Dataset Link

<https://www.kaggle.com/datasets/wyattowalsh/basketball>

Best Resource for Data Visualizations

<https://www.data-to-viz.com/>

Essential SQL Session Notes

Session Resources:

* <https://www.w3schools.com/sql/>
* <https://www.metabase.com/learn/databases/data-mart-data-warehouse-data-lake>
* <https://learnsql.com/blog/sql-order-of-operations/>
* <https://www.geeksforgeeks.org/sql-ddl-dql-dml-dcl-tcl-commands/>
* <https://learntocodewith.me/posts/sql-guide/>

-- Find the least 5 products by profit

Select top 5 product\_name, Round(sum(profit), 2) as totalProfit

from superstore

group by product\_name

order by totalProfit asc; -- Items that have a negative might want to be phased out and replaced with other products

-- Let's see the top 5 products' profit, sales, category, and quantity

Select product\_name, category, sum(quantity) as totalQTY,

round(sum(sales), 2) as totalSales, round(sum(profit), 2) as totalProfit

from superstore

group by product\_name, category

order by totalProfit desc;

-- Find the top quantity for each category

Select top 5 category, sum(quantity) as totalQTY

from superstore

group by category

order by totalQTY desc;

-- Find the top profit for each category

Select top 5 category, round(sum(profit), 2) as totalProfit

from superstore

group by category

order by totalProfit desc; -- Tech sold less quantity and made the most in profit, suggesting tech had higher-priced or high-value products versus the others

-- The most sold subcategory

select top 1 sub\_category, sum(quantity) as totalQTY

from superstore

group by sub\_category

order by totalQTY desc; -- Binders is the most sold subcategory

-- Most profit by subcategory

Select top 1 sub\_category, round(sum(profit), 2) as totalProfit

from superstore

group by sub\_category

order by totalProfit desc; -- Copiers is the most profitable subcategory

-- See how many products the store has to buy from

select count(distinct product\_id)

from superstore; -- 1862 different products a customer can buy

select customer\_name, state, country, region

from superstore

where category = 'Furniture';

-- Get top customers by category

Select customer\_name, category, round(sum(sales), 2) as totalSales

from superstore

group by customer\_name, category

order by totalSales desc;

-- See how many customers there are

select count(distinct customer\_id)

from superstore; -- Almost 800 customers total

-- Count of countries

select count(distinct country)

from superstore; -- Only the US is in the data set

select count(distinct state)

from superstore; -- 49 states

select count(distinct city)

from superstore;

-- See what columns and values we have

select \*

from superstore;

-- Which region has the most discounts and in what category

select region, category, round(sum(discount), 2) as totalDiscount

from superstore

group by region, category; -- The west seems to have the highest sum of discounts

-- Which region has discounts more often

Select region, count(discount) as discountCount

from superstore

group by region

order by discountCount desc; -- The western region tends to have discounts more often, yet they have lower discount values than the western region.

-- When using between we use and with between. When using SQL with a date type we use quotes. Example: '11-1-2016'.

-- If you have and or - or runs first. Using () you can control the order. This is like formulas in Salesforce. The category must be furniture and state CA or the date between those dates.

select customer\_name, region, country, state, category

from superstore

where (category = 'Furniture' and state = 'California') or ship\_date between '11/1/2016' and '11/30/2016';

-- If you want both to be true we change or make both conditions a must

select customer\_name, region, country, state, category

from superstore

where (category = 'Furniture' and state = 'California') and (ship\_date between '11/1/2016' and '11/30/2016');

-- Not != <>

-- Not with IN function - not in

-- Count. If we want to exclude nulls, count(\*). If we want to exclude nulls in a specific column, count(column). If we want distinct and no nulls, count(distinct city).

select count(order\_id) as totalOrders, state, region, category

from superstore

group by region, state, category

order by totalOrders desc;

VARCHAR Vs. CHAR

CHAR is fixed-length, and VARCHAR is dynamic.

CHAR(10) and VARCHAR(10): VARCHAR is very similar to the TEXT data type in Salesforce, where we can define a maximum length.

With VARCHAR, we can set a limit, and if the limit is not used (e.g., using 5 characters out of a maximum of 10), the system will not use all the memory storage. However, with CHAR, if we set a limit to 10, the system will always use the fixed 10 characters in storage, regardless of the actual length of the input.

If you know a fixed number of characters will be needed, such as a phone number, use CHAR. If you are undecided about characters, use VARCHAR for a dynamic string.

In summary, choose CHAR for fixed-length strings to optimize performance and VARCHAR for variable-length strings to optimize storage efficiency.

Business Thinking in Data Analysis Session Notes

“Business is pretty simple. The entire point of a business is to bring value to their shareholders.”

Know the financial terms for the business:

* Revenue
* Profit
* Operating Expenses: What does it take to keep the lights on (salaries/wages, rent, utilities,
* Non-operating expenses (legal fees, recruiters)
* Capital expenses

“What insights can you give me that will be driving these things?”

90% of your requests are related to something here.

Pareto principle: 80% of X is caused by 20% of X. Or 20% of X is caused by 80% of Y

Failing to plan is planning for failure.

Business Plan 101: A roadmap outlining a company’s goals, strategies and financial projections. (Exec summary, market analysis, marketing & sales plan, operational plan, financial projections).

Understanding the business plan will determine your role as a data analyst.

Strategic Thinking: Earnings Announcements are quarterly or annual reports publicly traded companies release detailing their financial performance.

This is one way to get data.

Compare companies and make predictions on future performance

Business is simple, data makes it complicated.

Analysis Frameworks:

1. Competitors: Market Analysis (a process of assessing the attractiveness and dynamics of a market).
2. SWOT: Strengths, Weaknesses, Opportunities, and Threats
3. Porter’s Five Forces: <https://www.investopedia.com/terms/p/porter.asp>
4. Customer Journey Map

Data by itself is nothing. Data put into context is a treasure.

Data = Key Business Concepts

Business Plan: Data regarding revenue, profit, costs, AND THEIR DRIVES (most important).

Inputs and outputs

Understand what CHANGED - what led to the changes? How did those outputs change?

Earnings Announcements

Understand the finances and how they related to the data (similar to a business plan)

The difference is strategic questions (guidance) —> understand how that relates to your business unit/function

Competitive Intelligence and business process refinement: Both use the same mindset — what are your drivers? How did they change? What happened? Do we continue or do we stop? What are our competitors doing?

Document it or It Didn’t Happen.

Documentation:

* NotebookLM
* GitHub

Notebook Etiquette as a Data Professional

* Purpose of Notebooks:

Notebooks are tools to share code, methodology, and information.

Well-organized notebooks lead to efficiency, reproducibility, and easier debugging.

* Structural Organization:

Clear sections with headings.

Markdown for Explanations - explain your code, assumptions, & results.

Code Organization

Use Meaningful Variable Names - use descriptive names that reflect the data’s content. Use something common in your industry.

Comments - explain complex code or assumptions

Modular Code - break down code into reusable functions

Formatting Tips - add Line breaks

* Visualization:

Use clear, informative plots and graphs

Label axes, titles, and legends appropriately

Example: (Show a well-formatted plot using Matplotlib or Seaborn).

You don’t need the whole rainbow. Consider coolers.co

* Advanced tips:

1. Version Control (Git):

Track changes, collaborate, and revert to previous versions.

Integrate with GitHub or Bitbucket for easy sharing.

1. Parameterization:

Create variables for storing things like file paths or model hyperparameters to make your notebook easily adaptable.

1. Table of Contents:

Colab provides an automatic table of contents for easy navigation

1. Documentation:

Document notebook in a way that the other person with little to no knowledge can also understand what’s going on. Be very particular about this and it can do wonders for your life.

Write your notebook like you’re Dory from Finding Nemo.

Use what is needed.

You don’t need to overdo your notebooks.

A good notebook looks like a textbook.