profit_Category
▶	Henderson	41.9136	Moderate Profit
	Henderson	219.582	High Profit
	Los Angeles	6.8714	Moderate Profit
	Fort Lauderdale	-383.031	Loss
	Fort Lauderdale	2.5164	Moderate Profit

SHOW HOW MANY CITIES IN EACH STATE HAD OVERALL NEGATIVE PROFIT.

```
SELECT State, COUNT(*) AS cities_with_loss
FROM (
    SELECT State, City, SUM(Profit) AS total_profit
    FROM samplesuperstore
    GROUP BY State, City
    HAVING total_profit < 0
) AS loss_cities
GROUP BY State;
```

Result Grid | Filter Rows:

	State	cities_with_loss
▶	Florida	13
	North Carolina	8
	Pennsylvania	7
	Texas	36
	Illinois	25

LIST PAIRS OF CITIES IN THE SAME STATE WHERE ONE CITY BELONGS TO THE 'CONSUMER' SEGMENT AND THE OTHER TO THE 'HOME OFFICE' SEGMENT

```
SELECT  
    A.City AS city_one,  
    B.City AS city_two,  
    A.State  
FROM  
    samplesuperstore A  
JOIN  
    samplesuperstore B  
ON  
    A.State = B.State  
    AND A.City < B.City  
    AND A.Segment = 'Consumer'  
    AND B.Segment = 'Home Office';
```

	city_one	city_two	State
▶	Carrollton	Fort Worth	Texas
	Carrollton	Fort Worth	Texas
	Dallas	Fort Worth	Texas
	Beaumont	Fort Worth	Texas
	Beaumont	Fort Worth	Texas

FIND PAIRS OF CITIES WHERE ONE CITY HAS PROFIT > 500 AND THE OTHER < 0, BUT BOTH ARE FROM THE SAME STATE AND CATEGORY.

SELECT

A.City AS profitable_city,

B.City AS loss_city,

A.State,

A.Category

FROM

samplesuperstore A

JOIN

samplesuperstore B

ON

A.State = B.State

AND A.Category = B.Category

AND A.City != B.City

GROUP BY

A.City, B.City, A.State, A.Category

HAVING

SUM(A.Profit) > 500 AND SUM(B.Profit) < 0;

Result Grid | Filter Rows: Export:

	profitable_city	loss_city	State	Category
▶	Hialeah	Fort Lauderdale	Florida	Furniture
	Pasadena	Fort Worth	Texas	Office Supplies
	Harlingen	Fort Worth	Texas	Office Supplies
	Brownsville	Fort Worth	Texas	Office Supplies
	Chester	Philadelphia	Pennsylvania	Furniture

CREATE A SELF-JOIN TO LIST PAIR OF CITIES IN THE SAME STATE AND CATEGORY BUT WITH DIFFERENT SEGMENTS

SELECT

```
A.City AS city_one,  
B.City AS city_two,  
A.State,  
A.Category,  
A.Segment AS segment_one,  
B.Segment AS segment_two
```

FROM

```
samplesuperstore A
```

JOIN

```
samplesuperstore B
```

ON

```
A.State = B.State  
AND A.Category = B.Category  
AND A.City < B.City  
AND A.Segment != B.Segment;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: F

	city_one	city_two	State	Category	segment_one	segment_two
▶	Bowling Green	Henderson	Kentucky	Furniture	Corporate	Consumer
	Bowling Green	Henderson	Kentucky	Furniture	Corporate	Consumer
	Anaheim	Los Angeles	California	Office Supplies	Consumer	Corporate
	Anaheim	Los Angeles	California	Office Supplies	Consumer	Corporate
	Escondido	Los Angeles	California	Office Supplies	Home Office	Corporate

FIND COMBINATIONS OF CITIES IN THE SAME REGION AND CATEGORY BUT DIFFERENT STATES.

```
SELECT
    A.City AS city_one,
    B.City AS city_two,
    A.Region,
    A.Category
FROM
    samplesuperstore A
JOIN
    samplesuperstore B
ON
    A.Region = B.Region
    AND A.Category = B.Category
    AND A.State != B.State
    AND A.City < B.City;
```

Result Grid | Filter Rows:

	city_one	city_two	Region	Category
▶	Fayetteville	Henderson	South	Furniture
	Greensboro	Henderson	South	Furniture
	Columbia	Henderson	South	Furniture
	Deltona	Henderson	South	Furniture
	Burlington	Henderson	South	Furniture

LIST THE TOP 5 STATES WITH THE HIGHEST AVERAGE DISCOUNT FOR THE 'CORPORATE' SEGMENT.

```
SELECT  
    State,  
    AVG(Discount) AS avg_discount  
FROM  
    samplesuperstore  
WHERE  
    Segment = 'Corporate'  
GROUP BY  
    State  
ORDER BY  
    avg_discount DESC  
LIMIT 5;
```

	State	avg_discount
▶	Illinois	0.4008333333333336
	Texas	0.3605769230769235
	Pennsylvania	0.3362244897959191
	Ohio	0.32845528455284545
	Tennessee	0.3214285714285712

DISPLAY TOTAL SALES AND TOTAL QUANTITY FOR EACH REGION AND SEGMENT COMBINATION.

SELECT

Region,

Segment,

SUM(Sales) **AS** total_sales,

SUM(Quantity) **AS** total_quantity

FROM

samplesuperstore

GROUP BY

Region, Segment;

	Region	Segment	total_sales	total_quantity
▶	South	Consumer	195580.97100000017	3178
	West	Corporate	225855.27449999977	3630
	West	Consumer	362880.7730000003	6477
	Central	Home Office	91212.64399999994	1657

FIND THE CITIES WHERE TOTAL PROFIT IS NEGATIVE BUT TOTAL SALES ARE ABOVE 1000.

```
SELECT  
    City,  
    SUM(Profit) AS total_profit,  
    SUM(Sales) AS total_sales  
FROM  
    samplesuperstore  
GROUP BY  
    City  
HAVING  
    total_profit < 0  
    AND total_sales > 1000;
```

The screenshot shows a database query results grid with the following columns: City, total_profit, and total_sales. The grid displays five rows of data corresponding to the cities listed in the query results.

	City	total_profit	total_sales
▶	Fort Lauderdale	-161.3551	4929.532500000001
	Concord	-718.6389000000001	10542.402000000002
	Philadelphia	-13837.767400000012	109077.01300000008
	Houston	-10153.548499999997	64504.76039999994
	Richardson	-24.4893	1288.4640000000002

FIND COMBINATIONS OF ROWS FROM THE SAME STATE BUT DIFFERENT SEGMENTS, SHOWING BOTH CITIES, CATEGORY, AND THE COMMON STATE.

```
SELECT  
    A.City AS city_one,  
    B.City AS city_two,  
    A.Category,  
    A.State  
FROM  
    samplesuperstore A  
JOIN  
    samplesuperstore B  
ON  
    A.State = B.State  
    AND A.Segment != B.Segment  
    AND A.City < B.City  
    AND A.Category = B.Category;
```

	city_one	city_two	Category	State
▶	Bowling Green	Henderson	Furniture	Kentucky
	Bowling Green	Henderson	Furniture	Kentucky
	Anaheim	Los Angeles	Office Supplies	California
	Anaheim	Los Angeles	Office Supplies	California
	Escondido	Los Angeles	Office Supplies	California

LIST STATES WHERE TOTAL SALES ARE ABOVE THE AVERAGE TOTAL STATE SALES.

```
SELECT State, SUM(Sales) AS total_sales
FROM samplesuperstore
GROUP BY State
HAVING total_sales > (
    SELECT AVG(state_sales)
    FROM (
        SELECT State, SUM(Sales) AS state_sales
        FROM samplesuperstore
        GROUP BY State
    ) AS state_totals
);
```

Result Grid | Filter Rows:

	State	total_sales
▶	California	457687.631500001
	Florida	89473.708
	North Carolina	55603.16399999997
	Washington	138641.26999999993
	Texas	170188.04580000002

IDENTIFY CITIES WITH HIGHER THAN AVERAGE SALES AND LOWER THAN AVERAGE QUANTITY

```
SELECT City,  
       ROUND(AVG(Sales), 2) AS AvgSales,  
       ROUND(AVG(Quantity), 2) AS AvgQty  
FROM samplesuperstore  
GROUP BY City  
HAVING AvgSales > (  
    SELECT AVG(Sales) FROM samplesuperstore  
)  
AND AvgQty < (  
    SELECT AVG(Quantity) FROM samplesuperstore  
)  
ORDER BY AvgSales DESC;
```

	City	AvgSales	AvgQty
▶	Independence	1208.68	3.00
	Lafayette	807.62	3.35
	Noblesville	772.8	3.75
	Kissimmee	751.98	2.00
	Minneapolis	733.5	3.48

CLASSIFY EACH STATE BASED ON TOTAL SALES AND TOTAL PROFIT THRESHOLDS.

SELECT

```
State,  
ROUND(SUM(Sales), 2) AS total_sales,  
ROUND(SUM(Profit), 2) AS total_profit,
```

CASE

```
WHEN SUM(Sales) > 15000 AND SUM(Profit) > 5000 THEN 'High Sales & High Profit'  
WHEN SUM(Sales) > 15000 AND SUM(Profit) <= 5000 THEN 'High Sales, Low Profit'  
WHEN SUM(Sales) <= 15000 AND SUM(Profit) > 5000 THEN 'Low Sales, High Profit'  
ELSE 'Low Sales & Low Profit'
```

```
END AS performance_segment
```

```
FROM samplesuperstore
```

```
GROUP BY State
```

```
ORDER BY total_sales DESC;
```

Result Grid | Filter Rows: Export:

	State	total_sales	total_profit	performance_segment
▶	California	457687.63	76381.39	High Sales & High Profit
	New York	310876.27	74038.55	High Sales & High Profit
	Texas	170188.05	-25729.36	High Sales, Low Profit
	Washington	138641.27	33402.65	High Sales & High Profit
	Pennsylvania	116511.91	-15559.96	High Sales, Low Profit



Project summary:

- EXECUTED AND ANALYZED 25 SQL QUERIES ON THE SAMPLESUPERSTORE DATASET.
 - FILTERING WITH WHERE
 - GROUPING DATA USING GROUP BY AND HAVING
 - SORTING AND LIMITING RESULTS
 - APPLIED THE SELF JOIN TECHNIQUE TO IDENTIFY PATTERNS WHICH HELPED REINFORCE HOW RELATIONAL LOGIC CAN BE APPLIED EVEN WITHIN A SINGLE TABLE WHEN NEEDED.
- USING AGGREGATE FUNCTIONS LIKE SUM(), AVG(), COUNT(), ROUND()

Final thoughts:

This project helped me gain practical experience in data analysis using SQL. It showed how structured queries can turn raw sales data into clear, decision-ready insights. It also improved my ability to write clean, efficient queries for both simple lookups and deeper business analysis. Every query provided a new perspective on how structured data can reveal business trends, such as high-profit cities, underperforming states, or customer behavior across regions.





Thank You