

Internship Program: Soulvibe.Tech

PROJECT

FARMER AND MARKET

DATASET ANALYSIS WITH SQL

Presented By: Siddesh Keshav Vaishnav
Batch Name: SVT/DAINT/2025/07/B11

01

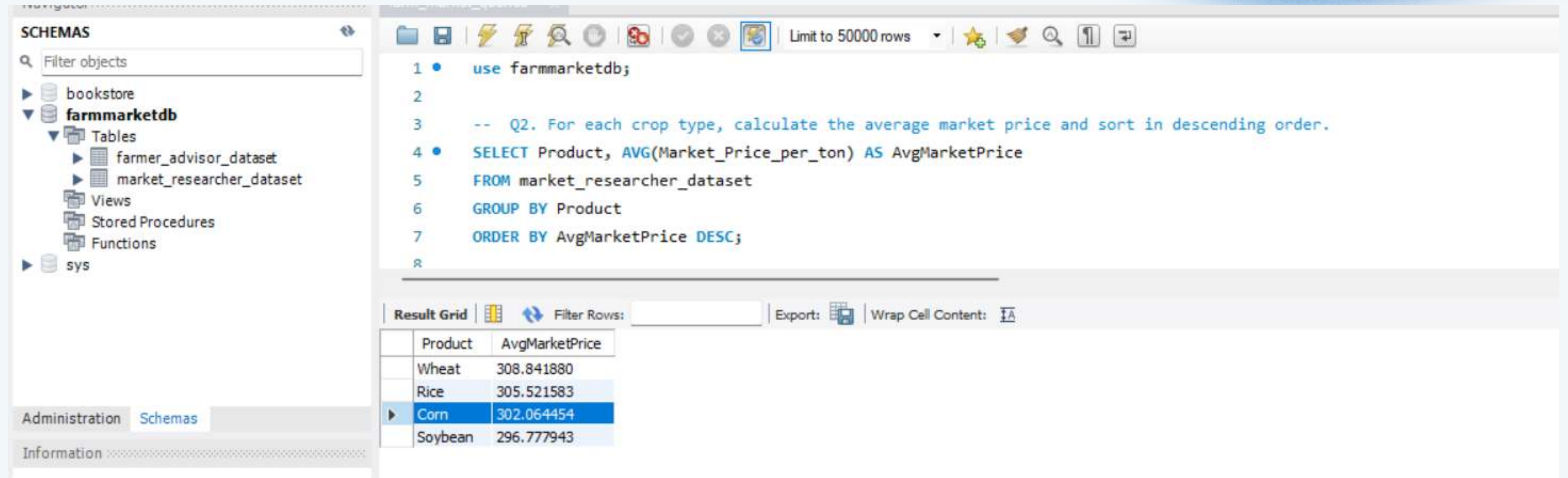
INTRODUCTION

OPTIMIZING AGRICULTURAL DECISION-MAKING WITH STRUCTURED SQL ANALYSIS

In this project, I aimed to analyze and bridge the gap between on-ground farming practices and market-level trends using structured SQL queries. By working with two datasets—`farmer_advisor_dataset` and `market_researcher_dataset`—I explored a variety of agricultural and economic patterns, such as crop yield distribution, advisor influence, market price fluctuations, and growth classifications. Each query was designed to extract meaningful insights that could assist farmers, advisors, and policymakers in making informed decisions.

Throughout the process, I focused on maintaining case sensitivity and data integrity while identifying correlations and anomalies across both datasets. My goal was to uncover hidden trends that could drive better crop planning, advisory strategies, and pricing awareness. This project reflects my interest in applying data analytics to solve real-world challenges in the agricultural domain.

Q2. FOR EACH CROP TYPE, CALCULATE THE AVERAGE MARKET PRICE AND SORT IN DESCENDING ORDER.



The screenshot shows a database management interface. On the left, the 'SCHEMAS' pane displays a tree view with 'bookstore' and 'farmmarketdb' (expanded to show 'Tables' and 'Views'). The main area contains a SQL query editor with the following code:

```
1 • use farmmarketdb;
2
3 -- Q2. For each crop type, calculate the average market price and sort in descending order.
4 • SELECT Product, AVG(Market_Price_per_ton) AS AvgMarketPrice
5 FROM market_researcher_dataset
6 GROUP BY Product
7 ORDER BY AvgMarketPrice DESC;
8
```

Below the query editor, the 'Result Grid' tab is active, displaying a table with the following data:

Product	AvgMarketPrice
Wheat	308.841880
Rice	305.521583
Corn	302.064454
Soybean	296.777943

Q5. .LIST ALL CROPS WHERE THE MAX AND MIN MARKET PRICES (FROM MARKETRESEARCHER) DIFFER BY MORE THAN ₹10 PER KG.

Navigator

SCHEMAS

Filter objects

bookstore

farmmarketdb

Tables

farmer_advisor_dataset

market_researcher_dataset

Views

Stored Procedures

Functions

sys

Administration

Schemas

Information

farm_market_queries

Limit to 50000 rows

7

ORDER BY AvgMarketPrice DESC;

8

*/

9

-- Q5. List all crops where the max and min market prices differ by more than ₹10 per ton.

10

SELECT Product

11

FROM market_researcher_dataset

12

GROUP BY Product

13

HAVING MAX(Market_Price_per_ton) - MIN(Market_Price_per_ton) > 10;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Product
Rice
Wheat
Soybean
Corn

04

Q7. RANK CROPS BY PROFIT PER UNIT (ASSUME: MARKET PRICE - AVERAGE COST FROM FARMERADVISOR) USING RANK().

The screenshot displays a database management interface. On the left, the 'Navigator' pane shows the 'farmmarketdb' schema with tables 'farmer_advisor_dataset' and 'market_researcher_dataset'. The main editor window, titled 'farm_market_queries*', contains a SQL query. The query calculates the profit per unit for different crops by subtracting the average fertilizer usage cost from the market price, and then ranks the crops by profit. The results are shown in a table with columns 'Crop_Type', 'ProfitPerUnit', and 'RankByProfit'.

```
-- Q7. Rank crops by profit per unit (assume: market price - average fertilizer usage as cost) using RANK().
SELECT
    f.Crop_Type,
    AVG(m.Market_Price_per_ton - f.Fertilizer_Usage_kg) AS ProfitPerUnit,
    RANK() OVER (ORDER BY AVG(m.Market_Price_per_ton - f.Fertilizer_Usage_kg) DESC) AS RankByProfit
FROM farmer_advisor_dataset f
JOIN market_researcher_dataset m
    ON f.Crop_Type = m.Product
GROUP BY f.Crop_Type;
```

Crop_Type	ProfitPerUnit	RankByProfit
Wheat	183.359358	1
Rice	180.296262	2
Corn	176.093924	3
Soybean	172.570491	4

Q8. IDENTIFY LOCATIONS WHERE THE CURRENT MARKET PRICE OF A CROP IS MORE THAN 20% ABOVE THE AVERAGE PRICE OF THAT CROP ACROSS ALL LOCATIONS.

Navigator

SCHEMAS

Filter objects

bookstore

farmmarketdb

Tables

farmer_advisor_dataset

market_researcher_dataset

Views

Stored Procedures

Functions

sys

Administration

Schemas

Information

No object selected

farm_market_queries*

Limit to 50000 rows

28

-- Q8. Identify crops where the current market price is more than 20% above the average across all mark

29

• SELECT Product, Market_Price_per_ton

30

FROM market_researcher_dataset

31

WHERE Market_Price_per_ton > 1.2 * (

32

SELECT AVG(Market_Price_per_ton)

33

FROM market_researcher_dataset AS sub

34

WHERE sub.Product = market_researcher_dataset.Product

35

);

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Market_Price_per_ton
▶	Rice	420.53
	Wheat	457.26
	Corn	389.65
	Soybean	447.41
	Soybean	481.95
	Corn	400.64
	Soybean	468.09
	Soybean	404.38
	Wheat	444.21
	Wheat	431.80
	Soybean	443.26
	Rice	426.99
	Corn	419.88
	Rice	464.68
	Wheat	370.65
	Wheat	370.82
	Rice	424.55
	Soybean	356.23
	Soybean	376.65
	Rice	405.83
	Corn	377.87
	Wheat	433.82

Q12. CREATE A NEW COLUMN THAT CLASSIFIES CROP GROWTH RATE AS:

- LOW (<20%)
- MEDIUM (20–50%)
- HIGH (>50%)

COUNT THE NUMBER OF CROPS IN EACH CATEGORY.

The screenshot shows a SQL query editor window titled "farm_market_queries". The query is as follows:

```
-- Q12. Create a new column that classifies crop yield as Low, Medium, or High, and count the number of crops in each category.
SELECT
CASE
    WHEN Crop_Yield_ton < 2 THEN 'Low'
    WHEN Crop_Yield_ton BETWEEN 2 AND 5 THEN 'Medium'
    ELSE 'High'
END AS YieldCategory,
COUNT(*) AS CropCount
FROM farmer_advisor_dataset
GROUP BY YieldCategory;
```

Below the query editor, the "Result Grid" is displayed, showing the results of the query:

	YieldCategory	CropCount
▶	Low	1144
	Medium	3331
	High	5525



farm_market_queries

```
-- Q15. Identify if any farmer has duplicate crop entries (by Crop_Type).
SELECT Crop_Type, COUNT(*) AS NumEntries
FROM farmer_advisor_dataset
GROUP BY Crop_Type
HAVING COUNT(*) > 1;
```

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

Crop_Type	NumEntries
Wheat	2522
Soybean	2559
Corn	2455
Rice	2464

Q16. LIST ALL CROPS GROWN BY FARMERS THAT ARE NOT LISTED IN THE MARKETRESEARCHER TABLE.

The screenshot shows a SQL query editor window titled "farm_market_queries*" with a toolbar at the top containing icons for file operations, execution, and settings. The query text is as follows:

```
57  -- Q16. List all crops grown by farmers that are not listed in the market dataset.
58  • SELECT DISTINCT f.Crop_Type
59     FROM farmer_advisor_dataset f
60     LEFT JOIN market_researcher_dataset m
61         ON f.Crop_Type = m.Product
62     WHERE m.Product IS NULL;
63
```

Below the query editor, there is a "Result Grid" section with a toolbar for filtering, exporting, and wrapping cell content. The first column header is "Crop_Type".

Crop_Type

09

Q19.LIST ALL ADVISORS ASSOCIATED WITH MORE THAN 5 DISTINCT CROP TYPES.

The screenshot shows a database query editor interface. On the left, a sidebar displays the database schema for 'farmmarketdb', including tables like 'farmer_advisor_dataset' and 'market_researcher_dataset'. The main window, titled 'farm_market_queries*', contains a SQL query. The query is as follows:

```
62 WHERE m.Product IS NULL; */
63
64 -- Q19. List all distinct crop types (as we don't have Advisor info).
65 • SELECT Crop_Type, COUNT(*) AS Frequency
66 FROM farmer_advisor_dataset
67 GROUP BY Crop_Type
68 HAVING COUNT(DISTINCT Crop_Type) > 5;
```

Below the query editor, there is a 'Result Grid' section with a toolbar containing icons for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. The 'Result Grid' section shows a table with two columns: 'Crop_Type' and 'Frequency'.

10

CONCLUSION

This project allowed me to deepen my understanding of SQL-based data analysis in the context of agriculture and market dynamics. By querying and interpreting two interconnected datasets, I was able to generate practical insights that highlight the value of data-driven decision-making for both farmers and market researchers. The process helped me recognize the importance of precision, especially in case-sensitive environments, and improved my ability to ask relevant analytical questions.

I believe this exercise not only sharpened my technical skills but also reinforced the potential of structured data analysis in addressing real-world problems. Moving forward, I'm motivated to explore more advanced tools and larger datasets to continue building impactful solutions at the intersection of agriculture, economics, and data science.



THANK YOU