**End-to-End Coffee Sales Analysis Using Snowflake, AWS, SQL, and Power BI**

**Project Overview**

This project demonstrates an end-to-end process for analyzing coffee sales data, starting from data integration via AWS, transforming and analyzing data in Snowflake, and visualizing the insights using Power BI.

**Table of Contents**

[AWS Setup](#aws-setup)

[Snowflake Setup and Data Loading](#snowflake-setup-and-data-loading)

[Data Transformation in Snowflake](#data-transformation-in-snowflake)

[Connecting Snowflake to Power BI](#connecting-snowflake-to-power-bi)

[Creating Power BI Dashboards](#creating-power-bi-dashboards)

[Conclusion](#conclusion)

[Appendix: Code Listings](#appendix-code-listings)

**AWS Setup**

**Step 1: Creating Roles and Policies**

* **IAM Policy Creation:** We created an IAM policy using the below code and attached role within specific to manage access securely. And even used the storage integration IAM aws user profile id and access code copied to trust relationships of IAM role

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": "s3:\*",

"Resource": "arn:aws:s3:::coffeesalesbucket/\*"

}

]

}

**Step 2: Uploading Data to S3**

* **Data Insertion:** Uploaded transactional data into Amazon S3 buckets.

aws s3 cp coffee\_sales\_data.csv s3://coffeesalesbucket/coffee\_sales\_folder/

**Snowflake Setup and Data Loading**

**Step 3: Creating Warehouse and Database in Snowflake**

sql code

CREATE OR REPLACE WAREHOUSE my\_warehouse

WITH WAREHOUSE\_SIZE = 'SMALL'

AUTO\_SUSPEND = 300

AUTO\_RESUME = TRUE

INITIALLY\_SUSPENDED = TRUE;

CREATE DATABASE COFFEE\_SALES\_DB;

USE DATABASE COFFEE\_SALES\_DB;

CREATE SCHEMA COFFEE\_SALES\_SCHEMA;

USE SCHEMA COFFEE\_SALES\_SCHEMA;

**Step 4: Creating Tables and Loading Data**

Sql code

CREATE OR REPLACE TABLE COFFEE\_SHOP\_SALES(

transact\_id INT NOT NULL,

transaction\_date VARCHAR(15),

transaction\_time VARCHAR(15),

transaction\_qty INT,

store\_id INT,

store\_location VARCHAR(20),

product\_id INT,

Unit\_Price FLOAT,

product\_category VARCHAR(30),

product\_type VARCHAR(30),

product\_detail VARCHAR(30)

);

---- CREATING FILE FORMAT -----

CREATE OR REPLACE FILE FORMAT CSV\_FORMAT1

TYPE = 'csv'

COMPRESSION = 'none'

FIELD\_DELIMITER = ','

SKIP\_HEADER = 1;

### Step 5: Creating Storage Integration and External Stage

Configure Snowflake to access data stored in your S3 bucket:

CREATE OR REPLACE STORAGE INTEGRATION s3\_integration

TYPE = EXTERNAL\_STAGE

STORAGE\_PROVIDER = S3

ENABLED = TRUE

STORAGE\_AWS\_ROLE\_ARN = 'arn:aws:iam::123456789012:role/coffeesalesrole'

STORAGE\_ALLOWED\_LOCATIONS = ('s3://coffeesalesbucket');

------- CREATING STAGE--------

CREATE OR REPLACE STAGE coffee\_sales\_schema.s3\_stage

STORAGE\_INTEGRATION = s3\_integration

URL = 's3://coffeesalesbucket/coffee\_sales\_folder'

FILE\_FORMAT = CSV\_FORMAT1;

### Step 6: Creating Pipe to Automate Data Loading

Automate data loading from the S3 bucket to your Snowflake table:

CREATE OR REPLACE PIPE coffee\_sales\_schema.coffee\_sales\_pipe

AUTO\_INGEST = TRUE

AS

COPY INTO coffee\_sales\_schema.coffee\_shop\_sales

FROM @coffee\_sales\_schema.s3\_stage

FILE\_FORMAT = CSV\_FORMAT1;

ALTER PIPE coffee\_sales\_schema.coffee\_sales\_pipe REFRESH;

**Data Transformation in Snowflake**

**Step 7: Running Transformations**

Sql code

Calculating Total Sales

CREATE OR REPLACE TABLE COFFEE\_SHOP\_SALES\_TRANSFORMED AS

SELECT

transact\_id,

transaction\_date,

transaction\_time,

transaction\_qty,

store\_id,

store\_location,

product\_id,

Unit\_Price,

product\_category,

product\_type,

product\_detail,

transaction\_qty \* Unit\_Price AS total\_sales

FROM COFFEE\_SHOP\_SALES;

------DATA TRANSFORMATION STEPS----------

----CONVERT DATE (transaction\_date) COLUMN TO PROPER DATE FORMAT---

UPDATE COFFEE\_SHOP\_SALES

SET transaction\_date = TO\_DATE(transaction\_date, 'DD-MM-YYYY');

---ALTER DATE (transaction\_date) COLUMN TO DATE DATA TYPE----

ALTER TABLE COFFEE\_SHOP\_SALES

MODIFY COLUMN transaction\_date DATE;

--CONVERT TIME (transaction\_time) COLUMN TO PROPER DATE FORMAT

UPDATE coffee\_shop\_sales

SET transaction\_time = STR\_TO\_DATE(transaction\_time, '%H:%i:%s');

--ALTER TIME (transaction\_time) COLUMN TO DATE DATA TYPE

ALTER TABLE coffee\_shop\_sales

MODIFY COLUMN transaction\_time TIME;

---DATA TYPES OF DIFFERENT COLUMNS---

DESC coffee\_shop\_sales;

**Connecting Snowflake to Power BI**

**Step 8: Establishing Connection**

* **Power BI Connector:** Used the Snowflake connector in Power BI to import data.

1. Open Power BI Desktop.
2. Click on **Get Data** > **Snowflake**.
3. Enter your Snowflake server and warehouse details.
4. Select the relevant database and schema.
5. Load the data into Power BI.

**Creating Power BI Dashboards**

**Step 9: Creating Calculated Columns and Measures using DAX**

Dax code

-- Date Table

DATE\_TABLE = CALENDAR(MIN(COFFEE\_SHOP\_SALES[TRANSACTION\_DATE]), MAX(COFFEE\_SHOP\_SALES[TRANSACTION\_DATE]))

 Purpose: Creates a date table covering the range from the earliest to the latest transaction date.

 Use: Helps in time-based analysis, such as trends over time.

-- Calculated Columns

Day Name = FORMAT('DATE\_TABLE'[Date], "DDD")

 Purpose: Extracts the day name (e.g., Mon, Tue) from the date.

 Use: Useful for daily trend analysis and comparisons.

Day Number = FORMAT('DATE\_TABLE'[Date], "D")

 Purpose: Extracts the day number (e.g., 1, 2) from the date.

 Use: Useful for daily trend analysis within a month.

Month = FORMAT('DATE\_TABLE'[Date],"MMM")

 Purpose: Extracts the month name (e.g., Jan, Feb) from the date.

 Use: Useful for monthly trend analysis and comparisons.

Month Number = MONTH('DATE\_TABLE'[Date])

 Purpose: Extracts the month number (e.g., 1, 2) from the date.

 Use: Useful for sorting months in the correct order.

Week Number = WEEKNUM('DATE\_TABLE'[Date], 2)

 Purpose: Extracts the week number from the date.

 Use: Useful for weekly trend analysis.

Weekday/Weekend = IF('DATE\_TABLE'[Day Name] = "Sat" || 'DATE\_TABLE'[Day Name] = "Sun", "Weekend", "Weekday")

 Purpose: Classifies dates as either "Weekday" or "Weekend".

 Use: Useful for comparing weekday vs. weekend sales.

-- Measures

CM Orders = VAR selected\_month = SELECTEDVALUE('DATE TABLE'[Month])

RETURN TOTALMTD(CALCULATE([Total Orders], 'DATE TABLE'[Month] = selected\_month), 'DATE TABLE'[Date])

 Purpose: Calculates the total orders for the current month.

 Use: Helps in tracking monthly order trends.

CM Quantity = VAR selected\_month = SELECTEDVALUE('DATE TABLE'[Month])

RETURN TOTALMTD(CALCULATE([Total Quantity Sold], 'DATE TABLE'[Month] = selected\_month), 'DATE TABLE'[Date])

 **Purpose**: Calculates the total quantity sold for the current month.

 **Use**: Helps in tracking monthly sales volume trends.

CM Sales = VAR selected\_month = SELECTEDVALUE('DATE TABLE'[Month])

RETURN TOTALMTD(CALCULATE([Total Sales], 'DATE TABLE'[Month] = selected\_month), 'DATE TABLE'[Date])

 Purpose: Calculates the total sales for the current month.

 Use: Helps in tracking monthly revenue trends.

Colour for Bars = IF([Total Sales] > [Daily Avg Sales], "Above Average", "Below Average")

 Purpose: Categorizes sales as above or below the daily average.

 Use: Helps in visualizing performance through color-coding.

Daily Avg Sales = AVERAGEX(ALLSELECTED(COFFEE\_SHOP\_SALES[TRANSACTION\_DATE]), [Total Sales])

 Purpose: Calculates the average daily sales.

 Use: Provides a benchmark for daily sales performance.

Foot Note = "Hover on this visual to see details"

 Purpose: Provides a tooltip for visuals.

 Use: Enhances user interaction with the dashboard.

Label for Product Category = SELECTEDVALUE(COFFEE\_SHOP\_SALES[PRODUCT\_CATEGORY]) & " | " & FORMAT ([Total Sales]/1000, "0.00K")

Label for Product Type = SELECTEDVALUE(COFFEE\_SHOP\_SALES[PRODUCT\_TYPE]) & " | " & FORMAT ([Total Sales]/1000, "0.00K")

Label for Store Location = SELECTEDVALUE(COFFEE\_SHOP\_SALES[STORE\_LOCATION]) & " | " & FORMAT ([Total Sales]/1000, "0.00K")

 Purpose: Creates labels combining product category/type/location with total sales.

 Use: Helps in quickly identifying sales performance by category, type, or location.

MoM Growth & Diff Orders =

VAR month\_diff = [CM Orders] - [PM Orders]

VAR mom = ([CM Orders] - [PM Orders]) / [PM Orders]

VAR \_sign = IF(month\_diff > 0, "+", "")

VAR \_sign\_trend = IF(month\_diff > 0, "▲", "▼")

RETURN \_sign\_trend & " " & \_sign & FORMAT(mom, "#0.0%") & " | " & \_sign & FORMAT(month\_diff/1000, "0.0K") & " vs LM"

MoM Growth & Diff Quantity =

VAR month\_diff = [CM Quantity] - [PM Quantity]

VAR mom = ([CM Quantity] - [PM Quantity]) / [PM Quantity]

VAR \_sign = IF(month\_diff > 0, "+", "")

VAR \_sign\_trend = IF(month\_diff > 0, "▲", "▼")

RETURN \_sign\_trend & " " & \_sign & FORMAT(mom, "#0.0%") & " | " & \_sign & FORMAT(month\_diff/1000, "0.0K") & " vs LM"

MoM Growth & Diff Sales =

VAR month\_diff = [CM Sales] - [PM Sales]

VAR mom = ([CM Sales] - [PM Sales]) / [PM Sales]

VAR \_sign = IF(month\_diff > 0, "+", "")

VAR \_sign\_trend = IF(month\_diff > 0, "▲", "▼")

RETURN \_sign\_trend & " " & \_sign & FORMAT(mom, "#0.0%") & " | " & \_sign & FORMAT(month\_diff/1000, "0.0K") & " vs LM"

 Purpose: Calculates Month-over-Month (MoM) growth and difference for orders, quantity, and sales.

 Use: Provides insights into how performance is changing month over month.

New MoM Label =

VAR month\_diff = [CM Sales] - [PM Sales]

VAR mom = ([CM Sales] - [PM Sales]) / [PM Sales]

VAR \_sign = IF(month\_diff > 0, "+", "")

VAR \_sign\_trend = IF(month\_diff > 0, "▲", "▼")

RETURN \_sign\_trend & " " & \_sign & FORMAT(mom, "#0.0%")

 Purpose: Creates a simplified MoM label for sales.

 Use: Provides a quick visual indicator of MoM sales growth.

Placeholder = 0

 Purpose: Acts as a placeholder value.

 Use: Used for layout or formatting purposes in visuals.

PM Orders = CALCULATE([CM Orders], DATEADD('DATE TABLE'[Date], -1, MONTH))

PM Quantity = CALCULATE([CM Quantity], DATEADD('DATE TABLE'[Date], -1, MONTH))

PM Sales = CALCULATE([CM Sales], DATEADD('DATE TABLE'[Date], -1, MONTH))

 Purpose: Calculates previous month’s orders, quantity, and sales.

 Use: Provides a basis for MoM comparisons.

Sales = COFFEE\_SHOP\_SALES[TRANSACTION\_QTY] \* COFFEE\_SHOP\_SALES[UNIT\_PRICE]

 Purpose: Calculates sales by multiplying quantity and unit price.

 Use: Provides the total sales amount.

Tool Tip KPI For Hours = "Hour No:" & " " & FORMAT(AVERAGE(COFFEE\_SHOP\_SALES[Hour]), 0)

 Purpose: Creates a tooltip showing the average hour number.

 Use: Enhances user interaction with hour-based data.

Total Orders = DISTINCTCOUNT(COFFEE\_SHOP\_SALES[TRANSACTION\_ID])

 Purpose: Counts distinct transaction IDs to get the total number of orders.

 Use: Measures the volume of orders.

-- Sales by Store Location, Product Type, etc.

Sales by Store Location = SUMMARIZE(COFFEE\_SHOP\_SALES\_TRANSFORMED, COFFEE\_SHOP\_SALES\_TRANSFORMED[store\_location], "Total Sales", [Total Sales])

**Step 10: Building Dynamic Dashboards**

* Created visualizations for:
  + Sales by Store Location
  + Sales by Product Type and Category
  + Sales by Days and Hours
  + Sales by Weekend/Weekdays
  + KPIs for Total Sales, Total Orders, Total Quantity Sold
  + Sales Trend Over Time

**Learnings from the Coffee Sales Analysis Project**

**Technical Skills and Tools**

**1. AWS Services**

* **IAM Role and Policy Creation**: Learned to create IAM roles and policies, understanding how to manage permissions securely and effectively.
* **S3 Bucket Management**: Gained experience in managing S3 buckets, including data upload and integration with other services.

**2. Snowflake**

* **Warehouse and Database Setup**: Understood how to set up and configure a Snowflake data warehouse, including creating databases and schemas.
* **Data Loading and Transformation**: Learned to load data into Snowflake from S3 using stages and pipes, and performed data transformation using SQL.
* **Storage Integration**: Mastered creating storage integrations between Snowflake and AWS S3, ensuring secure and efficient data transfer.

**3. Power BI**

* **Data Connection**: Connected Power BI to Snowflake, learning to import and refresh data dynamically.
* **DAX Calculations**: Developed skills in creating calculated columns and measures using DAX to generate meaningful insights and KPIs.
* **Dynamic Dashboards**: Built interactive dashboards, including various visualizations and slicers, enhancing data exploration and decision-making.

**Project Management and Best Practices**

**4. End-to-End Data Pipeline**

* **Data Integration**: Understood the complete data integration process from AWS to Snowflake, ensuring smooth data flow and integrity.
* **ETL Process**: Mastered the ETL (Extract, Transform, Load) process, including data extraction from S3, transformation in Snowflake, and loading into Power BI.
* **Automation**: Learned to automate data ingestion and transformation using Snowflake pipes, reducing manual intervention and improving efficiency.

**5. Data Visualization**

* **KPI Development**: Created key performance indicators (KPIs) to monitor business performance, enhancing the ability to make data-driven decisions.
* **Trend Analysis**: Performed trend analysis to identify patterns and insights over time, providing valuable business intelligence.
* **User Experience**: Focused on creating a user-friendly dashboard in Power BI, enabling stakeholders to easily interact with and understand the data.

**Analytical Thinking and Problem Solving**

**6. Data Transformation**

* **SQL Skills**: Improved SQL skills by performing complex data transformations, aggregations, and calculations in Snowflake.
* **Data Cleaning**: Gained experience in data cleaning and preparation, ensuring high-quality and accurate data for analysis.

**7. Business Insights**

* **Sales Analysis**: Analyzed coffee sales data to derive insights such as sales trends, best-selling products, and performance by store location.
* **Decision Support**: Provided actionable insights to support strategic business decisions, demonstrating the impact of data analytics on business growth.

**Collaboration and Communication**

**8. Documentation**

* **Detailed Reporting**: Created comprehensive documentation of the entire project, including code snippets, setup instructions, and analysis results.
* **Knowledge Sharing**: Shared findings and methodologies on platforms like GitHub, fostering a culture of collaboration and continuous learning.
* **Conclusion**
* The coffee sales analysis project provided a rich learning experience across various domains, including cloud services, data warehousing, ETL processes, data visualization, and business analytics. The skills and knowledge gained from this project are invaluable for any data professional aiming to deliver end-to-end data solutions that drive business value.