2024

رواد مصر الرقمية

Supply Chain Data Engineering and Analysis Project

DEPI INITIATIVE GRADUATION PROJECT
AI & DATA SCIENCE TRACK
MICROSOFT DATA ENGINEER PROFILE

PREPARED BY:

- Abdallah Albanna
- Khaled Tarek Mohamed
- Abdelrhman Ibrahim
- Abdelrhamn Samir
- Mohammed Elmehy

SUPERVISED BY: | Eng. Ahmed Essam Azab

Table Of Contents

1 Introduction

Dataset Overview: This project evaluates the supply chain inefficiencies of a Supply Chain Company. We will analyze business performance, customer and product data, and inventory management. Our goal is to identify key issues and suggest data-driven solutions.

The dataset provides a comprehensive overview of Just in Time Company's supply chain process, including orders, products, customers, stores, sales transactions, and shipping logistics. The data is organized using a star schema structure, enabling effective analysis of business performance.

Components:

- **Sales:** Contains transaction records including order details, product pricing, and discounts.
- **Transactions:** These include financial metrics like earnings and profit per order.
- **Customer Interactions:** Data on customer profiles, segments, and locations.
- **Shipping Logistics:** Real vs. scheduled delivery times, shipping methods, and delivery status.
- **Product Management:** Product information including IDs, descriptions, and availability.

Project Goals:

- **Personalize Interactions:** Tailor marketing based on customer data.
- **Evaluate Products:** Guide inventory and development decisions using product performance data.
- **Increase Revenue:** Refine pricing strategies and sales tactics based on sales data.
- **Optimize Shipping:** Reduce delays by analyzing real and scheduled shipping times.

2 Database Design

Process:

2.1 **Data Collection:** Data was gathered from various sources, including sales, inventory, and customer data.

2.2 **Understanding the Data:** Initial exploration of the dataset to identify relationships and patterns.

2.3 **Entity Relationship Diagram (ERD):** Data is grouped into entities such as orders, products, and customers, to establish how these entities relate to one another.

2.4 **Design Mapping:** Linking entities across tables, focusing on organizing data for optimal insights.

2.5 **SQL Table Creation:** Tables were created in SQL based on the entity relationships.

2.6 **Data Transfer:** Data was migrated from Excel files to the SQL database using SSIS (SQL Server Integration Services).

3 Extraction, Transformation, Load (ETL) Process

The first step in a data engineering workflow is building an ETL (Extract, Transform, Load) pipeline. This is essential because raw data is often scattered across different sources and contains various inconsistencies. Here's what each phase entails:

1. **Extract**: Collect data from different sources, which may vary in format or location.
2. **Transform**: Clean and prepare the data by handling issues like missing values, human errors, and inconsistencies.
3. **Load**: Once cleaned, load the data into a storage system or database for further analysis and use.

For this project, we will be using the **Pandas** library, a powerful Python tool for data manipulation and analysis.

Since our dataset is stored locally, we can proceed directly to loading the data and exploring its contents.

Data Exploration and Identifying Issues

Now that we've loaded a sample of the data, we can already identify some initial issues. For example:

- **Order YearMonth:** The column name and the values within it are inconsistent or improperly formatted.
- **Customer Country:** There are inconsistencies in the values, which need to be addressed.

However, to proceed with the data-cleaning process, we need a deeper understanding of the dataset. Simply looking at a sample is not enough. We need to explore further by:

- **Inspecting the Shape:** To know the number of rows and columns.
- **Listing Column Names:** To ensure consistency and identify potential naming issues.
- **Checking Data Types:** To verify if columns have appropriate data types (e.g., dates, integers, etc.).
- **Checking for Missing Values:** To identify columns that have incomplete data.

Step 1: Inspecting the Shape

The dataset contains 30871 rows and 24 columns

Step 2: Explore names of the columns

Upon inspecting the dataset, we’ve identified additional issues, such as spaces before some column names. These inconsistencies can cause problems when referencing columns in our transformations, so it’s important to clean them early in the process.

Step 3: Checking Data Types and Missing Values

Although our data doesn't have any missing values, we can already spot an issue with the **Discount %** column: its data type is `object`, while it should be **float**. This type of mismatch is an example of **messy data**, where the structure of the data doesn't align with our expectations.

Before solving the issues in our dataset, let's define two key terms:

1. **Dirty Data:** which has issues with its content is often called low-quality data and can include things like inaccurate data, corrupted data, and duplicate data.
2. **Messy Data:** Refers to data that has structural problems, such as incorrect data types, misaligned columns, or inconsistent formats.

As observed, the **Discount %** column's data type should be **float**, but it is currently an object. This issue of incorrect data type is a prime example of messy data. To resolve it, let's take a closer look at the column and investigate the underlying problem.

The issue with the **Discount %** column arises because some of its values are represented by dashes (-), which signifies no discount. Since these dashes are treated as text, they prevent the column from being recognized as a numerical data type.

To resolve this, we will:

1. Replace the dashes `-` with 0 to indicate no discount.
2. Convert the data type of the **Discount %** column from object to float, allowing us to perform numerical operations on it.

Above we see an issue in values of Customer country and this is an example on Dirty data so we need to see if there are other issues like this or not

There are 5 issues in our data so we are going to clean them

We will remove the unnecessary columns

Now we will create two new columns:

1. Create a Shipping time column
2. Create a column to see if there is a delay in shipping or not

First, we need to combine the three columns related to the shipping date (Year - Month - Day) and put them in a single column and do the same thing in order date and ensure to make the data type of these two columns as datetime.

Second, we will get the difference between the Shipment date and the Order date

Step 1: Create a Shipping Time Column

We need to calculate the time difference between when the order was placed and when it was shipped. To do this:

1. **Combine Date Columns:** We will merge the three columns related to both the Order date and Shipment date (Year, Month, and Day) into two separate columns: Order date and Shipment date. This will ensure that the date is represented as a single unit.
2. **Convert to Datetime:** Once combined, we'll convert both columns into datetime format to make it possible to compute the time difference between them.
3. **Calculate Shipping Time:** We will subtract the Order date from the Shipment date to calculate the time it took to ship the order. This will be stored in a new column called Shipping Time.

Step 2: Create a Delay Status Column

To check whether the order was shipped on time or delayed:

1. **Define Delay Criteria:** If the Shipping time is greater than the expected shipping days **Shipment Days - Scheduled**, we will mark the order as **Late**.
2. **Create a Delay Shipment Column:** We will add a new column, **Delay Shipment**, which will store the value **Late** if the shipping time exceeds the threshold, and **On-time** if the order was shipped within the expected time frame.

Data Modeling and Star Schema Design

After cleaning the dataset, we move to the Data Modeling phase of our project. This step is critical because it lays the foundation for organizing, storing, and relating our data, which will later be used for analysis and insights.

What is Data Modeling?

Data Modeling is the process of designing the structure of data, specifying how the data will be organized, stored, and connected with other data. It provides a blueprint for the data to ensure that it is both scalable and easy to analyze.

Why Star Schema?

There are many types of schemas for organizing data, but the Star Schema is particularly well-suited for our project because of its simplicity and efficiency in handling analytical queries. The Star Schema consists of:

- A central fact table that stores quantitative data.
- Dimension tables that store descriptive data and provide context to the facts.

The ERD diagram for our Data Modeling will be like that

Star Schema Structure

In the context of our supply chain project, the dimension tables describe the characteristics of orders, products, customers, shipments, and warehouses, while the fact table stores the numeric data relevant to our analysis.

1. **Dimension Tables:**
    - **DimProduct**: Contains information about the products such as department, category, and name.
    - **DimCustomer**: Holds data about the customers, including customer market, region, and country.
    - **DimDate**: Stores details about order and shipment dates.
    - **DimShipment**: Contains information about Shipment dates, Mode, Delay Status
    - **DimWarehouse**: Contains information about warehouses, such as location.

2. **Fact Table:**

    The fact table store quantitative, numerical data, which are typically measurements that are the main focus of analysis such as:

    - **Profit**
    - **Gross Sales**
    - **Discount**
    - **Shipping Time**

These dimensions and facts form a Star Schema, where the fact table is at the center, linked to each dimension table.

Creating the Dimension Tables

To create the dimension tables, we will:

2.1. Extract relevant columns from the dataset.

2.2. Remove duplicate values (dimension tables should not have duplicates).

2.3. Assign a primary key to each dimension table. This will allow us to create relationships between the dimension and fact tables.

Example: Creating the Product Dimension Table

We will start by creating the **DimProduct** table:

2.4. Extract relevant columns: **Product Department**, **Product Category**, **Product Name**.

2.5. Remove duplicates to ensure each product is uniquely represented.

2.6. Add a primary key column to uniquely identify each product.

We'll apply the same process to other dimension tables like **DimCustomer**, **DimDate**, **DimWarehouse,** and **DimShipment**.

This step ensures that our data is properly organized and ready for analysis using the star schema model. The relationships between these dimension tables and the central fact table will form the basis of our data warehouse.

Create a dimension table for customers:

Although we removed duplicates, we still notice that there are 2967 duplicates in our data.

Let's examine this issue.

We see a sample of this issue where different customers share the same Customer ID. This requires us to create a new unique Customer ID for each customer, ensuring that our Customer Dimension table has no duplicates and can be correctly linked to the Fact Table.

Create Warehouse Dimension

Create Shipment Dimension

Create Date Dimension

This ERD diagram illustrates the relationships between **the Fact Table** and **the Dimension Tables** in our data model. It follows the **Star Schema** approach, where the central **Fact Table** stores the transactional data, and the surrounding **Dimension Tables** contain descriptive information for context. This design helps to efficiently structure the data for analysis, ensuring quick and scalable queries.

Replacing Attributes with Foreign Keys in the Fact Table

Once the dimension tables have been created, the next step is to link them to the fact table. To do this, we need to replace the attributes from the dimension tables with their corresponding **primary keys**. These primary keys will act as **foreign keys** in the fact table, enabling the relationship between the fact table and the dimension tables.

Why Replace Attributes with Foreign Keys?

In a **Star Schema**, **the dimension tables** store descriptive attributes (e.g., **product name**, **customer region**), while **the fact table** stores quantitative data. To create relationships between the fact table and dimension tables, we use foreign keys. By replacing the attributes with foreign keys, we ensure that each row in the fact table is linked to the corresponding row in the dimension tables.

Steps to Replace Attributes:

- Identify the attributes from the fact table that correspond to the dimension tables (e.g., customer-related fields, product-related fields).
- Replace these attributes in the fact table with the primary keys from the relevant dimension tables.

These primary keys will now act as foreign keys, linking the fact table to each dimension table.

Example Process:

- **DimProduct**: Replace product-related columns (like Product Name) in the fact table with the ProductID from the DimProduct table.
- **DimCustomer**: Replace customer-related columns (like Customer Country) with the CustomerID from the DimCustomer table.
- **DimDate**: Replace date-related columns (like Order Date) with the DateID from the DimDate table.
- **DimShipment**: Replace shipment-related columns (like Shipment date) with the ShipmentID from the DimShipment table.
- **DimWarehouse**: Replace warehouse-related columns with the WarehouseID from the DimWarehouse table.

By doing this, we create a clean and efficient relationship between the fact table and the dimension tables, which is essential for querying and analyzing the data.

Load dimension tables and fact table

3. Designing Data Warehouse

Introduction to DWH

This part of the report outlines the implementation process of a Data Warehouse (DWH) using Azure Synapse Analytics to address business analysis needs through dimensional modeling. The project involves using Azure Synapse Analytics workspace, an Azure Storage account, and a Data Lake to store data files. Additionally, the report describes the creation of a serverless SQL pool for data loading and querying purposes.

The objectives are:

- Design and implement a DWH using dimensional modeling.
- Use Azure Synapse Analytics to store and manage data.
- Load Fact and Dimension tables into the Synapse database for analysis.

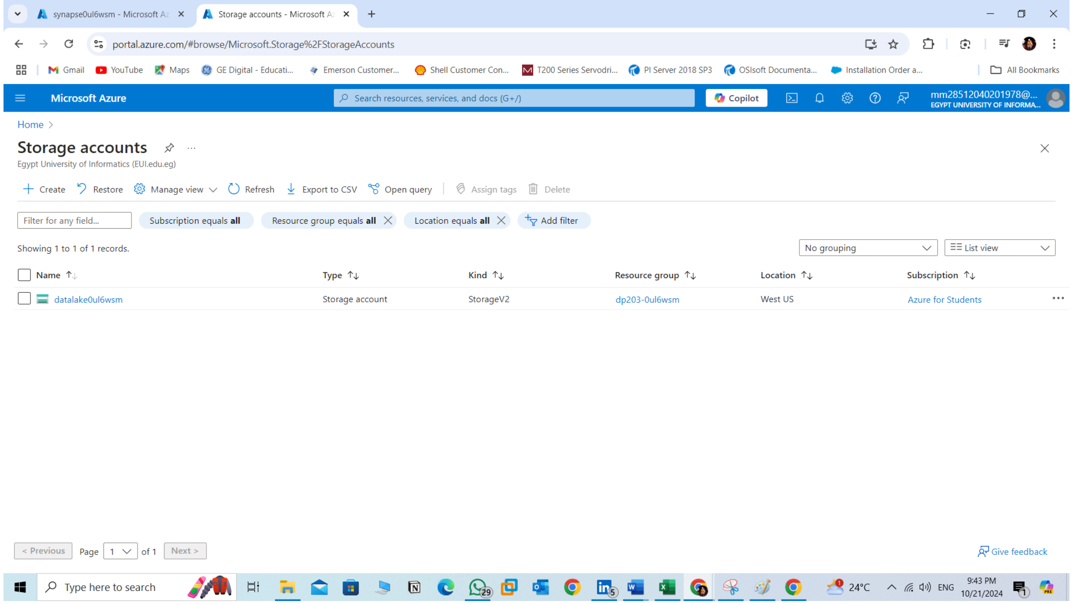
Data Warehouse Design Overview

The data warehouse is designed to support business analytics using dimensional modeling. This approach allows for organizing data into Fact and Dimension tables, facilitating efficient querying and analysis.

- **Azure Synapse Analytics Workspace**: Acts as the central point for data integration, processing, and analysis.
- **Azure Storage Account**: Provides storage for the data lake where the initial data files are uploaded.
- **Serverless SQL Pool**: Used for querying the data in the Data Lake and integrating it into the Synapse environment.

Creating a Storage Account

- **Navigate to Azure Portal**: Go to the Azure portal and search for "Storage Accounts."
- **Create a Storage Account**: Click "Create" and fill in the required details, such as subscription, resource group, and storage account name.
- **Configuration**: Choose the performance, replication, and access tier options based on data volume and access frequency.
- **Review and Create**: Validate the information and create the storage account.



portal.azure.com/#browse/Microsoft.Storage%2FStorageAccounts

Storage accounts

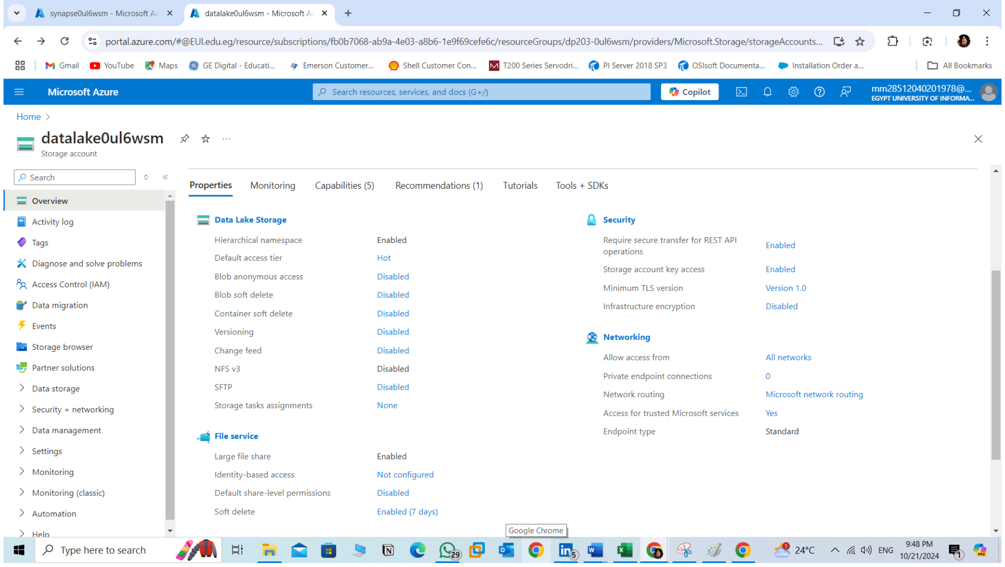
Egypt University of Informatics (EUI.edu.eg)

Showing 1 to 1 of 1 records.

Name	Type	Kind	Resource group	Location	Subscription
datalake0ul6wsm	Storage account	StorageV2	dp203-0ul6wsm	West US	Azure for Students

Creating a Data Lake Storage

Datalake is where the initial data files will be uploaded.



portal.azure.com/#@EUI.edu.eg/resource/subscriptions/fb0b7068-ab9a-4e03-a8b6-1e9f69cefe6c/resourceGroups/dp203-0ul6wsm/providers/Microsoft.Storage/storageAccounts...

datalake0ul6wsm

Storage account

Data Lake Storage

Hierarchical namespace	Enabled
Default access tier	Hot
Blob anonymous access	Disabled
Blob soft delete	Disabled
Container soft delete	Disabled
Versioning	Disabled
Change feed	Disabled
NFS v3	Disabled
SFTP	Disabled
Storage tasks assignments	None

File service

Large file share	Enabled
Identity-based access	Not configured
Default share-level permissions	Disabled
Soft delete	Enabled (7 days)

Security

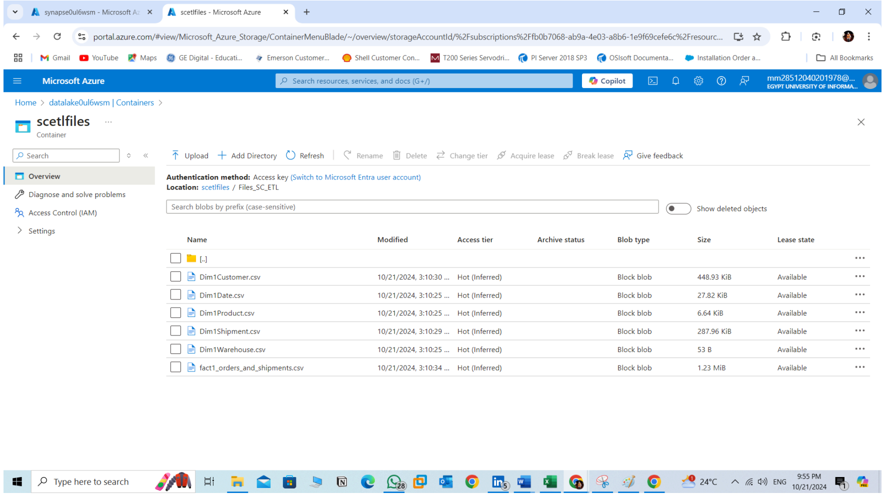
Require secure transfer for REST API operations	Enabled
Storage account key access	Enabled
Minimum TLS version	Version 1.0
Infrastructure encryption	Disabled

Networking

Allow access from	All networks
Private endpoint connections	0
Network routing	Microsoft network routing
Access for trusted Microsoft services	Yes
Endpoint type	Standard

Creating a Container on the Data Lake and uploading Fact and Dimension Tables

Created the Container folder “scetlfiles” (Supply Chain ETL Files) to upload the Fact and Dimension Tables to it.



synapse0ul6wsm - Microsoft Az
scetlfiles - Microsoft Azure
portal.azure.com/#view/Microsoft_Azure_Storage/ContainerMenuBlade/~/overview/storageAccountId/%2Fsubscriptions%2Ffb0b7068-ab9a-4e03-a8b6-1e9f69cefe6c%2Fresourc...
Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
Home > datalake0ul6wsm | Containers >
scetlfiles
Container
Search
Overview
Diagnose and solve problems
Access Control (IAM)
Settings
Upload
Add Directory
Refresh
Rename
Delete
Change tier
Acquire lease
Break lease
Give feedback
Search blobs by prefix (case-sensitive)
Show deleted objects

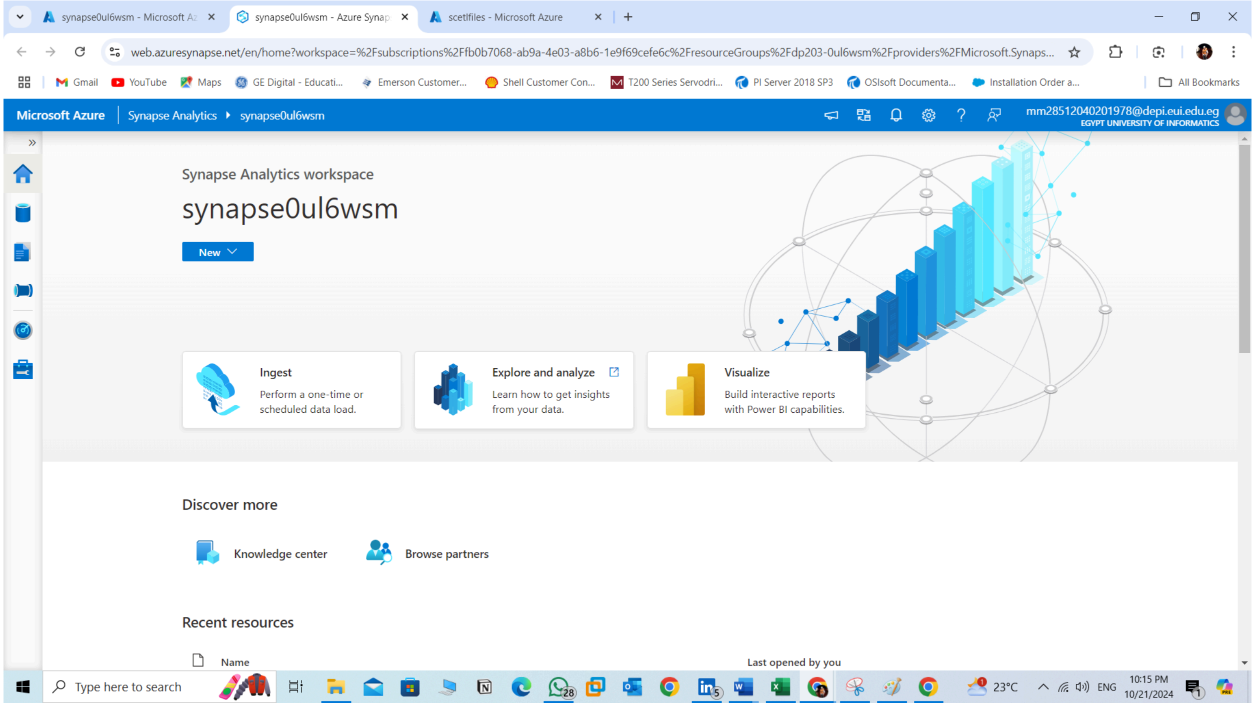
Authentication method: Access key (Switch to Microsoft Entra user account)
Location: scetlfiles / Files_SC_ETL

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[..]						
Dim1Customer.csv	10/21/2024, 3:10:30 ...	Hot (Inferred)		Block blob	448.93 KiB	Available
Dim1Date.csv	10/21/2024, 3:10:25 ...	Hot (Inferred)		Block blob	27.82 KiB	Available
Dim1Product.csv	10/21/2024, 3:10:25 ...	Hot (Inferred)		Block blob	6.64 KiB	Available
Dim1Shipment.csv	10/21/2024, 3:10:29 ...	Hot (Inferred)		Block blob	287.96 KiB	Available
Dim1Warehouse.csv	10/21/2024, 3:10:25 ...	Hot (Inferred)		Block blob	53 B	Available
fact1_orders_and_shipments.csv	10/21/2024, 3:10:34 ...	Hot (Inferred)		Block blob	1.23 MiB	Available

4. Data loading to Azure Synapse

Creating a Synapse Workspace

- **In Azure Portal**: Search for "Azure Synapse Analytics" and select "Create."
- **Workspace Details**: Enter the details like subscription, resource group, workspace name, and region.
- **Configure Networking**: Select the networking options that fit the organization’s security needs.
- **Review and Create**: Review the configurations and create the Synapse workspace.



Synapse Analytics workspace

synapse0ul6wsm

Ingest
Perform a one-time or scheduled data load.

Explore and analyze
Learn how to get insights from your data.

Visualize
Build interactive reports with Power BI capabilities.

Discover more

Knowledge center

Browse partners

Recent resources

Name

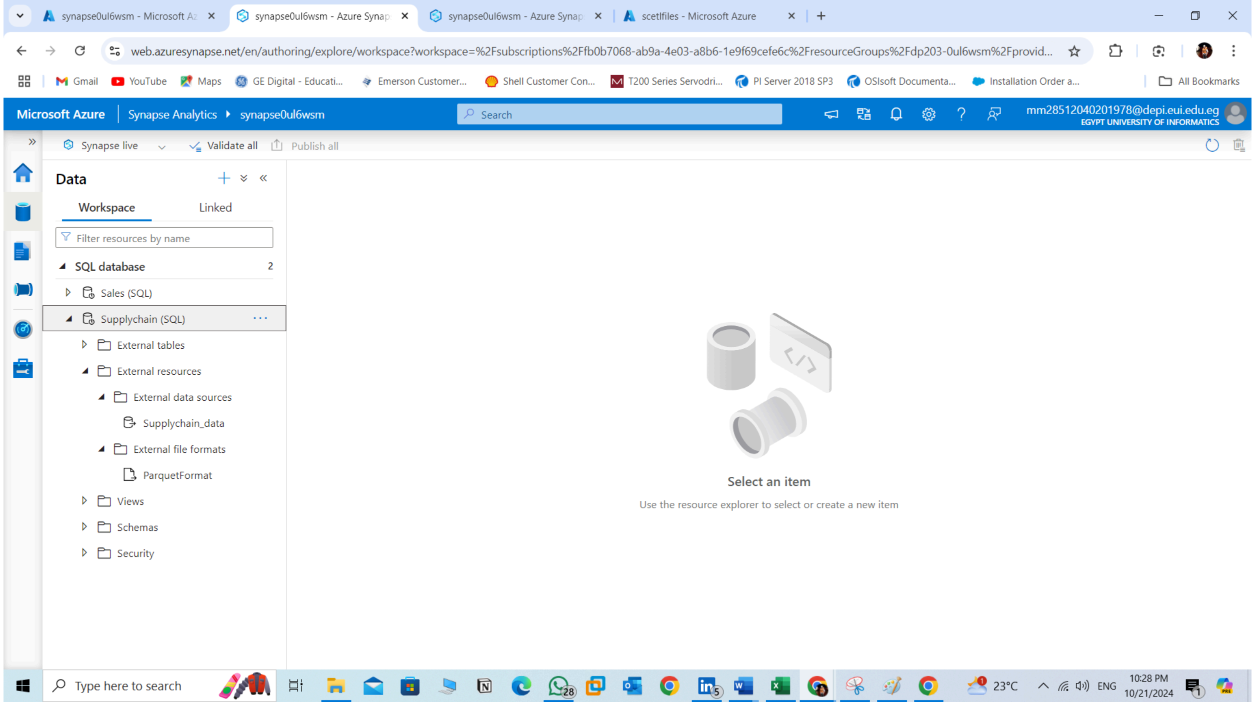
Last opened by you

Creating a SQL Database (SUPPLY CHAIN)

- **Open Synapse Studio**: Once the workspace is created, open Synapse Studio.
- **Create SQL Database**: Go to the "Manage" tab, and under "SQL Pools," create a new SQL database named "SUPPLY CHAIN."
- **Configure Database Settings**: Set the compute tier and size based on the expected data volume and query performance requirements.

Create an external data source and file format

By defining an external data source in a database, it can be used to reference the data lake location where you want to store files for external tables. An external file format enables you to define the format for those files - for example, Parquet or CSV.



web.azuresynapse.net/en/authoring/explore/workspace?workspace=%2Fsubscriptions%2Ffb0b7068-ab9a-4e03-a8b6-1e9f69cefe6c%2FresourceGroups%2Fdp203-0ul6wsm%2Fprovid...

mm28512040201978@depi.eui.edu.eg
EGYPT UNIVERSITY OF INFORMATICS

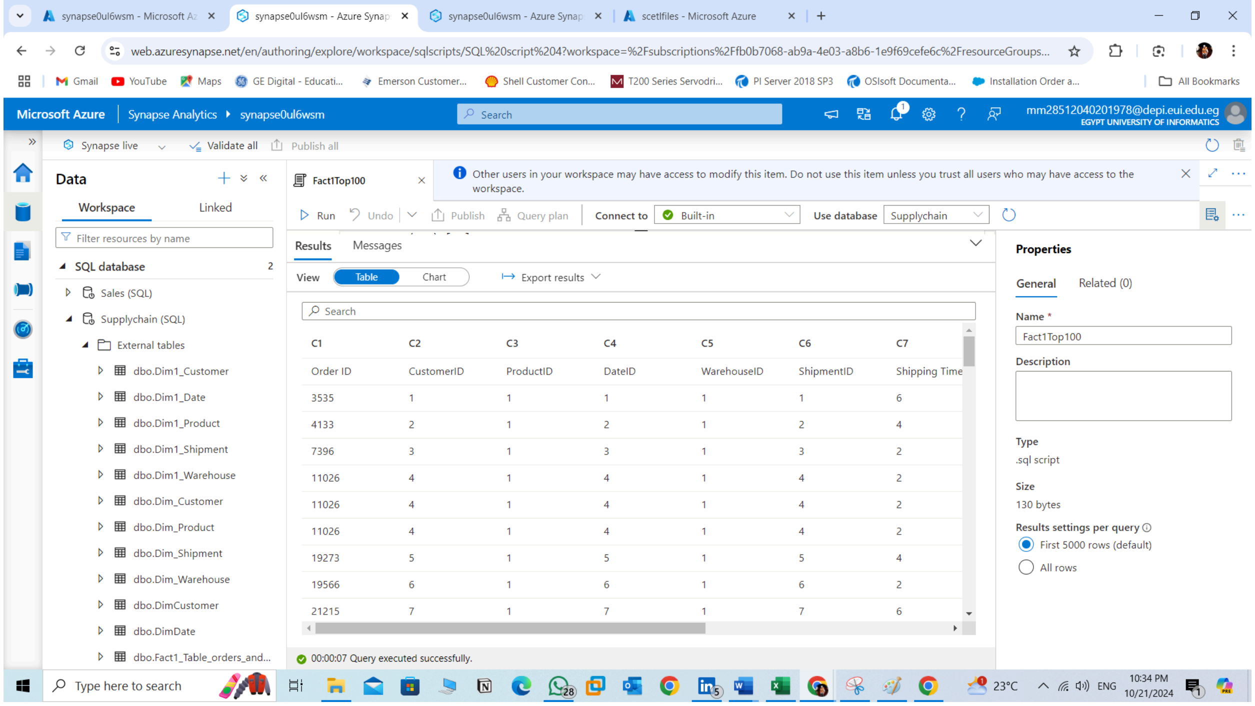
- SQL database 2
  - Sales (SQL)
  - Supplychain (SQL)
    - External tables
    - External resources
      - External data sources
        - Supplychain_data
      - External file formats
        - ParquetFormat
    - Views
    - Schemas
    - Security

Select an item

Use the resource explorer to select or create a new item

10:28 PM 10/21/2024

Loading Fact and Dimension Tables into the Created Supply Chain SQL Database



web.azuresynapse.net/en/authoring/explore/workspace/sqlscripts/SQL%20script%204?workspace=%2Fsubscriptions%2Ffb0b7068-ab9a-4e03-a8b6-1e9f69cefe6c%2FresourceGroups...

mm28512040201978@depi.eui.edu.eg
EGYPT UNIVERSITY OF INFORMATICS

- SQL database 2
  - Sales (SQL)
  - Supplychain (SQL)
    - External tables
      - dbo.Dim1_Customer
      - dbo.Dim1_Date
      - dbo.Dim1_Product
      - dbo.Dim1_Shipment
      - dbo.Dim1_Warehouse
      - dbo.Dim_Customer
      - dbo.Dim_Product
      - dbo.Dim_Shipment
      - dbo.Dim_Warehouse
      - dbo.DimCustomer
      - dbo.DimDate
      - dbo.Fact1_Table_orders_and...

Fact1Top100

Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Connect to: Built-in   Use database: Supplychain

C1	C2	C3	C4	C5	C6	C7
Order ID	CustomerID	ProductID	DateID	WarehouseID	ShipmentID	Shipping Time
3535	1	1	1	1	1	6
4133	2	1	2	1	2	4
7396	3	1	3	1	3	2
11026	4	1	4	1	4	2
11026	4	1	4	1	4	2
11026	4	1	4	1	4	2
19273	5	1	5	1	5	4
19566	6	1	6	1	6	2
21215	7	1	7	1	7	6

00:00:07 Query executed successfully.

Name *: Fact1Top100

Description

Type: .sql script

Size: 130 bytes

Results settings per query
● First 5000 rows (default)
○ All rows

10:34 PM 10/21/2024

5. Machine Learning and Visualization

Overview: The project applies machine learning techniques to enhance decision-making in supply chain management.

Machine Learning Models:

- **Logistic Regression:** Used to predict shipment mode based on customer and order data.
- **Random Forest Regression:** Applied to predict profit from sales-related features.

Model Evaluation:

- **Logistic Regression:** Evaluated using accuracy, precision, recall, F1-score, and confusion matrix.
- **Random Forest Regression:** Evaluated using MAE, MSE, and R^2 score.

Visualization:

- **Profit Prediction:** Scatter plots visualize actual vs. predicted profits.
- **Shipment Mode Classification:** Confusion matrix heatmaps to evaluate model performance.

---

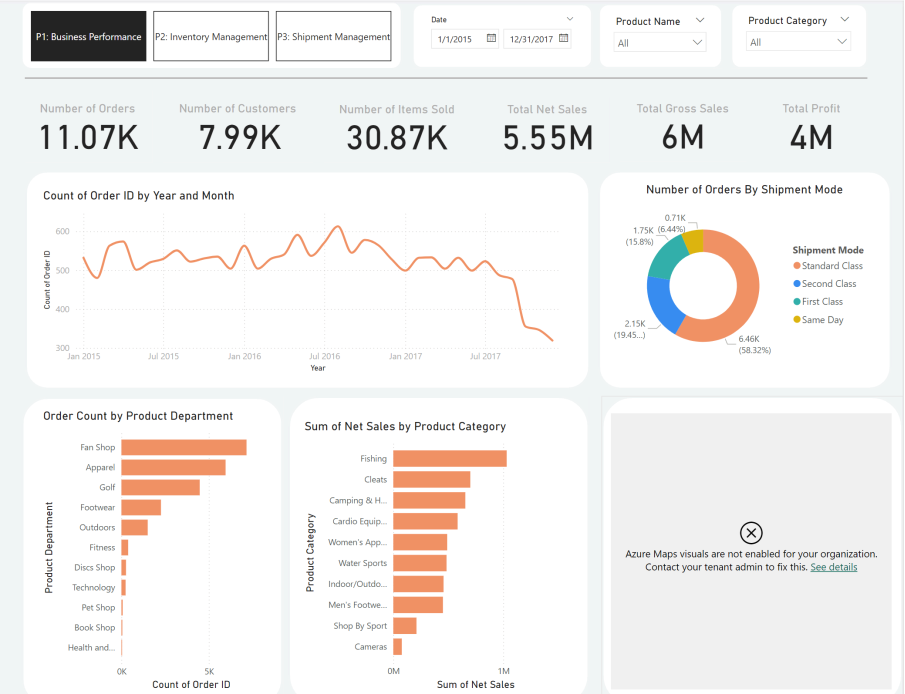
6. Data Reporting in Power BI

Goals & Insights:

- **Business Performance:** Analysis of financial metrics and overall company health.
- **Inventory Management:** Evaluating current inventory practices and identifying areas for improvement.
- **Shipment Management**: Overseeing daily shipping and distribution operations to customers

Visualization Tools:

- **Power BI Dashboards:** Used for business performance, inventory, shipment management, and customer insights.

- 

P1: Business Performance | P2: Inventory Management | P3: Shipment Management

Date: 1/1/2015 – 12/31/2017

Product Name: All

Product Category: All

Number of Orders: 11.07K

Number of Customers: 7.99K

Number of Items Sold: 30.87K

Total Net Sales: 5.55M

Total Gross Sales: 6M

Total Profit: 4M

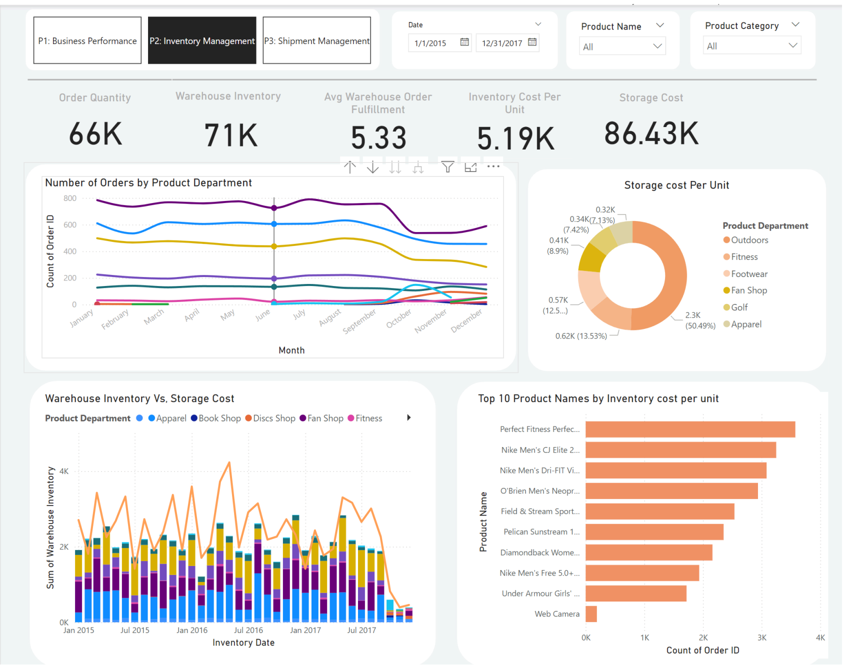
Count of Order ID by Year and Month

Number of Orders By Shipment Mode

Order Count by Product Department

Sum of Net Sales by Product Category

Azure Maps visuals are not enabled for your organization. Contact your tenant admin to fix this. See details

- 

P1: Business Performance | P2: Inventory Management | P3: Shipment Management

Date: 1/1/2015 – 12/31/2017

Product Name: All

Product Category: All

Order Quantity: 66K

Warehouse Inventory: 71K

Avg Warehouse Order Fulfillment: 5.33

Inventory Cost Per Unit: 5.19K

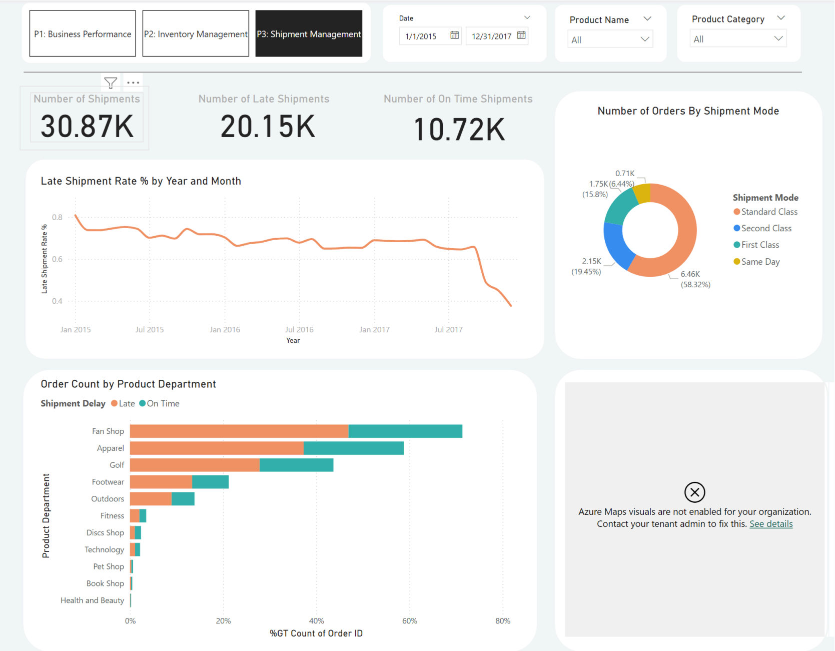
Storage Cost: 86.43K

Number of Orders by Product Department

Storage cost Per Unit

Warehouse Inventory Vs. Storage Cost

Top 10 Product Names by Inventory cost per unit

- 

7. Conclusion

This project utilized data engineering, visualization, and machine learning to assess and enhance supply chain processes. Key insights were gathered to improve business performance, product management, and customer satisfaction. Actionable recommendations were provided to enhance inventory management and shipping logistics.