



**MEC Academy**

# Data Analysis

## SuperMarket Analysis

# Data Analysis

Team Member:

- Noor Fared
- Mohamed Ataa
- Omar Mahmoud

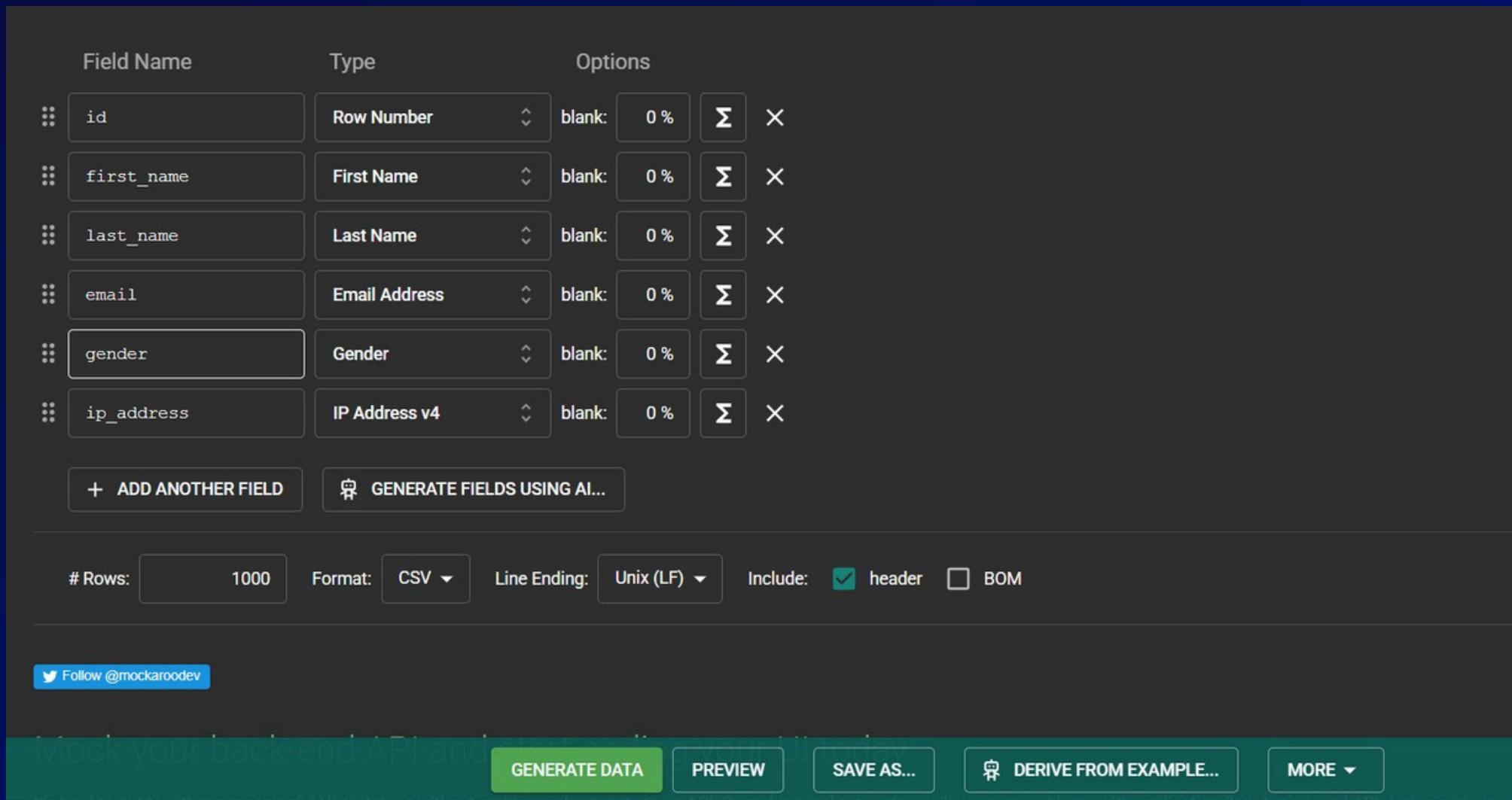




# Tools Used

- 1-mockaroo(Data generation)**
- 2-SQL Server Management Studio(SQL)**
- 3-Jupyter Notebook(Python)**
- 4-PowerBi (Dashboard)**

# Data generation



The screenshot shows the Mockaroo data generation interface. It displays a table of fields with their types and generation options. The fields are:

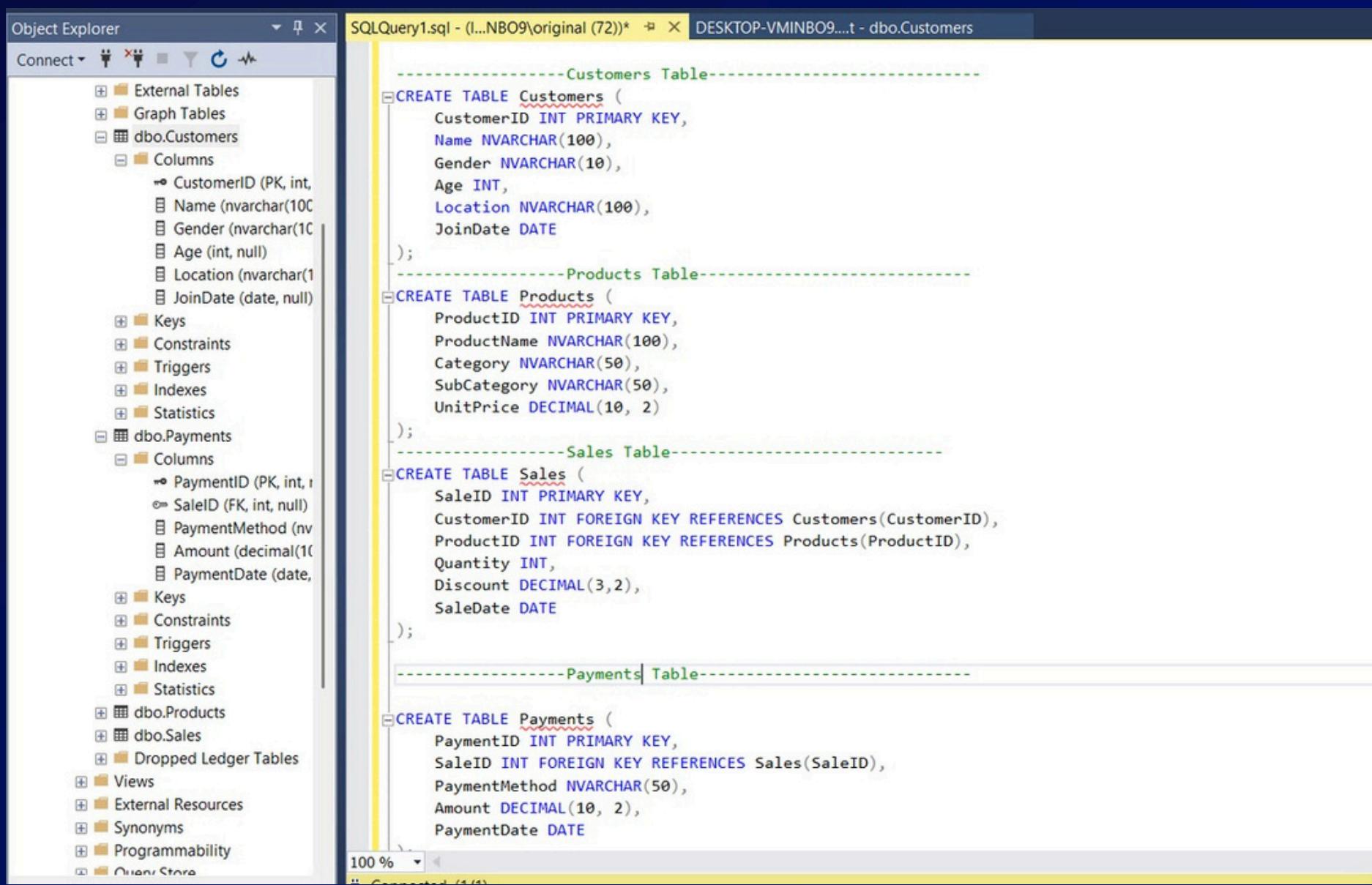
Field Name	Type	Options
id	Row Number	blank: 0 % $\Sigma$ X
first_name	First Name	blank: 0 % $\Sigma$ X
last_name	Last Name	blank: 0 % $\Sigma$ X
email	Email Address	blank: 0 % $\Sigma$ X
gender	Gender	blank: 0 % $\Sigma$ X
ip_address	IP Address v4	blank: 0 % $\Sigma$ X

Below the table are buttons for "ADD ANOTHER FIELD" and "GENERATE FIELDS USING AI...". At the bottom, there are settings for "# Rows: 1000", "Format: CSV", "Line Ending: Unix (LF)", "Include: header" (checked), and "BOM" (unchecked). A "Follow @mockaroodev" button is also present. At the bottom right of the interface is a "Mock your back-end API and generate data" button.

GENERATE DATA PREVIEW SAVE AS... DERIVE FROM EXAMPLE... MORE ▾

Use mockaroo  
program to generate  
data used in the  
.analysis

# SQL Codes



The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The Object Explorer on the left lists database objects: External Tables, Graph Tables, dbo.Customers (with columns CustomerID, Name, Gender, Age, Location, JoinDate), dbo.Payments (with columns PaymentID, SaleID, PaymentMethod, Amount, PaymentDate), dbo.Products, dbo.Sales, and Views. The SQLQuery1.sql window on the right contains four CREATE TABLE statements:

```
-----Customers Table-----
CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY,
    Name NVARCHAR(100),
    Gender NVARCHAR(10),
    Age INT,
    Location NVARCHAR(100),
    JoinDate DATE
);

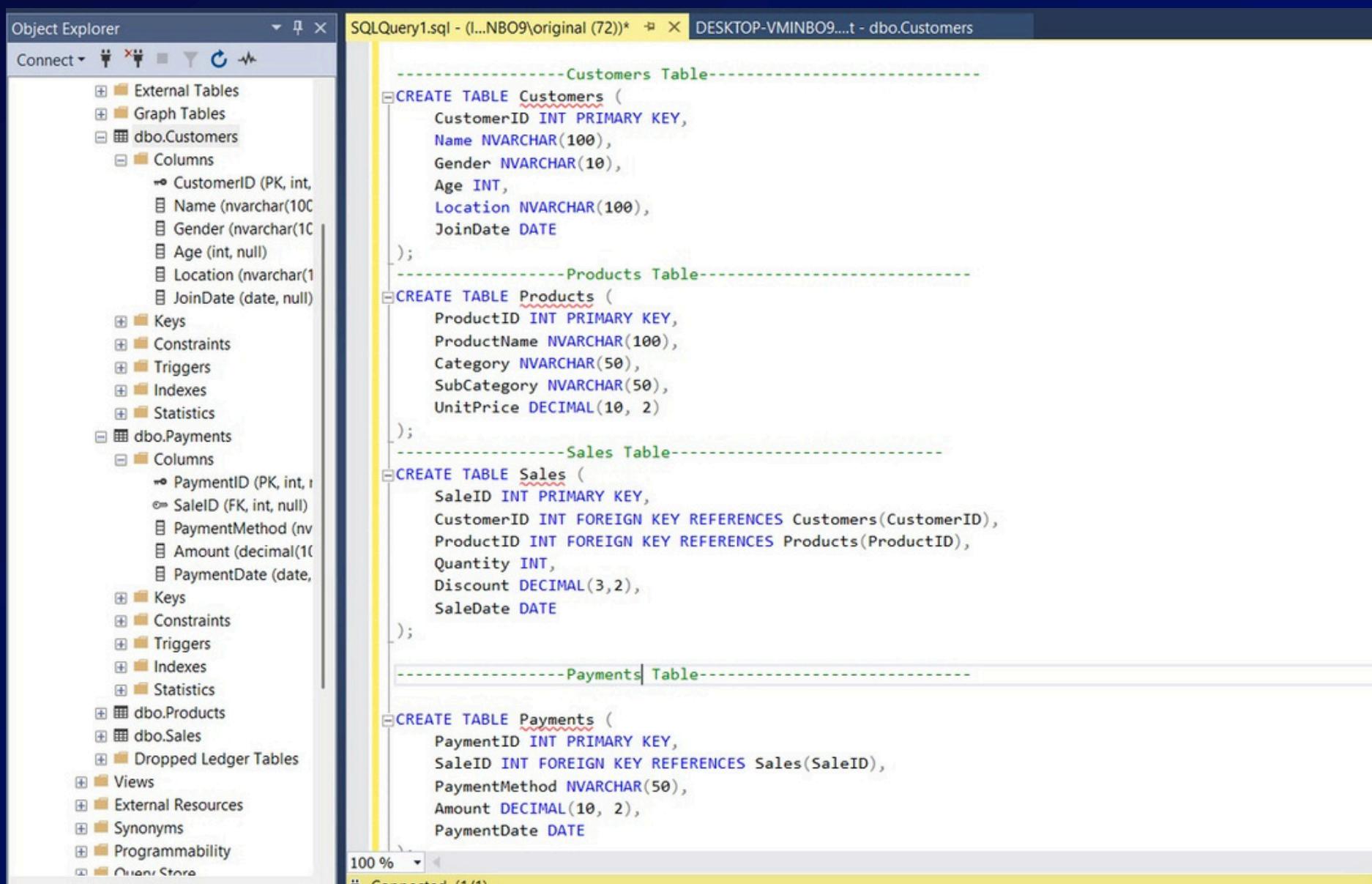
-----Products Table-----
CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName NVARCHAR(100),
    Category NVARCHAR(50),
    SubCategory NVARCHAR(50),
    UnitPrice DECIMAL(10, 2)
);

-----Sales Table-----
CREATE TABLE Sales (
    SaleID INT PRIMARY KEY,
    CustomerID INT FOREIGN KEY REFERENCES Customers(CustomerID),
    ProductID INT FOREIGN KEY REFERENCES Products(ProductID),
    Quantity INT,
    Discount DECIMAL(3,2),
    SaleDate DATE
);

-----Payments Table-----
CREATE TABLE Payments (
    PaymentID INT PRIMARY KEY,
    SaleID INT FOREIGN KEY REFERENCES Sales(SaleID),
    PaymentMethod NVARCHAR(50),
    Amount DECIMAL(10, 2),
    PaymentDate DATE
);
```

First, createtables in  
SQL and use codes  
from mockaroo  
program to generate  
.data

# Insert Data



```
Object Explorer SQLQuery1.sql - (...NBO9\original (72)* DESKTOP-VMINBO9...t - dbo.Customers
Connect ▾

External Tables
Graph Tables
dbo.Customers
  Columns
    CustomerID (PK, int, Name nvarchar(100), Gender nvarchar(10), Age int, Location nvarchar(100), JoinDate date)
  Keys
  Constraints
  Triggers
  Indexes
  Statistics
  Columns
    PaymentID (PK, int, SaleID (FK, int, null), PaymentMethod nvarchar(50), Amount decimal(10, 2), PaymentDate date)
  Keys
  Constraints
  Triggers
  Indexes
  Statistics
  dbo.Products
  dbo.Sales
  Dropped Ledger Tables
  Views
  External Resources
  Synonyms
  Programmability
  Query Store

CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY,
    Name NVARCHAR(100),
    Gender NVARCHAR(10),
    Age INT,
    Location NVARCHAR(100),
    JoinDate DATE
);

CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName NVARCHAR(100),
    Category NVARCHAR(50),
    SubCategory NVARCHAR(50),
    UnitPrice DECIMAL(10, 2)
);

CREATE TABLE Sales (
    SaleID INT PRIMARY KEY,
    CustomerID INT FOREIGN KEY REFERENCES Customers(CustomerID),
    ProductID INT FOREIGN KEY REFERENCES Products(ProductID),
    Quantity INT,
    Discount DECIMAL(3,2),
    SaleDate DATE
);

CREATE TABLE Payments (
    PaymentID INT PRIMARY KEY,
    SaleID INT FOREIGN KEY REFERENCES Sales(SaleID),
    PaymentMethod NVARCHAR(50),
    Amount DECIMAL(10, 2),
    PaymentDate DATE
);
```

Using data from the  
mocaroo website to  
enter data and save it  
on the server

# Create View

SQLQuery2.sql - (I...NBO9\original (62)\* DESKTOP-VMINBO9...object - Diagram\_0\*

```
CREATE VIEW vw_SalesSummary AS
SELECT
    c.Name AS CustomerName,
    p.ProductName,
    p.Category,
    ROUND(p.UnitPrice * s.Quantity * (1 - s.Discount), 2) AS TotalAmount,
    s.SaleDate
FROM
    Sales s
JOIN Customers c ON s.CustomerID = c.CustomerID
JOIN Products p ON s.ProductID = p.ProductID;

select * from vw_SalesSummary
```

Results

	CustomerName	ProductName	Category	TotalAmount	SaleDate
1	Karim Hassan	Sprite Bottle	Electronics	19772.1600	2023-12-15
2	Hanan Nabil	Samsung G...	Electronics	112.9500	2024-04-27
3	Youssef Hassan	iPhone 14	Fashion	14936.9200	2023-08-06
4	Emile Ali	PlayBoy Cla...	Groceries	50000.7400	2024-09-20

SQLQuery2.sql - (I...NBO9\original (62)\* DESKTOP-VMINBO9...object - Diagram\_0\*

```
ALTER VIEW vw_RecentSales AS
SELECT
    s.SaleID,
    s.CustomerID,
    c.Name AS CustomerName,
    s.ProductID,
    p.ProductName,
    s.Quantity,
    s.Discount,
    s.SaleDate
FROM
    Sales s
JOIN Customers c ON s.CustomerID = c.CustomerID
JOIN Products p ON s.ProductID = p.ProductID
WHERE
    s.Discount > 0;

Select * from vw_RecentSales
```

Create a view to use in  
Python Phase



# Python Phase

Using Python to clean and  
analyze data using pandas  
numpy - matplotlib - Seaborn  
Streamlit -



# Python Phase

## Phase1: Load Data

```
[38]: import pandas as pd

# قراءة الملفات CSV
df1 = pd.read_csv('df1.csv', delimiter=';')
df2 = pd.read_csv('df2.csv', delimiter=';')
```

```
[39]: df1
```

	SaleID	CustomerID	Name	ProductID	ProductName	UnitPrice	Quantity	Discount	SaleDate
0	1	97	Karim Hassan	2	Sprite Bottle	5005.61	5	0.21	15/12/2023
1	2	13	Hanan Nabil	12	Samsung Galaxy	144.81	1	0.22	27/04/2024
2	3	58	Youssef Hassan	11	iPhone 14	18215.75	1	0.18	06/08/2023
3	4	12	Laila Ali	4	RayBan Glasses	13480.84	5	0.11	29/09/2024
4	5	93	Laila Saeed	1	Dettol Soap	498.72	4	0.18	19/02/2024
...	...	...	...	...	...	...	...	...	...
487	496	9	Karim Khaled	32	Nike Air Max	5469.95	2	0.09	07/08/2023
488	497	25	Nour Ibrahim	35	iPhone 14	20317.29	1	0.14	22/11/2024
489	498	36	Omar Khaled	44	Dell Monitor	7327.62	2	0.18	17/08/2024

# Python Phase

## Phase2: Clean Data And Transform Data

```
[41]: #تحويل التاريخ إلى نوع تاريخي
df1['SaleDate'] = pd.to_datetime(df1['SaleDate'])

# حساب FinalPrice
df1['FinalPrice'] = df1['UnitPrice'] * df1['Quantity'] * (1 - df1['Discount'])

# مerging الجدولين باستخدام الاسم باسم المنتج
merged_df = pd.merge(df1, df2, left_on=['Name', 'ProductName'], right_on=['Name', 'Product'], how='left')

# إنشاء المجموعات بناءً على الأعمار
bins = [0, 17, 25, 35, 45, 60, 100]
labels = ['<18', '18-25', '26-35', '36-45', '46-60', '60+']
merged_df['AgeGroup'] = pd.cut(merged_df['Age'], bins=bins, labels=labels, right=True)

C:\Users\original\AppData\Local\Temp\ipykernel_8220\4046797612.py:2: UserWarning: Parsing dates in %d/%m/%Y format when dayfirst=False (the default) was
specified. Pass `dayfirst=True` or specify a format to silence this warning.
  df1['SaleDate'] = pd.to_datetime(df1['SaleDate'])
```

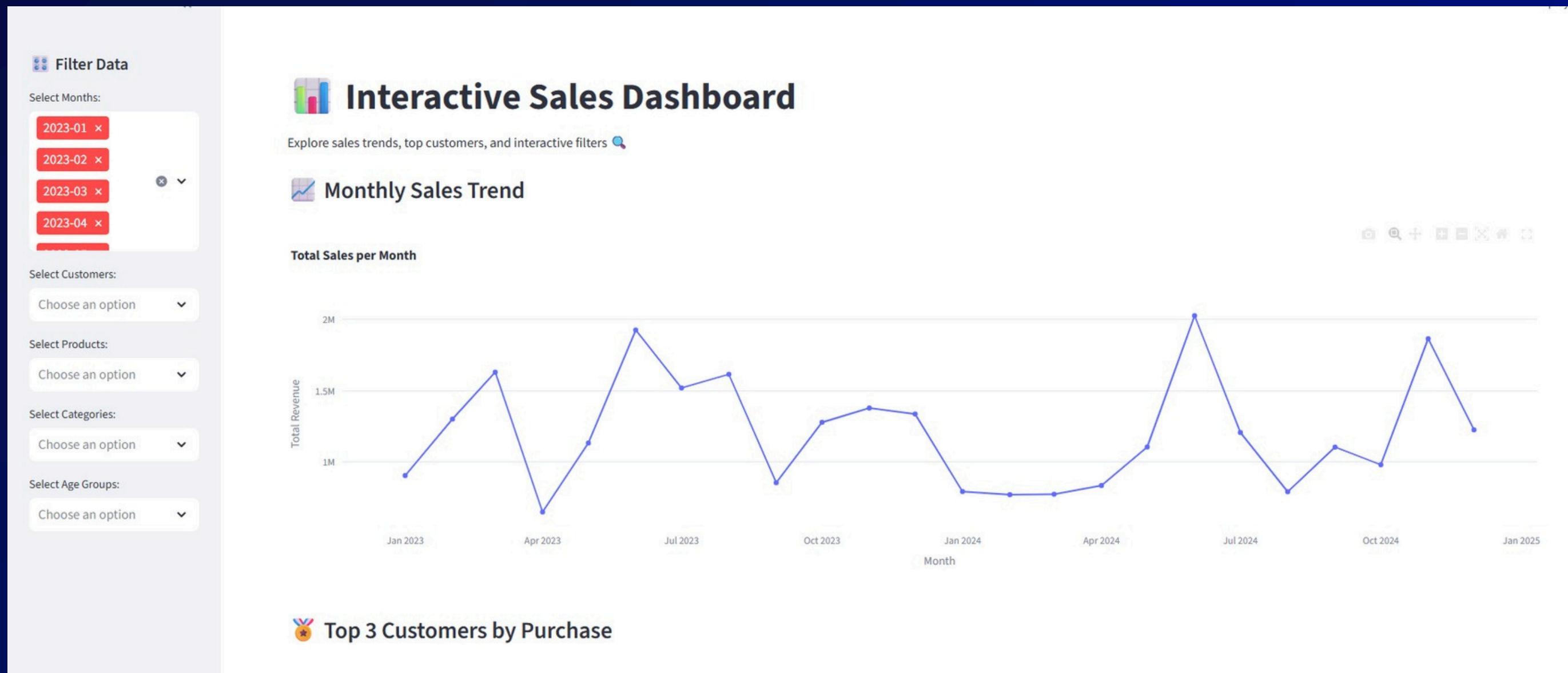
# Python Phase

```
[57]: # إعداد المظهر العام للرسم
sns.set(style='whitegrid')

# رسم اتجاه المبيعات الشهري (Sales Trend Over Months)
plt.figure(figsize=(12, 6)) # حجم الرسم
sns.lineplot(data=monthly_sales, x='Month', y='FinalPrice', marker='o') # رسم الخط
plt.title('Sales Trend Over Months') # عنوان الرسم
plt.ylabel('Total Revenue') # Y تسمية المحور
plt.xlabel('Month') # X تسمية المحور
plt.xticks(rotation=45) # تدوير أسماء الشهور
plt.tight_layout() # تنسيق المسافات
plt.show() # عرض الرسم
```



# Streamlit View





# Streamlit View

Deploy

### Raw Data Table

	SaleDate	Name	ProductName	Category	AgeGroup	FinalPrice	Month
0	2023-08-06 00:00:00	Youssef Hassan	iPhone 14	Fashion	36-45	14936.915	2023-08
4	2023-08-06 00:00:00	Youssef Hassan	iPhone 14	Fashion	36-45	14936.915	2023-08
5	2023-08-06 00:00:00	Youssef Hassan	iPhone 14	Electronics	26-35	14936.915	2023-08
6	2024-09-29 00:00:00	Laila Ali	RayBan Glasses	Groceries	46-60	59989.738	2024-09
7	2024-02-19 00:00:00	Laila Saeed	Dettol Soap	Electronics	46-60	1635.8016	2024-02
8	2023-05-20 00:00:00	Omar Mahmoud	Samsung Galaxy	Electronics	26-35	425.7414	2023-05
9	2023-05-20 00:00:00	Omar Mahmoud	Samsung Galaxy	Electronics	36-45	425.7414	2023-05
10	2023-05-20 00:00:00	Omar Mahmoud	Samsung Galaxy	Groceries	36-45	425.7414	2023-05
11	2023-05-20 00:00:00	Omar Mahmoud	Samsung Galaxy	Electronics	36-45	425.7414	2023-05
12	2023-04-28 00:00:00	Ahmed Osman	RayBan Glasses	Groceries	46-60	64169.2755	2023-04
13	2024-07-29 00:00:00	Hanan Ali	iPhone 14	Electronics	26-35	38602.851	2024-07

# Streamlit View

[Download Filtered CSV](#)

Deploy :

## Customer Details

Select a customer to view details:

Mohammed Atta

### Mohammed Atta

- Total Orders: 97
- Total Purchases: 3,172,175.01 EGP
- Top Purchased Products:

	Product	Count
0	Samsung Galaxy	25
1	iPhone 14	16
2	Dettol Soap	16
3	Pepsi Can	9
4	Oreo Pack	9

# Power bi Phase

Pages Required:

- Sales Overview
- Customer Insights
- Product Trends

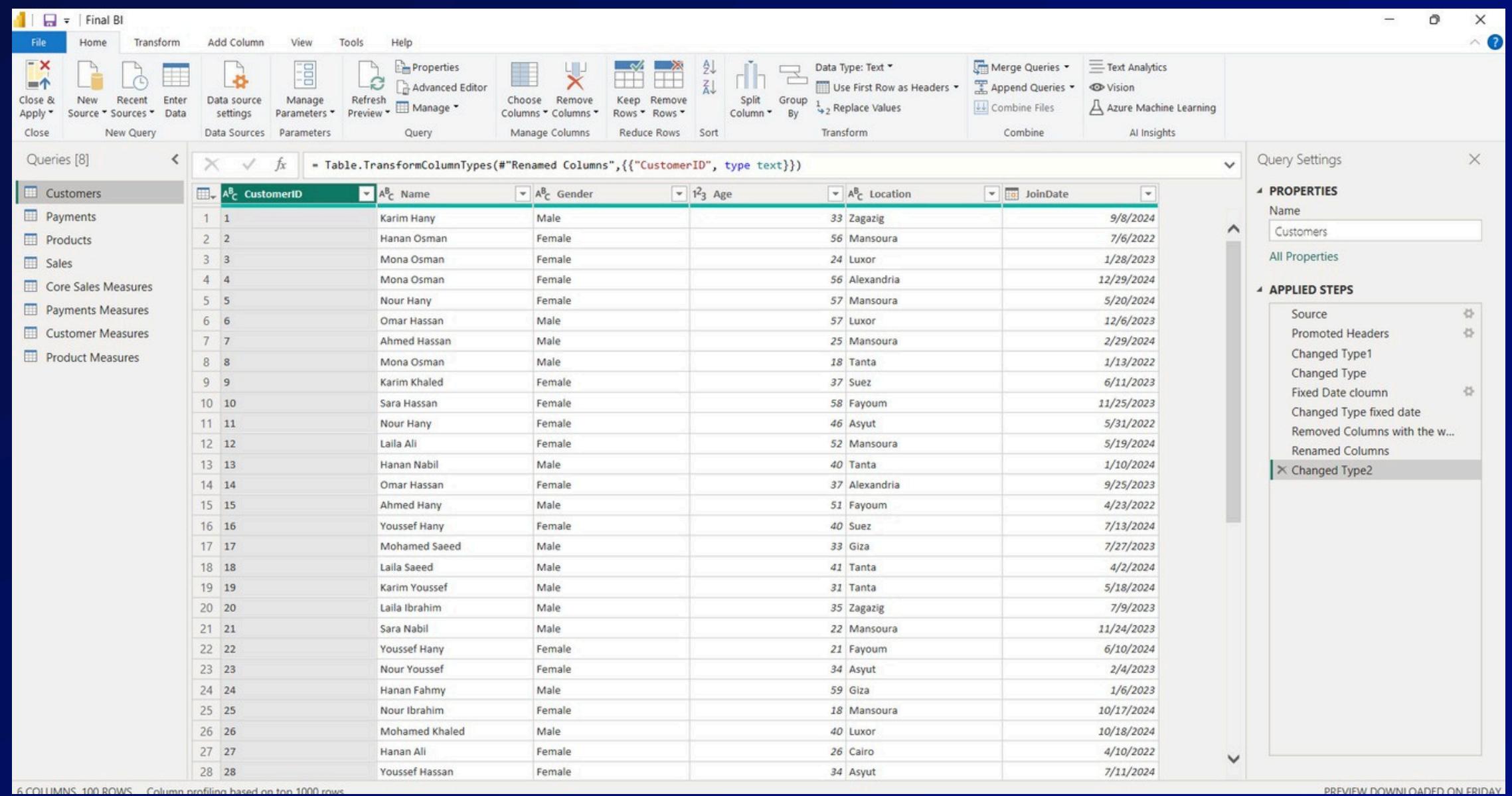
# Power bi Phase

Power BI include:

- Tooltips
- Drilldown + Drillthrough
- Bookmarks
- Slicers (date, category, gender)
- Navigation buttons
- Titles for charts & pages

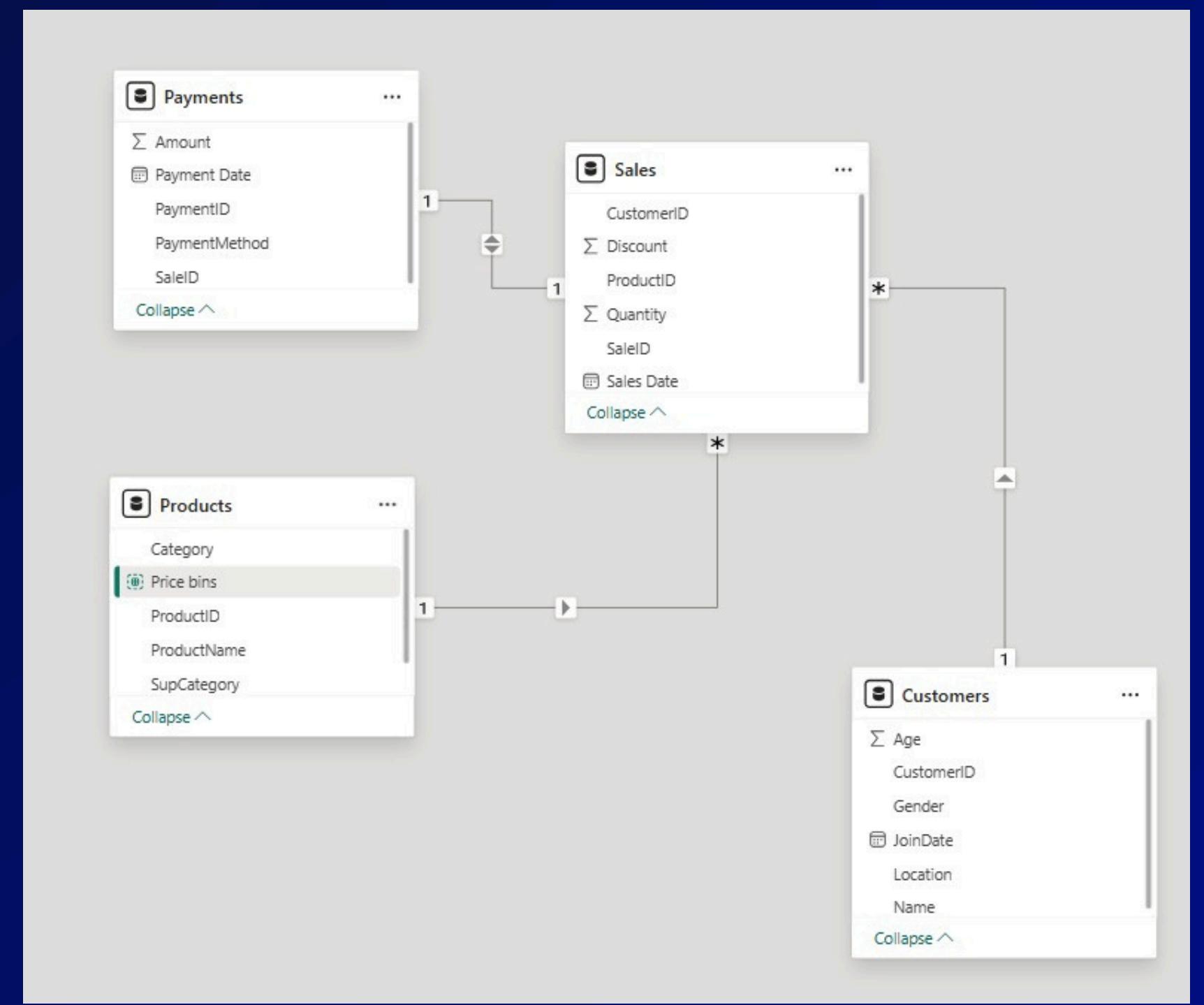
# first Phase

## Clean Data By Use Power Quary



# Second Phase

## Setting relationships between tables



# Third Phase

## Sales Overview Dashboard



# Third Phase

## Customer Insights Dashboard



# Third Phase

## Product Trends Dashboard





**MEC Academy**



**Thank you for  
listening!**