

E-Commerce Data Analysis and Business Intelligence

Prepared By:

- Aya Eid Mahmoud
- Mohamed Nabil Saber
- Arwa Adel Saad
- Ahmed Khaled Mohamed
- Noha Belal Ibrahim

Under Supervision:

Information Technology Institute (ITI)

Sohag Branch



Acknowledgement:

First, we would like to thank Allah for helping us to complete this project successfully.

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

Thanks to Information Technology Institute (ITI) especially Sohag Branch for helping us to reach this level of awareness.

We would like to express our deep and sincere gratitude to our branch manager **Eng. Romany** for giving us the opportunity to lead us and providing guidance throughout this project.

We are also especially indebted to expressing our thanks to **Eng. Amr Soliman** for his constant guidance and supervision.

Finally, we would like to express our gratitude to everyone who helped us during the graduation project.

Abstract:

The rapid growth of e-commerce has generated massive amounts of transactional and behavioral data, creating new opportunities for data-driven decision-making. In recent years, business intelligence and data analytics techniques have become essential tools for extracting meaningful insights from such data.

This project focuses on analyzing e-commerce data to identify customer purchasing patterns, monitor sales performance, and detect trends that can support strategic business decisions.

By integrating data from multiple sources, performing data cleansing and transformation, and applying advanced visualization techniques, the analysis provides actionable insights that can help improve marketing strategies, optimize inventory management, and enhance customer satisfaction.

Table of content

Chapter01.Introduction:	11
1.1 The Importance of Data in E-Commerce:	12
1.2 Project Overview and Objectives :	14
1.3 Target Audience and Scope :	17
Chapter02.Data Preparation:	19
2.1 Entity RelationShip Diagram (ERD):	20
2.2 Mapping:	25
2.3 Creation OLTP Tables:	27
2.4 Implementation:	34
2.5 Business Validation:	34
Chapter03.SSIS Implementation:	48
3.1 Overview of SSIS :	49

3.2 SSIS Architecture:	50
3.3 Data Flow Design:	51
3.3.1 Customer Dimension:	52
3.3.2 Product Dimension:	53
3.3.3 ShipMethod Dimension:	53
3.3.4 Fact Sales:.....	54
3.3.5 Fact CartActivity:	55
3.3.6 Fact Payment:	56
3.3.7 Fact Return:	56
3.3.8 Fact review:	57
Chapter04.Data Modeling(SSAS):	59
4.1 Overview of SSAS:	60
4.2 SSAS Architecture:	61
4.3 Data Model Design:	62
4.3.1 Sales Cube:	63
4.3.2 Payment Cube:	64
4.3.3 ShipMethod Cube:	64

4.3.4 Return Cube:	65
4.3.5 CartActivity Cube:	65
Chapter05.Reporting With(SSRS):	66
5.1 Overview of SSRS:	67
5.2 Developed Reports:	67
5.2.1 Selling Product Report:	68
5.2.2 Cart Report:	69
5.2.3 Customer Report:	70
5.2.4 Product Report :.....	71
5.2.5 Return Report :.....	71
Chapter 06. Visualization.....	72
6.1 Power BI:	73
6.2 Tablue:	82
Chapter07: Conclusion:	84
8.1 Future scope	85
8.2 Conclusions	86

References	87
-------------------------	-----------

List of Figure:

Figure 1:	20
Figure 2:	25
Figure 3:	33
Figure 4:	52
Figure 5:	53
Figure 6:	53
Figure 7:.....	54
Figure 8:	55
Figure 9:.....	56
Figure10:	56
Figure11:.....	57
Figure12:.....	58
Figure13:	63
Figure14:	64
Figure15:.....	64
Figure16:.....	65

Figure17:.....	65
Figure18:.....	68
Figure 19:	68
Figure 20:	69
Figure 21:	70
Figure 22:	71
Figure 23:	71
Figure 24:	73
Figure 25:.....	74
Figure 26:	74
Figure 27:.....	75
Figure28:	75
Figure29:.....	76
Figure30:.....	76
Figure31:	77
Figure32:	77
Figure33:.....	78
Figure34:.....	78

Figur35:.....	81
Figure36:.....	82
Figure 37:	83
Figure 38:	83

Chapter 01

Introduction

1.1The Importance of Data in E-Commerce

In the rapidly evolving landscape of electronic commerce, data has emerged as the most valuable asset for businesses striving to gain a competitive edge and achieve sustainable growth. The sheer volume, velocity, and variety of data generated daily from online sales, customer interactions, product browsing, and marketing campaigns present both a significant challenge and an unparalleled opportunity. Effectively harnessing this data is no longer a luxury but a fundamental necessity for understanding market dynamics, optimizing operations, and personalizing customer experiences. Without robust data analysis capabilities, ecommerce businesses risk operating in the dark, making decisions based on intuition rather than evidence, and ultimately falling behind competitors who leverage data to their advantage. The digital nature of e-commerce transactions means that every click, every purchase, every product view, and every customer interaction leaves a digital footprint. This rich tapestry of data, when properly collected, processed, and analyzed, can reveal profound insights into consumer behavior, purchasing patterns, product performance, and market trends. For instance, analyzing sales data can help identify peak shopping seasons, popular product categories, and the effectiveness of promotional campaigns.

Understanding customer data allows businesses to segment their audience, personalize recommendations, and tailor marketing messages, leading to increased customer satisfaction and loyalty.

Operational data, such as inventory levels and shipping times, can be analyzed to optimize supply chain efficiency and reduce costs. In essence, data provides the intelligence required to navigate the complexities of the online marketplace, enabling businesses to adapt quickly to changing conditions and proactively address challenges.

Moreover, the competitive intensity within the e-commerce sector necessitates a data-driven approach. With low barriers to entry, new online stores are constantly emerging, making it crucial for established players to differentiate themselves through superior customer experiences and operational excellence. Data analysis empowers businesses to identify their unique selling propositions, understand customer pain points, and innovate their offerings. It allows for the precise measurement of marketing campaign effectiveness, ensuring that resources are allocated to channels and strategies that yield the highest return on investment. Furthermore, predictive analytics, powered by historical data, can forecast future trends, anticipate demand, and identify potential risks, enabling proactive strategic planning rather than reactive problem-solving. The ability to transform raw data into actionable insights is therefore a cornerstone of modern e-commerce success, driving everything from product development and pricing strategies to customer service and marketing initiatives.

1.2 Project Overview and Objectives

This documentation outlines a practical model for transforming raw e-commerce data into interactive dashboards and comprehensive reports that empower decision-makers to understand current business situations, evaluate performance, and forecast future trends. The project leverages a suite of Microsoft Business Intelligence (BI) tools—SQL Server Integration Services (SSIS), SQL Server Analysis Services (SSAS), and SQL Server Reporting Services (SSRS)—alongside modern visualization platforms

like Power BI and Tableau. The primary objective is to demonstrate an end-to-end data analytics solution for an e-commerce ecosystem, from data generation and preparation to advanced data modeling, reporting, and interactive visualization. The core objectives of this project are multifaceted:

1. Data Simulation and Collection:

To simulate a realistic e-commerce environment by generating diverse and semi-realistic

datasets using SQL scripts and artificial intelligence tools. This ensures that the data reflects real-world scenarios in terms of data types, table relationships, and record counts, providing a robust foundation for analysis.

2. ETL Process Implementation:

To design and implement a robust Extract, Transform, Load (ETL) process using SSIS.

This involves extracting data from the simulated operational database, performing necessary cleaning and transformations (e.g., handling orphan keys, enum checks, value checks, uniqueness, and date logic validation), and loading the refined data into a data warehouse. The goal is to ensure data quality, consistency, and readiness for analytical processing.

3. Data Modeling for Analytical Processing: To develop a comprehensive dimensional data model within a data warehouse, optimized for business intelligence. This includes designing dimension tables (e.g., Customer, Product, ShipMethod) and fact tables (e.g., Sales, Payment, CartActivity, Return, Review). Furthermore, to build OLAP cubes using SSAS (Sales Cube, Payment Cube, ShipMethod Cube, Return Cube, ActivityCart Cube) to enable fast, multidimensional analysis and preaggregation of key metrics.

4. Reporting and Visualization Development: To create a variety of reports and interactive dashboards using SSRS, Power BI, and Tableau. This involves designing detailed operational reports (e.g., Sales Reports, Product Reports, Cart Reports, Customer Reports, Return Reports) with SSRS, and developing

interactive dashboards with Power BI and Tableau that feature key performance indicators (KPIs), drilling through capabilities, and tooltips. The aim is to provide clear, intuitive visual solutions that facilitate performance evaluation and strategic decision-making.

5. Technology Integration and Demonstration: To showcase the seamless integration of various tools and technologies, including Python for data generation, SQL Server for database management, SSIS for ETL, SSAS for data modeling, SSRS for reporting, and Power BI/Tableau for visualization. This demonstrates a holistic approach to e-commerce data analytics, highlighting how different components work together to deliver comprehensive insights.

By achieving these objectives, this project aims to provide a practical blueprint for e-commerce businesses to transform their raw

data into actionable intelligence, enabling them to understand their current standing, identify opportunities, and forecast future trends with greater accuracy and efficiency.

1.3 Target Audience and Scope

This documentation is intended for a broad audience involved in the e-commerce ecosystem, ranging from business analysts and data scientists to IT professionals, project managers, and executive decision-makers. Each section is designed to provide relevant information and insights tailored to the needs of different stakeholders. For instance, technical sections on ETL and data modeling

will be particularly useful for data engineers and developers, while discussions on reporting and visualization will appeal to business users and analysts. Executive summaries and strategic implications will cater to management and leadership, providing high-level overviews and actionable recommendations. The scope of this documentation is comprehensive, covering the entire lifecycle of e-commerce data, from its initial generation and collection to its transformation, modeling, analysis, and visualization. It delves into the theoretical underpinnings of data warehousing and business intelligence, alongside practical implementation details using specific Microsoft BI tools and popular

visualization platforms. While the project uses a simulated e-commerce database for demonstration purposes, the principles, methodologies, and best practices discussed are directly applicable to real-world e-commerce environments with live data. The documentation aims to be a valuable resource for anyone looking

to understand, implement, or optimize data analytics solutions within the e-commerce domain.

Specifically, the documentation will cover:

E-commerce Data Fundamentals: An overview of typical e-commerce data sources and key entities.

Data Preparation and ETL: Detailed explanation of data cleaning, transformation, and the SSIS process.

Data Modeling: Concepts of dimensional modeling, star/snowflake schemas, and SSAS cube design.

Reporting and Visualization: Practical application of SSRS, Power BI, and Tableau for creating reports and interactive dashboards.

Tools and Technologies: A summary of all technologies utilized in the project and their roles.

Case Studies/Examples: Illustrative examples and screenshots where applicable to clarify concepts.

Best Practices: Recommendations for effective data management, analysis, and reporting in e-commerce. This documentation does not aim to be a step-by-step tutorial for each software tool but rather a conceptual and architectural guide to building a robust e-commerce data analytics solution.

It assumes a basic understanding of database concepts and business intelligence principles. The goal is to provide a holistic view of the process, enabling readers to grasp the interconnectedness of different components and apply these insights to their specific e-commerce data challenges.

Chapter02

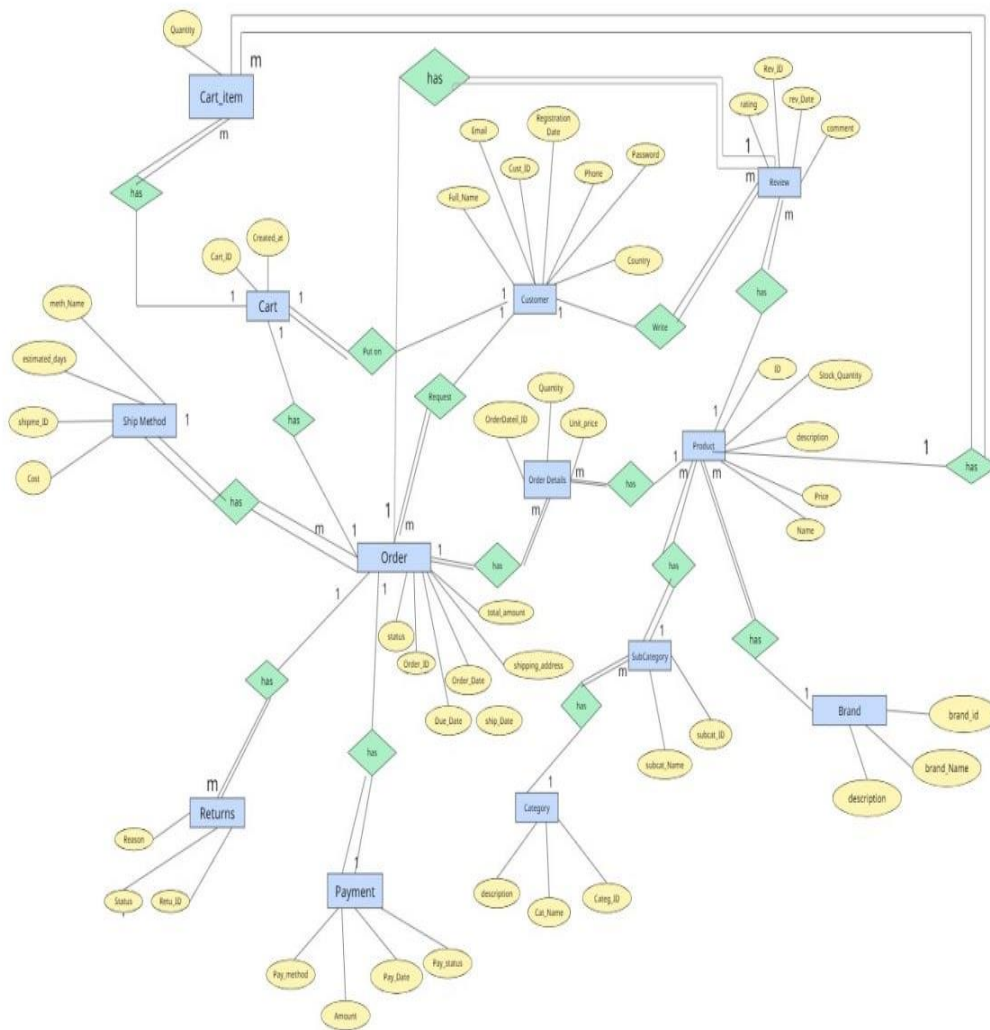
Data Preparation

2.1 Entity Relationship Diagram (ERD):

This ERD represents the database structure of an online retail shop. It includes **entities**, **attributes**, and **relationships** between them.

Entities and Attributes

Online Retail Shop (E-Commerce)



Figure(1)

Customer

- **Purpose:** Stores registered customer information.
- **Attributes:**
 - Cust_ID – Customer unique identifier
 - Full_Name – Full name of the customer
 - Email – Email address
 - Password – Account password
 - Phone – Contact number
 - Country – Country of residence
 - Registration_Date – Date of account registration

Cart

- **Purpose:** Represents the shopping cart for each customer.
- **Attributes:**
 - Cart_ID – Unique identifier for the cart
 - Created_at – Date and time the cart was created

Cart_Item

- **Purpose:** Stores the products inside a customer's cart.
- **Attributes:**
 - Quantity – Number of units for the product in the cart

Product

- **Purpose:** Represents the products available for sale.
- **Attributes:**
 - ID – Product unique identifier
 - Name – Product name
 - Description – Product details
 - Price – Unit price
 - Stock_Quantity – Available stock

Brand

- **Purpose:** Represents the brand of a product.
- **Attributes:**
 - Brand_ID – Unique brand identifier
 - Brand_Name – Name of the brand
 - Description – Brand details

Category & SubCategory

- **Purpose:** Classifies products into main and sub-categories.
- **Category Attributes:**
 - Cat_ID – Category unique identifier
 - Cat_Name – Category name
 - Description – Category description
- **SubCategory Attributes:**
 - SubCat_ID – Sub-category unique identifier
 - SubCat_Name – Sub-category name

Order

- **Purpose:** Represents customer purchase orders.
- **Attributes:**
 - Order_ID – Order unique identifier
 - Status – Order status
 - Order_Date – Date of order creation
 - Due_Date – Expected delivery date
 - Ship_Date – Actual shipping date
 - Shipping_Address – Delivery address
 - Total_Amount – Total order amount

Order_Details

- **Purpose:** Holds detailed product information for each order.
- **Attributes:**
 - OrderDetail_ID – Unique detail identifier
 - Quantity – Number of units ordered
 - Unit_Price – Price per unit

Payment

- **Purpose:** Stores payment information.
- **Attributes:**
 - Pay_Method – Payment method (e.g., credit card, PayPal)
 - Pay_Date – Payment date
 - Pay_Status – Payment status
 - Amount – Paid amount

Returns

- **Purpose:** Records returned products.
- **Attributes:**
 - Return_ID – Return unique identifier
 - Reason – Reason for return
 - Status – Return status

Ship_Method

- **Purpose:** Defines shipping methods.
- **Attributes:**
 - Shipmethod_ID – Shipping method unique identifier
 - Meth_Name – Method name
 - Estimated_Days – Estimated delivery days
 - Cost – Shipping cost

Review

- **Purpose:** Stores customer feedback for products.
- **Attributes:**
 - Review_ID – Review unique identifier
 - Rating – Product rating
 - Review_Date – Date of review submission
 - Comment – Review content

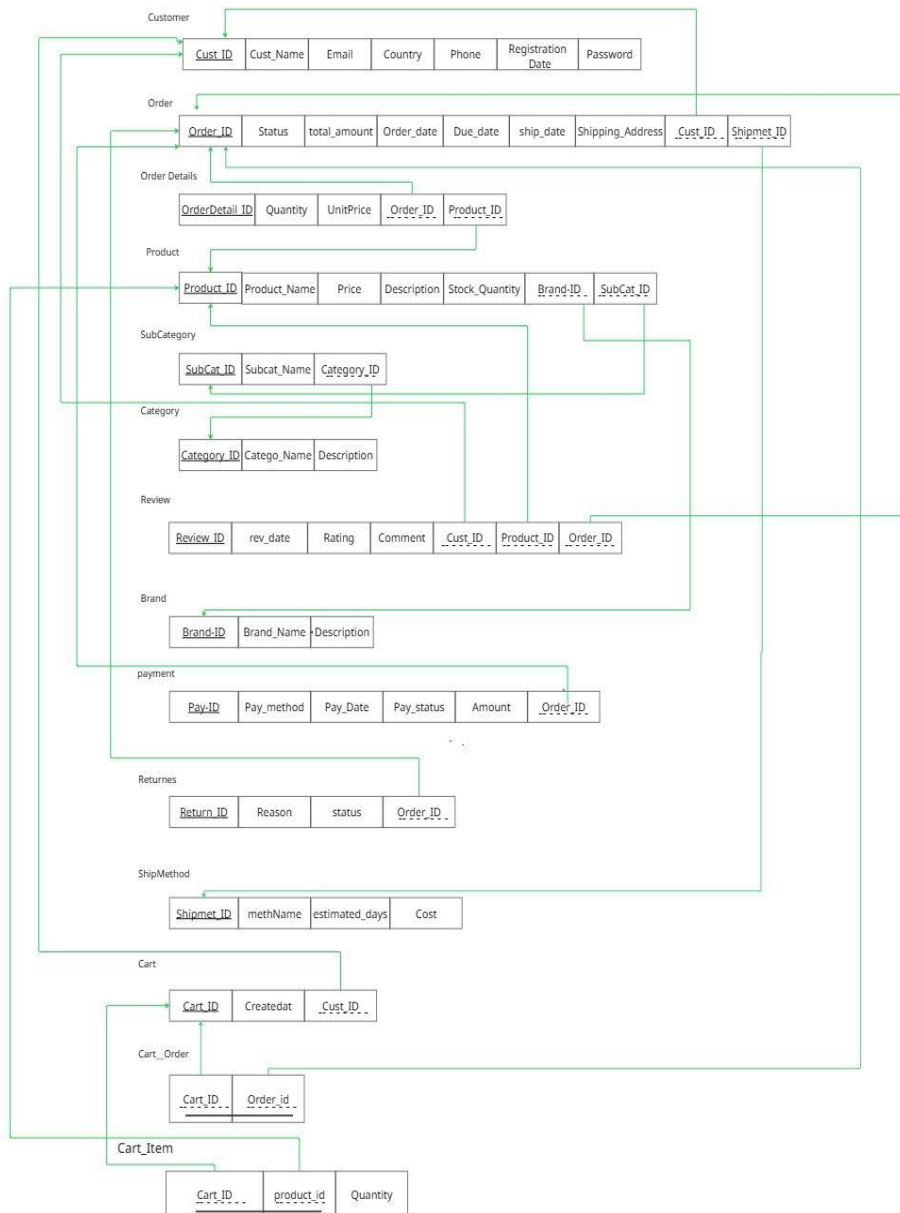
2. Relationships

- **Customer ↔ Cart:** A customer can have multiple carts (**1 to m**).
- **Cart ↔ Cart_Item:** A cart can contain multiple products (**1 to m**).
- **Cart_Item ↔ Product:** A product can appear in multiple carts (**m to m**).
- **Order ↔ Customer:** A customer can place multiple orders (**1 to m**).
- **Order ↔ Order_Details ↔ Product:** Each order can contain multiple products, and each product can appear in multiple orders (**m to m**).
- **Order ↔ Payment:** An order can have one or more payments (**1 to m**).
- **Order ↔ Returns:** An order can have multiple returns (**1 to m**).
- **Order ↔ Ship_Method:** Each order has one shipping method (**1 to 1**).
- **Product ↔ Review ↔ Customer:** Customers can write reviews for products, and products can have multiple reviews (**m to m**).
- **Product ↔ Brand:** A product belongs to one brand (**m to 1**).
- **Product ↔ SubCategory ↔ Category:** A product belongs to one sub-category, and a sub-category belongs to one main category (**m to 1**).

2.2 Mapping :

The mapping diagram below shows the relationship between the ERD entities and their corresponding database tables.

Mapping



Figure(2)

- **Customer** table contains customer information (name, email, country, phone number, registration date, password) and is linked to **Order**, **Review**, and **Cart** through the `Cust_ID` field.
- **Order** table contains order information (status, total amount, order date, due date, ship date, shipping address) and is linked to **Customer**, **ShipMethod**, **Payment**, **Returns**, and **Order_Details**.
- **Order_Details** table contains the products within each order (quantity, unit price) and is linked to **Order** and **Product**.
- **Product** table contains product data (name, price, description, stock quantity) and is linked to **Brand**, **SubCategory**, **Review**, and **Order_Details**.
- **SubCategory** and **Category** are used to classify products in a hierarchical structure, with the `Category_ID` field linking main categories to subcategories.
- **Review** table contains customer reviews (rating, review date, comment) and is linked to **Customer**, **Product**, and **Order**.
- **Brand** table contains brand details and is linked to **Product** through the `Brand_ID` field.
- **Payment** table contains payment information (payment method, payment date, payment status, amount) and is linked to **Order** through the `Order_ID` field.
- **Returns** table contains return information (reason, status) and is linked to **Order** through the `Order_ID` field.

- **ShipMethod** table contains available shipping methods (method name, estimated delivery days, cost) and is linked to **Order** through the `Shipment_ID` field.
- **Cart** table contains cart details and creation date, and is linked to **Customer** and **Cart_Order**.
- **Cart_Order** table links carts to orders.
- **Cart_Item** table contains the products and quantities in a cart, and is linked to **Cart** and **Product**.

2.3 Creation OLTP Tables:

```

CREATE DATABASE E_Commerce;
GO
USE E_Commerce;
GO
CREATE SCHEMA users;
GO
CREATE SCHEMA products;
GO
CREATE SCHEMA cart;
GO
CREATE SCHEMA orders;
GO

CREATE TABLE users.Customer (
    Cust_ID INT PRIMARY KEY IDENTITY(1,1),
    Cust_Name NVARCHAR(100) NOT NULL,
    Email NVARCHAR(100) UNIQUE,
    Address NVARCHAR(200),
    Phone NVARCHAR(20) UNIQUE,

```

```

        Registration_Date DATE DEFAULT GETDATE(),
        Password NVARCHAR(100) NOT NULL
    );
GO

CREATE TABLE products.Category (
    Category_ID INT PRIMARY KEY IDENTITY(1,1),
    Category_Name NVARCHAR(100) NOT NULL,
    Description NVARCHAR(200)
);

CREATE TABLE products.Brand (
    Brand_ID INT PRIMARY KEY IDENTITY(1,1),
    Brand_Name NVARCHAR(100) NOT NULL,
    Description NVARCHAR(200)
);

CREATE TABLE products.SubCategory (
    SubCat_ID INT PRIMARY KEY IDENTITY(1,1),
    SubCat_Name NVARCHAR(100) NOT NULL,
    Category_ID INT NOT NULL,
    CONSTRAINT FK_SubCategory_Category FOREIGN
KEY (Category_ID)
    REFERENCES products.Category(Category_ID)
);

CREATE TABLE products.Product (
    Product_ID INT PRIMARY KEY IDENTITY(1,1),
    Product_Name NVARCHAR(100) NOT NULL,
    Price DECIMAL(10,2) CHECK (Price >= 0),
    Description NVARCHAR(200),
    Stock_Quantity INT CHECK (Stock_Quantity >=
0),
    Brand_ID INT,

```

```

        SubCat_ID INT,
        FOREIGN KEY (Brand_ID) REFERENCES
products.Brand(Brand_ID),
        FOREIGN KEY (SubCat_ID) REFERENCES
products.SubCategory(SubCat_ID)
);
GO
CREATE TABLE orders.ShipMethod (
    ShipMethod_ID INT PRIMARY KEY IDENTITY(1,1),
    Method_Name NVARCHAR(100) NOT NULL,
    Estimated_Days INT CHECK (Estimated_Days >
0),
    Cost DECIMAL(10,2) CHECK (Cost >= 0)
);

CREATE TABLE orders.Orders (
    Order_ID INT PRIMARY KEY IDENTITY(1,1),
    Cust_ID INT NOT NULL,
    ShipMethod_ID INT NOT NULL,
    Order_Date DATETIME DEFAULT GETDATE(),
    Ship_Date DATETIME,
    Due_Date DATETIME,
    Status NVARCHAR(50) CHECK (Status IN
('Pending', 'Shipped', 'Delivered', 'Cancelled',
'Returned')),
    Total_Amount DECIMAL(10,2) NOT NULL,
    Shipping_Address NVARCHAR(255) NOT NULL,
    FOREIGN KEY (Cust_ID) REFERENCES
users.Customer(Cust_ID),
    FOREIGN KEY (ShipMethod_ID) REFERENCES
orders.ShipMethod(ShipMethod_ID)
);

CREATE TABLE orders.OrderDetails (

```

```

        OrderDetail_ID INT PRIMARY KEY IDENTITY(1,1),
        Quantity INT CHECK (Quantity > 0),
        UnitPrice DECIMAL(10,2) CHECK (UnitPrice >=
0),
        Order_ID INT NOT NULL,
        Product_ID INT NOT NULL,
        FOREIGN KEY (Order_ID) REFERENCES
orders.Orders(Order_ID),
        FOREIGN KEY (Product_ID) REFERENCES
products.Product(Product_ID)
);

```

```

CREATE TABLE orders.Returned (
    Return_ID INT PRIMARY KEY IDENTITY(1,1),
    Order_ID INT NOT NULL,
    Product_ID INT NOT NULL,
    Reason NVARCHAR(500),
    Status NVARCHAR(50) NOT NULL CHECK (Status IN
('Requested', 'Approved', 'Rejected',
'Processed')),
    Requested_At DATETIME DEFAULT GETDATE(),
    Processed_At DATETIME,
    FOREIGN KEY (Order_ID) REFERENCES
orders.Orders(Order_ID),
    FOREIGN KEY (Product_ID) REFERENCES
products.Product(Product_ID)
);

```

```

CREATE TABLE orders.Payment (
    Payment_ID INT PRIMARY KEY IDENTITY(1,1),
    Order_ID INT NOT NULL,
    Payment_Date DATETIME DEFAULT GETDATE(),

```

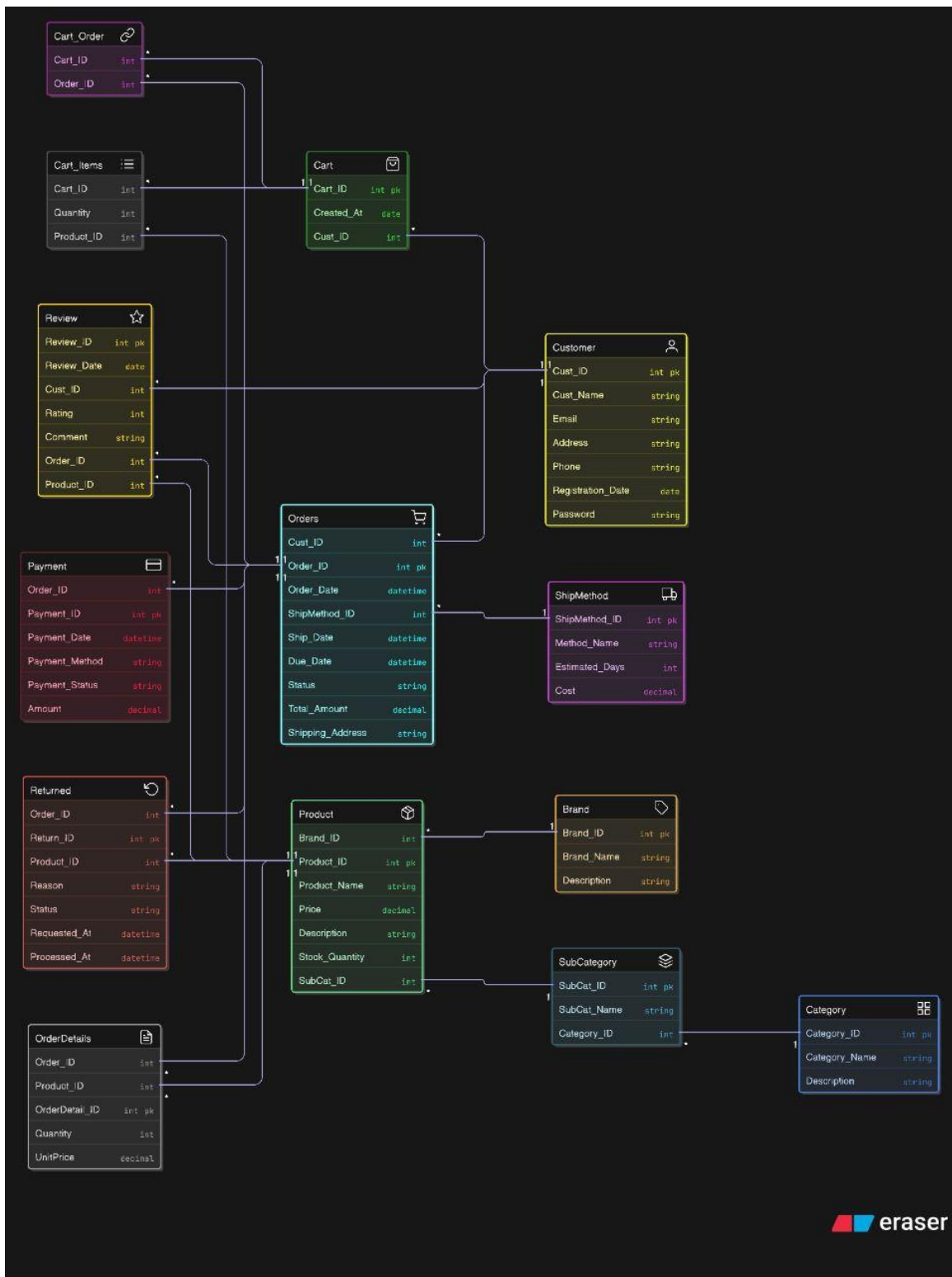
```

        Payment_Method NVARCHAR(50) NOT NULL CHECK
(Payment_Method IN ('Credit Card', 'Debit Card',
'PayPal', 'Cash on Delivery', 'Bank Transfer')),
        Payment_Status NVARCHAR(50) NOT NULL CHECK
(Payment_Status IN ('Pending', 'Completed',
'Failed', 'Refunded')),
        Amount DECIMAL(10,2) NOT NULL,
        FOREIGN KEY (Order_ID) REFERENCES
orders.Orders(Order_ID)
);
GO
CREATE TABLE users.Review (
    Review_ID INT PRIMARY KEY IDENTITY(1,1),
    Review_Date DATE DEFAULT GETDATE(),
    Rating INT CHECK (Rating BETWEEN 1 AND 5),
    Comment NVARCHAR(300),
    Cust_ID INT,
    Product_ID INT,
    Order_ID INT NULL,
    FOREIGN KEY (Cust_ID) REFERENCES
users.Customer(Cust_ID),
    FOREIGN KEY (Product_ID) REFERENCES
products.Product(Product_ID),
    FOREIGN KEY (Order_ID) REFERENCES
orders.Orders(Order_ID)
);
GO
CREATE TABLE cart.Cart (
    Cart_ID INT PRIMARY KEY IDENTITY(1,1),
    Created_At DATE DEFAULT GETDATE(),
    Cust_ID INT NOT NULL,
    FOREIGN KEY (Cust_ID) REFERENCES
users.Customer(Cust_ID)
);

```

```
CREATE TABLE cart.Cart_Items (  
    Cart_ID INT,  
    Product_ID INT,  
    Quantity INT CHECK (Quantity > 0),  
    PRIMARY KEY (Cart_ID, Product_ID),  
    FOREIGN KEY (Cart_ID) REFERENCES  
cart.Cart(Cart_ID),  
    FOREIGN KEY (Product_ID) REFERENCES  
products.Product(Product_ID)  
);
```

```
CREATE TABLE cart.Cart_Order (  
    Cart_ID INT,  
    Order_ID INT,  
    PRIMARY KEY (Cart_ID, Order_ID),  
    FOREIGN KEY (Cart_ID) REFERENCES  
cart.Cart(Cart_ID),  
    FOREIGN KEY (Order_ID) REFERENCES  
orders.Orders(Order_ID)  
);  
GO
```

Figure(3)

2.4 Implementation:

- We have created our own virtual database to simulate an e-commerce system. The data was generated using SQL scripts we designed ourselves, ensuring that it is as realistic as possible in terms of data types, table relationships, and record counts, so that it reflects a real-world scenario that can be analyzed. Logical relationships between tables (such as foreign keys and composite keys) were also taken into account.
- The data was generated using artificial intelligence tools and custom scripts to produce diverse and semi-realistic datasets. We designed the schema and data based on well-known e-commerce database models, with some modifications to suit our specific analysis objectives.

2.5 Business Validation:

Data validation and quality assurance are continuous processes throughout the ETL pipeline, not just a final step. They ensure that the data loaded into the data warehouse is accurate, consistent, complete, and reliable. The presentation specifically mentions several key validation methods:

1. Orphan Keys and Referential Integrity

Concept: Referential integrity ensures that relationships between tables remain consistent.

An orphan key is a foreign key in a fact table that does not have a corresponding primary key in the dimension table. This can happen if a dimension record is deleted from the source system but the

corresponding fact records are not, or if a fact record is created with an invalid dimension key.

Validation in SSIS:

The Lookup Transformation is the primary tool for validating referential integrity in SSIS. When loading a fact table, a lookup can be performed against each dimension table to ensure that the foreign

keys in the fact record have a matching primary key in the dimension table.

Rows that do not find a match (i.e., have orphan keys) can be redirected to an error output for further investigation and correction. This prevents loading invalid data into the data warehouse and maintains the integrity of the dimensional model.

2. Enum and Value Checks

Concept: Enum (enumeration) checks validate that the values in a column belong to a predefined set of allowed values.

For example, a PaymentStatus column might only allow values like 'Paid', 'Pending', 'Failed', or 'Refunded'. Value checks ensure that data falls within a valid range or meets specific criteria (e.g., UnitPrice must be a positive number).

Validation in SSIS: The Conditional Split Transformation can be used to implement enum and value checks.

By defining conditions that check for valid values or ranges, rows that do not meet the criteria can be routed to a separate output for error handling.

For example, a condition could be `PaymentStatus == "Paid" || PaymentStatus == "Pending" || PaymentStatus == "Failed" || PaymentStatus == "Refunded"` . Any rows that do not satisfy this condition would be considered invalid. Similarly, a condition like `UnitPrice > 0` can be used for value checks.

3. Uniqueness and Date Logic Validation

Concept: Uniqueness checks ensure that there are no duplicate records where they are not allowed (e.g., duplicate CustomerID in the Dim-Customer table). Date logic validation ensures that dates are logical and consistent (e.g., OrderDate must be before ShipDate , BirthDate cannot be in the future).

Validation in SSIS: The Sort Transformation with the option to remove duplicate rows can be used to enforce uniqueness. Alternatively, the Aggregate Transformation with a COUNT operation can be used to identify duplicate keys. For date logic validation, the Derived Column Transformation or Script Component can be used to apply custom logic.

For example, an expression like `ShipDate > OrderDate` can be used to create a flag for valid records, and then a Conditional Split can be used to separate valid and invalid rows.

By implementing these data validation and quality assurance techniques within the SSIS ETL process, e-commerce businesses can ensure that their data warehouse is populated with high quality, trustworthy data, which is essential for accurate reporting and reliable decision-making

```
-- Data Consistency Check Script for E_Commerce
DB
```

```
-- ===== Orphaned Foreign Keys =====
-- Orders without valid Customers
```

```
SELECT 'Orphan Orders (Customer)' AS Test,
COUNT(*) AS Failures
FROM orders.Orders o
LEFT JOIN users.Customer c ON o.Cust_ID =
c.Cust_ID
WHERE c.Cust_ID IS NULL;
```

```
-- Orders without valid Ship Methods
```

```
SELECT 'Orphan Orders (ShipMethod)', COUNT(*)
FROM orders.Orders o
LEFT JOIN orders.ShipMethod s ON o.ShipMethod_ID
= s.ShipMethod_ID
WHERE s.ShipMethod_ID IS NULL;
```

```
-- OrderDetails with missing Orders
```

```
SELECT 'Orphan OrderDetails (Order)', COUNT(*)
FROM orders.OrderDetails od
LEFT JOIN orders.Orders o ON od.Order_ID =
o.Order_ID
WHERE o.Order_ID IS NULL;
```

```
-- OrderDetails with missing Products
```

```
SELECT 'Orphan OrderDetails (Product)', COUNT(*)
FROM orders.OrderDetails od
LEFT JOIN products.Product p ON od.Product_ID =
p.Product_ID
WHERE p.Product_ID IS NULL;
```

-- Returned with missing Orders

```
SELECT 'Orphan Returned (Order)', COUNT(*)  
FROM orders.Returned r  
LEFT JOIN orders.Orders o ON r.Order_ID =  
o.Order_ID  
WHERE o.Order_ID IS NULL;
```

-- Returned with missing Products

```
SELECT 'Orphan Returned (Product)', COUNT(*)  
FROM orders.Returned r  
LEFT JOIN products.Product p ON r.Product_ID =  
p.Product_ID  
WHERE p.Product_ID IS NULL;
```

-- Payments with missing Orders

```
SELECT 'Orphan Payments (Order)', COUNT(*)  
FROM orders.Payment p  
LEFT JOIN orders.Orders o ON p.Order_ID =  
o.Order_ID  
WHERE o.Order_ID IS NULL;
```

-- Reviews with missing Customers

```
SELECT 'Orphan Reviews (Customer)', COUNT(*)  
FROM users.Review r  
LEFT JOIN users.Customer c ON r.Cust_ID =  
c.Cust_ID  
WHERE c.Cust_ID IS NULL;
```

-- Reviews with missing Products

```
SELECT 'Orphan Reviews (Product)', COUNT(*)  
FROM users.Review r  
LEFT JOIN products.Product p ON r.Product_ID =  
p.Product_ID
```

```

WHERE p.Product_ID IS NULL;

-- Reviews with missing Orders
SELECT 'Orphan Reviews (Order)', COUNT(*)
FROM users.Review r
LEFT JOIN orders.Orders o ON r.Order_ID =
o.Order_ID
WHERE r.Order_ID IS NOT NULL AND o.Order_ID IS
NULL;

-- Cart with missing Customers
SELECT 'Orphan Cart (Customer)', COUNT(*)
FROM cart.Cart c
LEFT JOIN users.Customer u ON c.Cust_ID =
u.Cust_ID
WHERE u.Cust_ID IS NULL;

-- Cart Items with missing Products
SELECT 'Orphan Cart_Items (Product)', COUNT(*)
FROM cart.Cart_Items ci
LEFT JOIN products.Product p ON ci.Product_ID =
p.Product_ID
WHERE p.Product_ID IS NULL;

-- Cart Items with missing Carts
SELECT 'Orphan Cart_Items (Cart)', COUNT(*)
FROM cart.Cart_Items ci
LEFT JOIN cart.Cart c ON ci.Cart_ID = c.Cart_ID
WHERE c.Cart_ID IS NULL;

-- Cart_Order with missing Cart
SELECT 'Orphan Cart_Order (Cart)', COUNT(*)
FROM cart.Cart_Order co
LEFT JOIN cart.Cart c ON co.Cart_ID = c.Cart_ID

```

```

WHERE c.Cart_ID IS NULL;

-- Cart_Order with missing Order
SELECT 'Orphan Cart_Order (Order)', COUNT(*)
FROM cart.Cart_Order co
LEFT JOIN orders.Orders o ON co.Order_ID =
o.Order_ID
WHERE o.Order_ID IS NULL;

-- ===== Enum Validations =====
SELECT 'Invalid Order Status', COUNT(*)
FROM orders.Orders
WHERE Status NOT IN ('Pending', 'Shipped',
'Delivered', 'Cancelled', 'Returned');

SELECT 'Invalid Returned Status', COUNT(*)
FROM orders.Returned
WHERE Status NOT IN ('Requested', 'Approved',
'Rejected', 'Processed');

SELECT 'Invalid Payment Method', COUNT(*)
FROM orders.Payment
WHERE Payment_Method NOT IN ('Credit Card',
'Debit Card', 'PayPal', 'Cash on Delivery', 'Bank
Transfer');

SELECT 'Invalid Payment Status', COUNT(*)
FROM orders.Payment
WHERE Payment_Status NOT IN ('Pending',
'Completed', 'Failed', 'Refunded');

-- ===== Value Validations =====
SELECT 'Negative Price or Stock in Product',
COUNT(*)

```



```

FROM products.Product
WHERE Price < 0 OR Stock_Quantity < 0;

SELECT 'Invalid Quantity/Price in OrderDetails',
COUNT(*)
FROM orders.OrderDetails
WHERE Quantity <= 0 OR UnitPrice < 0;

SELECT 'Invalid Estimated Days/Cost in
ShipMethod', COUNT(*)
FROM orders.ShipMethod
WHERE Estimated_Days <= 0 OR Cost < 0;

SELECT 'Negative Amount in Payment', COUNT(*)
FROM orders.Payment
WHERE Amount < 0;

SELECT 'Invalid Quantity in Cart_Items', COUNT(*)
FROM cart.Cart_Items
WHERE Quantity <= 0;

-- ===== Uniqueness Violations =====
SELECT 'Duplicate Emails', COUNT(*)
FROM (
    SELECT Email FROM users.Customer
    WHERE Email IS NOT NULL
    GROUP BY Email HAVING COUNT(*) > 1
) dup;

SELECT 'Duplicate Phones', COUNT(*)
FROM (
    SELECT Phone FROM users.Customer
    WHERE Phone IS NOT NULL
    GROUP BY Phone HAVING COUNT(*) > 1

```

```

) dup;

-- ===== Date Validations =====
SELECT 'Invalid Dates (Ship/Due before Order)',
COUNT(*)
FROM orders.Orders
WHERE
    (Ship_Date IS NOT NULL AND Ship_Date <
Order_Date)
    OR (Due_Date IS NOT NULL AND Due_Date <
ISNULL(Ship_Date, Order_Date));

-- NULL Checks
-- =====

-- Products without names
SELECT 'Products with NULL Names', COUNT(*)
FROM products.Product
WHERE Product_Name IS NULL;

-- Customers without email addresses (if
required)
SELECT 'Customers with NULL Emails', COUNT(*)
FROM users.Customer
WHERE Email IS NULL;

-- Shipping methods without cost
SELECT 'Ship Methods with NULL Cost', COUNT(*)
FROM orders.ShipMethod
WHERE Cost IS NULL;

-- OrderDetails with missing Quantity or
UnitPrice

```

```

SELECT 'OrderDetails with NULL Quantity or
UnitPrice', COUNT(*)
FROM orders.OrderDetails
WHERE Quantity IS NULL OR UnitPrice IS NULL;

-- Payments with missing Payment_Status
SELECT 'Payments with NULL Payment_Status',
COUNT(*)
FROM orders.Payment
WHERE Payment_Status IS NULL;

-- Cross-Field Validations
-- =====

-- Validate Total = Quantity * UnitPrice (with
0.01 margin)
SELECT 'Mismatched Payment Amounts', COUNT(*)
FROM orders.Payment p
JOIN (
    SELECT
        o.Order_ID,
        SUM(od.Quantity * od.UnitPrice) AS
CalculatedTotal
    FROM orders.Orders o
    JOIN orders.OrderDetails od ON o.Order_ID =
od.Order_ID
    GROUP BY o.Order_ID
) expected ON p.Order_ID = expected.Order_ID
WHERE ABS(p.Amount - expected.CalculatedTotal) >
0.01;

```

```

-- Reverse Referential Integrity Checks
-- =====

-- Customers who have never placed an order
SELECT 'Customers without Orders', COUNT(*)
FROM users.Customer c
LEFT JOIN orders.Orders o ON c.Cust_ID =
o.Cust_ID
WHERE o.Order_ID IS NULL;

-- Products never purchased
SELECT 'Products without Sales', COUNT(*)
FROM products.Product p
LEFT JOIN orders.OrderDetails od ON p.Product_ID
= od.Product_ID
WHERE od.Order_ID IS NULL;

-- Business Rule Violations
-- =====

-- Delivered orders with no associated payment
SELECT 'Delivered Orders without Payments',
COUNT(*)
FROM orders.Orders o
LEFT JOIN orders.Payment p ON o.Order_ID =
p.Order_ID
WHERE o.Status = 'Delivered' AND p.Payment_ID IS
NULL;

-- Reviews with ratings outside valid range
(e.g., 1 to 5)

```

```

SELECT 'Reviews with Invalid Ratings', COUNT(*)
FROM users.Review
WHERE Rating < 1 OR Rating > 5;

-- Products with NULL or zero price
SELECT 'Products with NULL or 0 Price', COUNT(*)
FROM products.Product
WHERE Price IS NULL OR Price = 0;

--Basic Statistical Anomaly Checks
-- =====

-- Products with extremely high prices (above 3x
average)
SELECT 'Price Outliers (Very Expensive
Products)', COUNT(*)
FROM products.Product
WHERE Price > (
    SELECT AVG(Price) * 3 FROM products.Product
);

-- Customers with unusually high order count
(>100 orders)
SELECT 'High Volume Customers (>100 Orders)',
COUNT(*)
FROM (
    SELECT Cust_ID, COUNT(*) AS OrderCount
    FROM orders.Orders
    GROUP BY Cust_ID
    HAVING COUNT(*) > 100
) HighVolume;

SELECT
CASE

```

```

        WHEN EXISTS (
            SELECT 1
            FROM users.Customer c
            LEFT JOIN orders.Orders o ON
c.Cust_ID = o.Cust_ID
            WHERE o.Order_ID IS NULL
        )
        THEN ' No - Some customers have no
orders '
        ELSE 'Yes - All customers have at least
one order '
        END AS Result;

SELECT COUNT(*) FROM users.Customer;
SELECT COUNT(DISTINCT Cust_ID) FROM
orders.Orders;

SELECT
    (SELECT SUM(Total_Amount) FROM orders.Orders)
AS Total_Order_Amount,
    (SELECT SUM(Amount) FROM orders.Payment) AS
Total_Payment_Amount,
    (SELECT SUM(UnitPrice * Quantity) FROM
orders.OrderDetails) AS
Calculated_OrderDetails_Total;

SELECT 'Delivered Orders without Payments',
COUNT(DISTINCT o.Order_ID)
FROM orders.Orders o
LEFT JOIN orders.Payment p ON o.Order_ID =
p.Order_ID
WHERE o.Status = 'Delivered' AND p.Payment_ID IS
NULL;

```

```
SELECT COUNT(*) AS TotalOrders FROM
orders.Orders;
```

```
SELECT COUNT(*) AS DeliveredOrders
FROM orders.Orders
WHERE Status = 'Delivered';
```

```
SELECT COUNT(DISTINCT o.Order_ID) AS
DeliveredWithoutPayment
FROM orders.Orders o
LEFT JOIN orders.Payment p ON o.Order_ID =
p.Order_ID
WHERE o.Status = 'Delivered' AND p.Payment_ID IS
NULL;
```

```
SELECT Status, COUNT(*) AS Order_Count
FROM orders.Orders
WHERE Status IN ('Cancelled',
'Pending', 'Delivered')
GROUP BY Status;
```

Chapter03.

SSIS Implementation:

3.1 Overview of SSIS :

SQL Server Integration Services (SSIS) is a platform for building high-performance data integration and workflow solutions.

It is a component of Microsoft SQL Server and is used to perform **Extract, Transform, and Load (ETL)** operations.

SSIS enables developers and data engineers to design, deploy, and manage data workflows between different sources and destinations.

Key Features

- **Data Extraction:** Connects to various sources such as SQL Server, Oracle, Excel, flat files, and cloud-based data.
- **Data Transformation:** Cleanses, formats, and transforms raw data to meet business requirements using built-in and custom transformations.
- **Data Loading:** Efficiently loads processed data into target systems such as data warehouses, databases, or flat files.
- **Workflow Automation:** Automates complex tasks like file transfers, database updates, and sending emails.
- **Error Handling & Logging:** Provides robust error tracking, logging, and debugging tools.

Common Use Cases

- Migrating data between systems
- Building and maintaining a **Data Warehouse**
- Automating repetitive data processing tasks
- Integrating data from multiple heterogeneous sources

- Performing incremental data loads and updates

In summary, SSIS is a powerful and scalable ETL tool that plays a key role in modern data integration and business intelligence solutions.

3.2 SSIS Architecture:

The SSIS architecture is built on a set of components that work together to perform data integration and workflow operations. Its main components include:

1. **SSIS Service** – Manages the execution and storage of SSIS packages.
2. **SSIS Designer** – A graphical interface within SQL Server Data Tools (SSDT) used to create and configure packages.
3. **SSIS Packages** – The core executable units in SSIS, containing control flow, data flow, event handlers, and parameters.
4. **Control Flow** – Defines the workflow of tasks (e.g., execute SQL, send email, file system operations).
5. **Data Flow** – Manages the extraction, transformation, and loading of data between sources and destinations.
6. **Connection Managers** – Define connections to source and destination systems (databases, files, web services).
7. **Event Handlers** – Respond to events during package execution (e.g., on error, on completion).
8. **SSIS Runtime Engine** – Executes control flow and coordinates the running of tasks and containers.
9. **Data Flow Pipeline Engine** – Handles the movement and transformation of data between components in the data flow.

This architecture allows SSIS to provide a scalable, high-performance platform for data migration, transformation, and workflow automation.

3.3 Data Flow Design:

The **Data Flow** in SSIS defines how data is extracted from sources, transformed, and loaded into destinations.

It is designed and configured using the **Data Flow Task**, which contains multiple components working together to process data efficiently.

Key Components of Data Flow

1. Data Sources

- Origin points where SSIS retrieves data.
- Examples: SQL Server, Oracle, Excel, Flat Files, APIs, Azure Blob Storage.

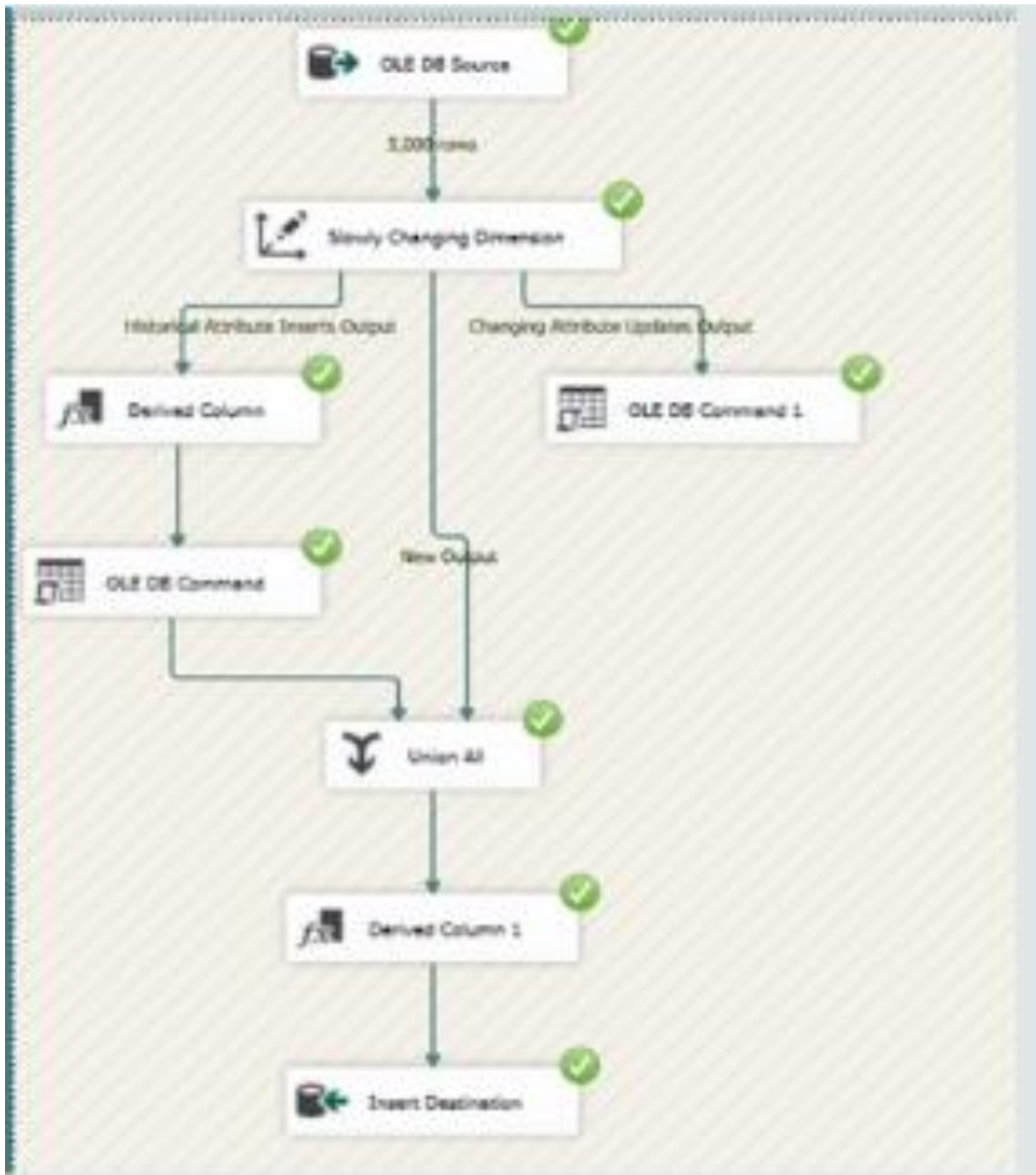
2. Transformations

- Operations applied to data to clean, standardize, aggregate, or manipulate it.
- Examples:
 - *Derived Column* – Create new calculated fields.
 - *Data Conversion* – Change data types.
 - *Sort* – Arrange data in a specific order.
 - *Lookup* – Match and enrich data from other datasets.
 - *Aggregate* – Summarize data (e.g., totals, averages).

3. Data Destinations

- Final storage locations for processed data.
- Examples: SQL Server, Data Warehouse, CSV files, Cloud storage.

3.3.1 Customer Dimension:



Figure(4)

3.3.2 Product Dimension:



Figure(5)

3.3.3 ShipMethod Dimension:



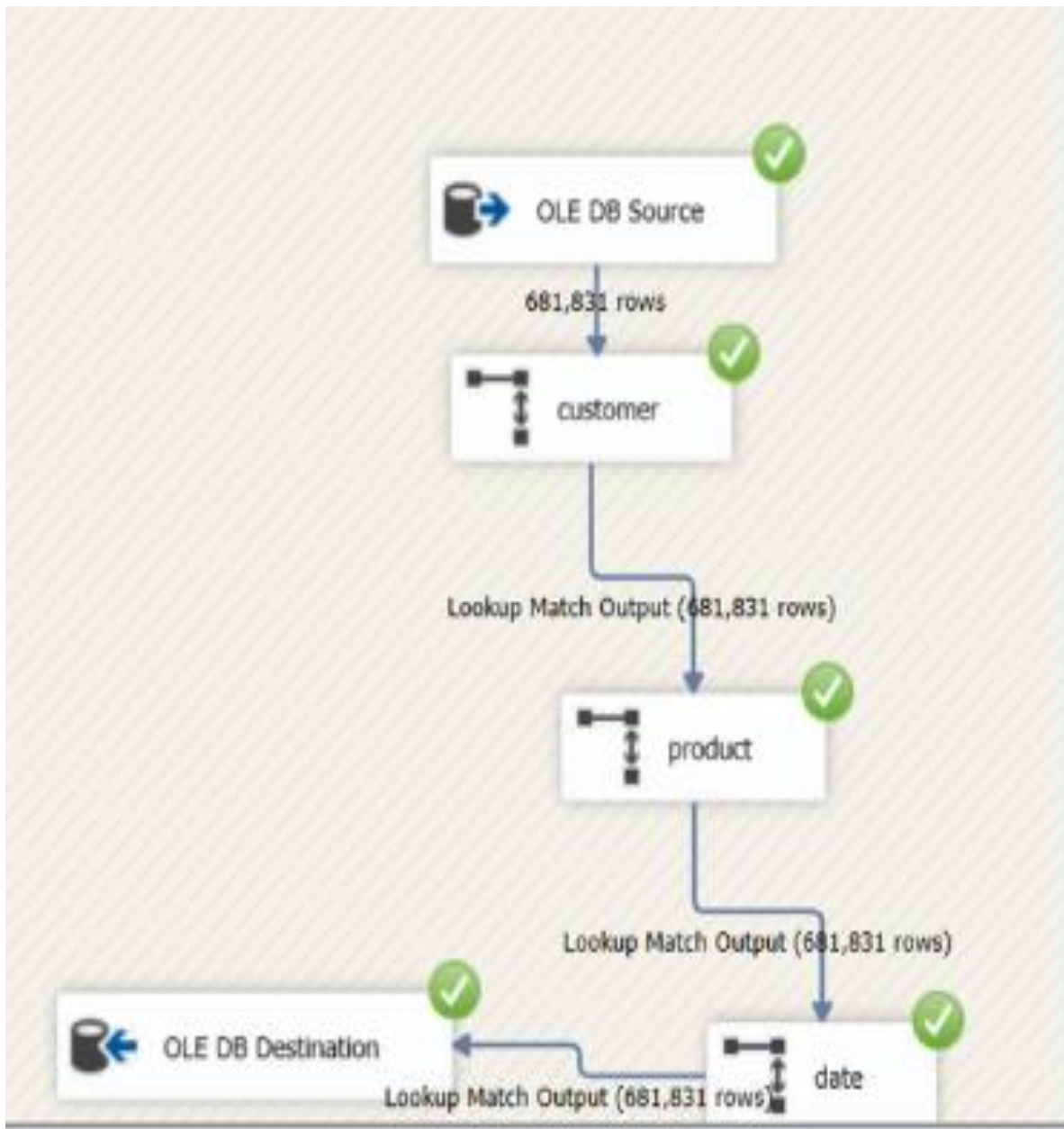
Figure(6)

3.3.4 Fact Sales:



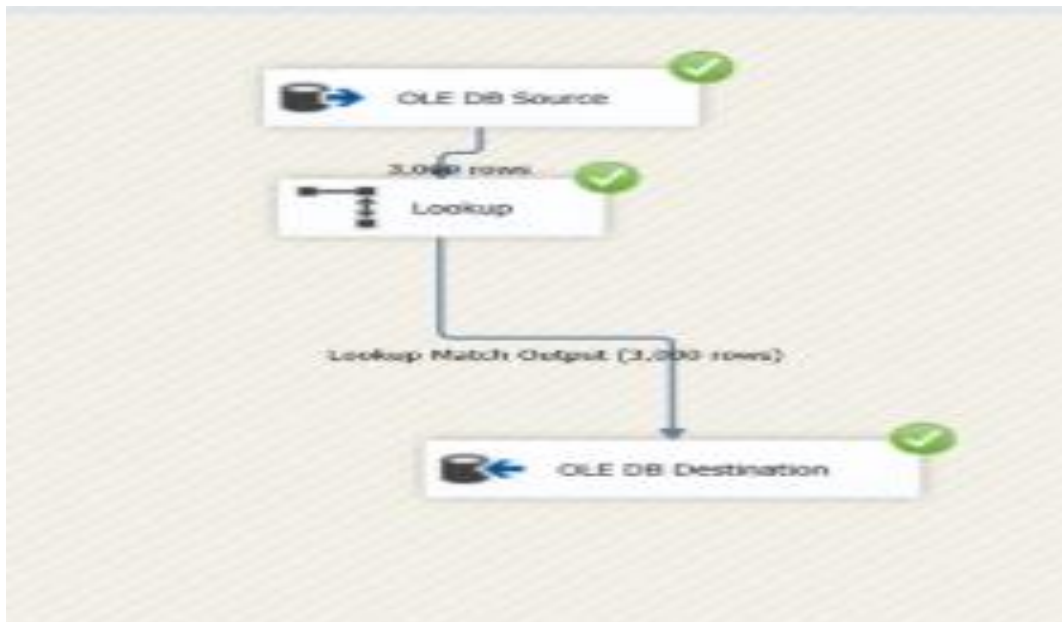
Figure(7)

3.3.5 Fact CartActivity:



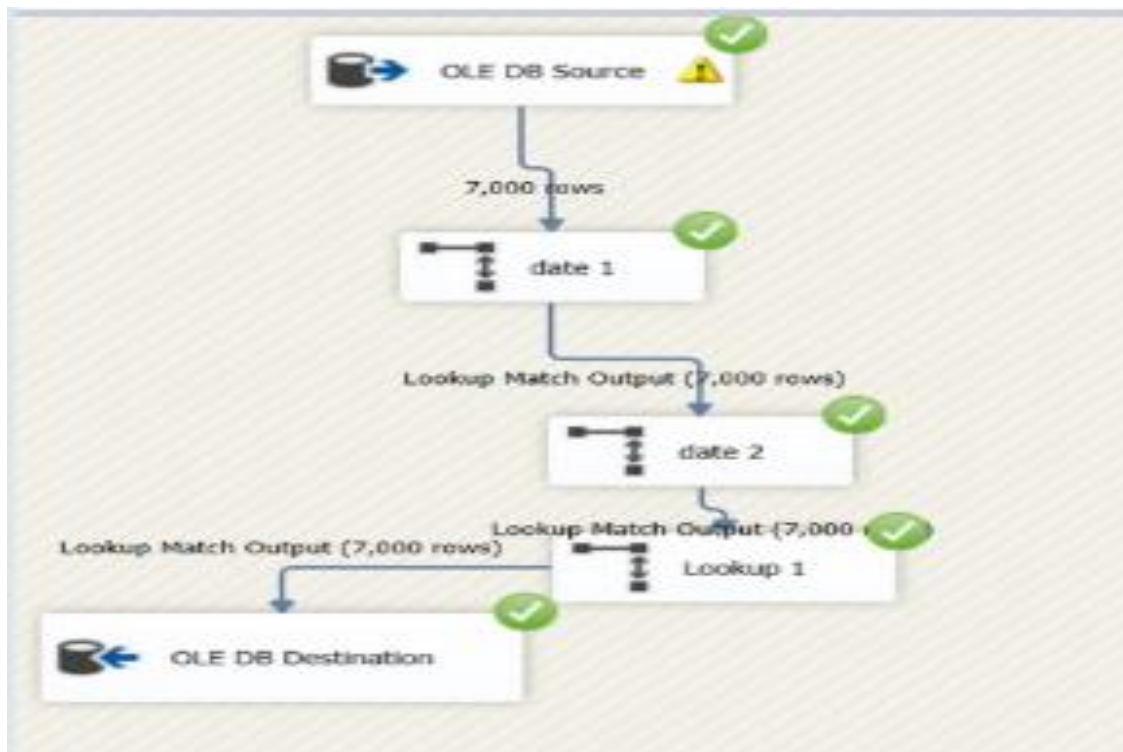
Figure(8)

3.3.6 Fact Payment:



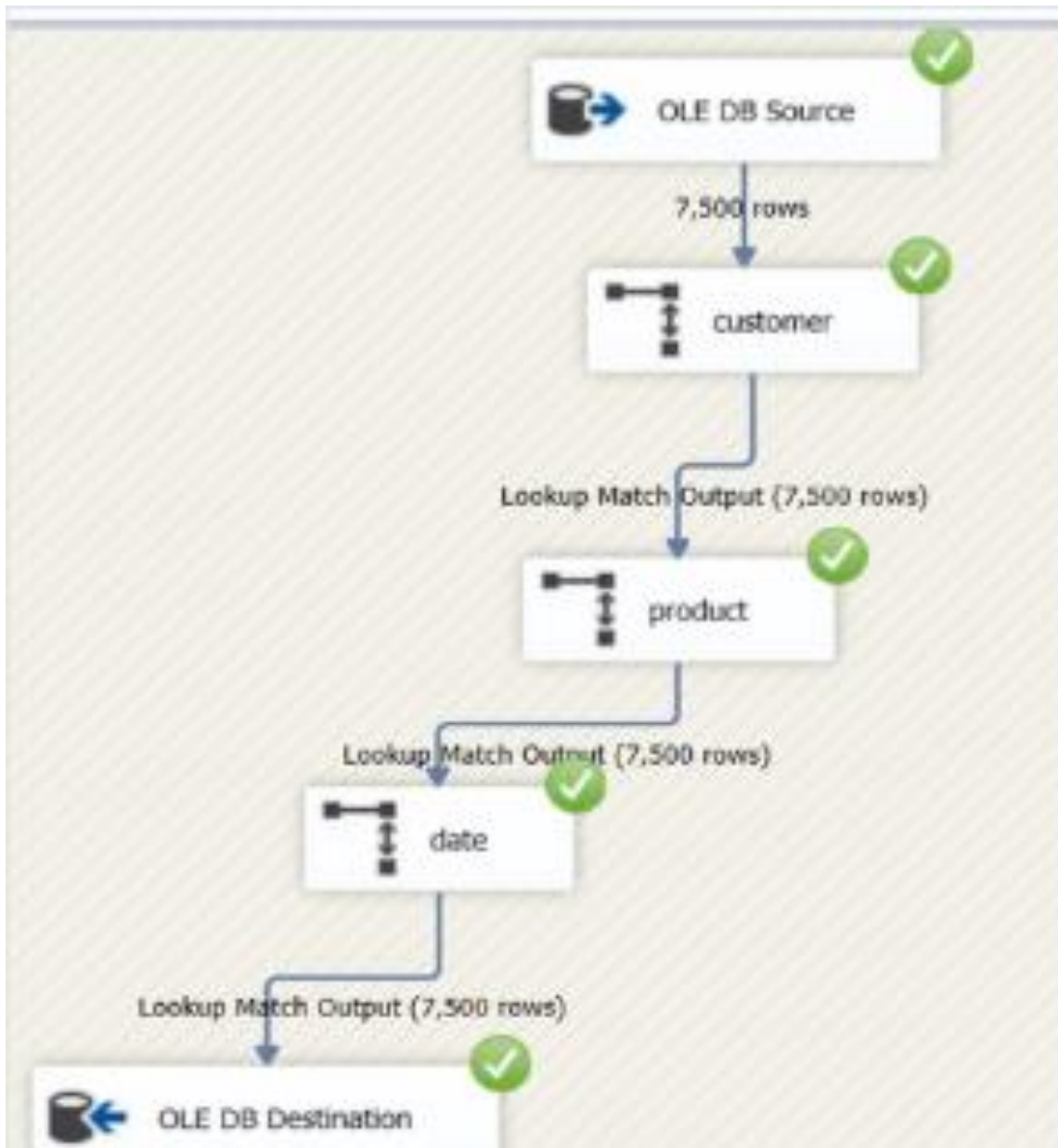
Figure(9)

3.3.7 Fact Return:



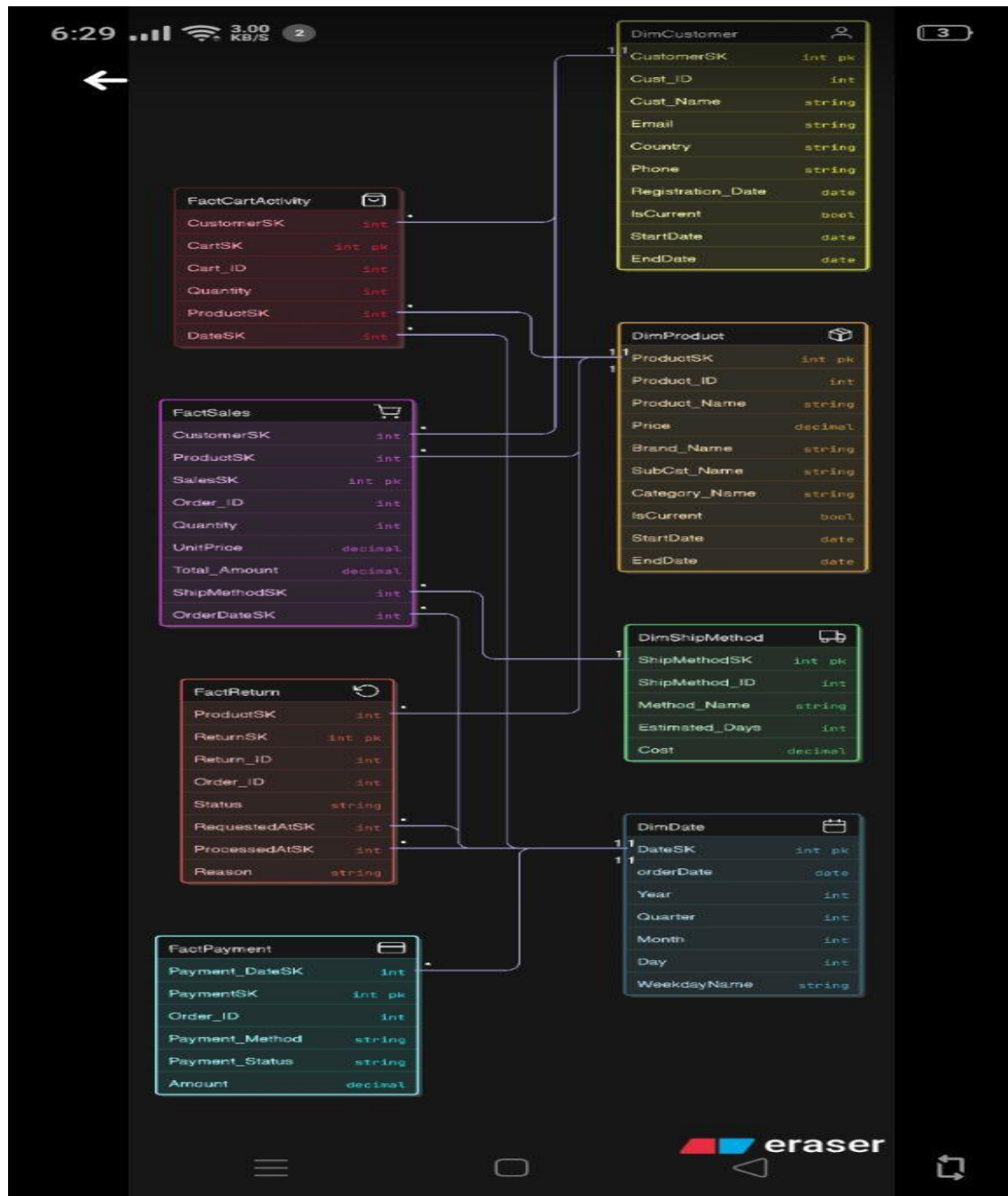
Figure(10)

3.3.8 Fact Review:



Figure(11)

Dimension Table(DWH)



Figure(12)

Chapter04.

Data Modeling(SSAS)

4.1 Overview of SSAS:

SQL Server Analysis Services (SSAS) is an online analytical processing (OLAP) and data mining tool from Microsoft, used for analyzing and visualizing large volumes of data.

It enables the creation of **multidimensional** and **tabular** data models that can be queried using **MDX** (Multidimensional Expressions) or **DAX** (Data Analysis Expressions).

SSAS is designed to support **business intelligence (BI)** solutions by providing a fast, interactive way to explore and analyze data from various sources.

Key Features

- **Data Modeling** – Build OLAP cubes or tabular models for structured analysis.
- **High Performance** – Pre-calculates and aggregates data to deliver fast query responses.
- **Integration with BI Tools** – Works seamlessly with Power BI, Excel, and other visualization tools.
- **Security** – Role-based access control at the model, dimension, and cell level.
- **Data Mining** – Offers algorithms to identify patterns, trends, and predictions.

Common Use Cases

- Creating centralized data models for reporting and analysis.
- Building complex calculations and KPIs for business dashboards.
- Analyzing large datasets for trends, patterns, and anomalies.
- Providing self-service analytics capabilities to business users.

In summary, SSAS transforms raw data into well-structured analytical models, empowering organizations to make data-driven decisions quickly and effectively.

4.2 SSAS Architecture:

The architecture of **SQL Server Analysis Services (SSAS)** is designed to support fast, scalable, and secure analytical processing. It consists of several core components that work together to manage, store, and process multidimensional and tabular data models.

Main Components

1. Data Sources

- External systems from which SSAS retrieves data.
- Examples: SQL Server, Oracle, Excel files, cloud databases.

2. Data Source View (DSV)

- A logical layer that defines the schema used in SSAS without altering the actual source data.
- Provides a unified view of multiple data sources.

3. Dimensions

- Structures that categorize and describe data (e.g., Time, Geography, Product).
- Provide context for measures in analysis.

4. Measures and Measure Groups

- **Measures:** Numeric data points used for analysis (e.g., Sales Amount, Quantity).
- **Measure Groups:** Collections of related measures.

5. Cubes (for Multidimensional Models)

- Multidimensional structures containing measures and dimensions.
- Pre-aggregated for high performance in OLAP queries.

6. Tabular Models

- In-memory models that use columnar storage and the xVelocity (VertiPaq) engine for fast calculations.

7. Storage Modes

- **MOLAP** (Multidimensional OLAP) – Data is pre-aggregated and stored in SSAS.
- **ROLAP** (Relational OLAP) – Queries source data directly.
- **HOLAP** (Hybrid OLAP) – Combination of MOLAP and ROLAP.

8. Processing Engine

- Loads and processes data from the source into the SSAS model.

9. Query Engine

- Handles requests from client tools, executing MDX or DAX queries.

10. Security Layer

- Implements role-based security to control access at the model, dimension, or cell level.

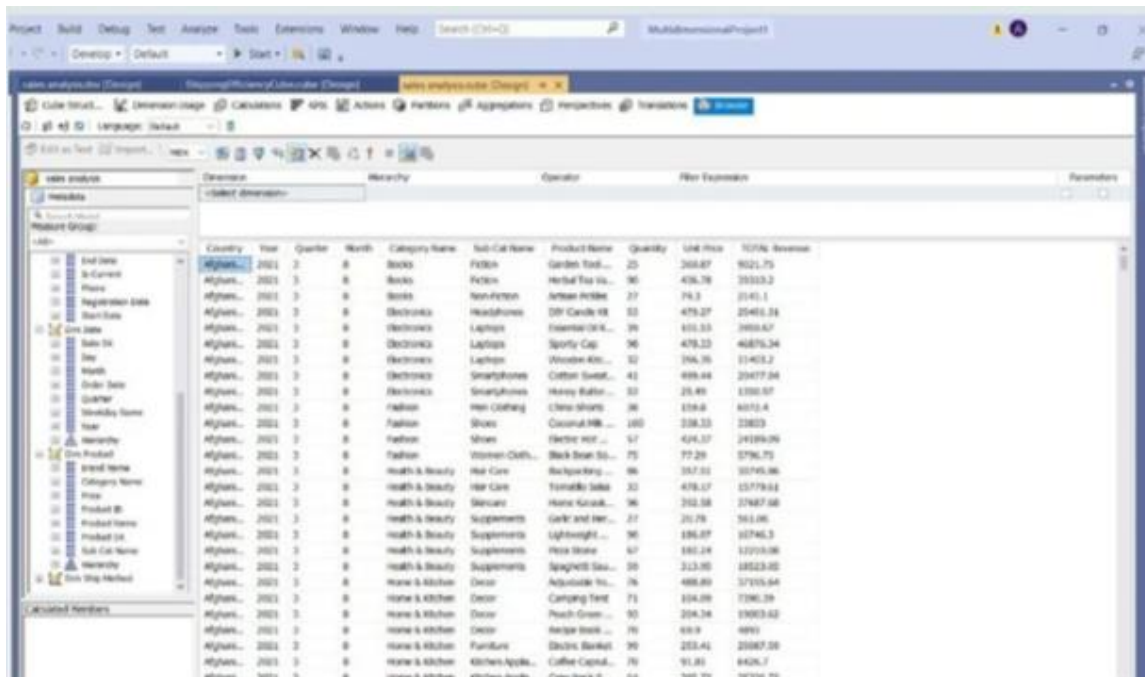
4.3 Data Model Design:

The Data Model Design process in SSAS focuses on structuring data in a way that enables fast, flexible, and insightful analysis. A well-designed model ensures that business users can easily explore data, create reports, and gain actionable insights.

The SSAS Cube is a core component in Microsoft SQL Server Analysis Services used to organize and analyze multi-dimensional data. It combines dimensions such as Country, Year, and Product with measures like Quantity and Revenue, enabling fast and interactive analytics.

Through the cube, users can browse data, perform aggregations, and use analysis features such as Drill Down and Drill Up to navigate between detailed and summarized views effortlessly.

4.3.1 Sales Cube:



Country	Year	Quarter	Month	Category Name	Sub-Category Name	Product Name	Quantity	Unit Price	Total Revenue
Afghanistan	2012	1	8	Books	Fiction	Garden Tool...	25	200.87	5021.75
Afghanistan	2012	1	8	Books	Fiction	Herbal Tea...	90	436.78	39310.2
Afghanistan	2012	1	8	Books	Non-Fiction	Jetson Hdd...	27	78.3	2115.1
Afghanistan	2012	1	8	Electronics	Headphones	DP Candle H...	53	479.27	25451.38
Afghanistan	2012	1	8	Electronics	Laptops	Insental DCL...	39	151.33	5901.67
Afghanistan	2012	1	8	Electronics	Laptops	Sporty Cap...	96	479.22	46076.34
Afghanistan	2012	1	8	Electronics	Laptops	Wooden Kic...	32	356.36	11403.2
Afghanistan	2012	1	8	Electronics	Smartphones	Cotton Swea...	41	489.44	20177.84
Afghanistan	2012	1	8	Electronics	Smartphones	Heavy Balle...	53	25.89	1380.57
Afghanistan	2012	1	8	Fashion	Men Clothing	Chino Shor...	98	159.8	15651.4
Afghanistan	2012	1	8	Fashion	Shoes	Coconut M...	180	218.33	39300
Afghanistan	2012	1	8	Fashion	Shoes	Electric H...	57	424.37	24199.09
Afghanistan	2012	1	8	Fashion	Women Cloth...	Black Bean S...	75	77.28	5796.75
Afghanistan	2012	1	8	Health & Beauty	Hair Care	Backpackin...	86	157.51	13545.86
Afghanistan	2012	1	8	Health & Beauty	Hair Care	Yarnable Sal...	33	479.17	15779.61
Afghanistan	2012	1	8	Health & Beauty	Skin Care	Home Kiosk...	96	252.58	24257.68
Afghanistan	2012	1	8	Health & Beauty	Supplements	Gelatin and...	27	21.78	588.06
Afghanistan	2012	1	8	Health & Beauty	Supplements	Lightweight...	90	186.87	16818.3
Afghanistan	2012	1	8	Health & Beauty	Supplements	Hot Stone...	67	180.24	12075.08
Afghanistan	2012	1	8	Health & Beauty	Supplements	SpagHetti S...	39	213.95	8343.05
Afghanistan	2012	1	8	Home & Kitchen	Decor	Adjustable...	76	488.80	37108.80
Afghanistan	2012	1	8	Home & Kitchen	Decor	Camping Tent	71	104.88	7346.68
Afghanistan	2012	1	8	Home & Kitchen	Decor	Pouch Green...	90	224.34	20190.60
Afghanistan	2012	1	8	Home & Kitchen	Decor	Neige Brak...	70	69.9	4893
Afghanistan	2012	1	8	Home & Kitchen	Furniture	Electric Bunk...	90	233.41	20987.50
Afghanistan	2012	1	8	Home & Kitchen	Kitchen Appla...	Coffee Capsu...	70	91.81	6426.7
Afghanistan	2012	1	8	Home & Kitchen	Kitchen Appla...	Coke Back S...	68	385.75	26231.00

Figure(13)

4.3.2 Payment Cube:

Payment Method	Payment Status	Amount	Year	Month	Day	Quarter	Number of Orders
Bank Transfer	Completed	17313...	2023	4	12	2	1
Bank Transfer	Completed	21043...	2024	2	26	1	1
Bank Transfer	Completed	26315...	2023	11	10	4	1
Bank Transfer	Completed	27114...	2024	9	14	3	1
Bank Transfer	Completed	28533...	2024	4	19	2	1
Bank Transfer	Completed	29358...	2023	2	11	1	1
Bank Transfer	Completed	29746...	2023	3	6	1	1
Bank Transfer	Completed	30323...	2024	1	20	1	1
Bank Transfer	Completed	31249...	2024	11	4	4	1
Bank Transfer	Completed	31259...	2023	1	13	1	1
Bank Transfer	Completed	31933...	2023	2	2	1	1
Bank Transfer	Completed	31984...	2024	2	23	1	1
Bank Transfer	Completed	33114...	2023	1	8	1	1
Bank Transfer	Completed	33636...	2023	9	12	3	1
Bank Transfer	Completed	34209...	2024	5	6	2	1
Bank Transfer	Completed	34491...	2023	2	28	1	1
Bank Transfer	Completed	35253...	2024	3	3	1	1
Bank Transfer	Completed	35311...	2024	10	4	4	1
Bank Transfer	Completed	35389...	2024	2	1	1	1
Bank Transfer	Completed	36719...	2024	10	18	4	1
Bank Transfer	Completed	36905...	2024	1	30	1	1
Bank Transfer	Completed	37404...	2024	7	15	3	1
Bank Transfer	Completed	37449...	2023	6	7	2	1
Bank Transfer	Completed	37525...	2024	10	5	4	1

Figure(14)

4.3.3 ShipMethod Cube:

Method Name	Country	Fact Sales Count
Economy Shipping	Algeria	45
Economy Shipping	Albania	129
Economy Shipping	Angola	30
Economy Shipping	Anguilla and Barbuda	30
Economy Shipping	Argentina	288
Economy Shipping	Australia	39
Economy Shipping	Azerbaijan	88
Economy Shipping	Bangladesh	62
Economy Shipping	Belize	32
Economy Shipping	Bolivia	61
Economy Shipping	Bosnia and Herzegovina	40
Economy Shipping	Bolivia	43
Economy Shipping	Brazil	551
Economy Shipping	Bulgaria	129
Economy Shipping	Cameroon	34
Economy Shipping	Canada	328
Economy Shipping	Chad	30
Economy Shipping	China	3917
Economy Shipping	Colombia	185
Economy Shipping	Costa Rica	40
Economy Shipping	Croatia	118
Economy Shipping	Cuba	108
Economy Shipping	Czech Republic	467
Economy Shipping	Democratic Republic of the Congo	37
Economy Shipping	Denmark	40
Economy Shipping	Dominican Republic	34

Figure(15)

4.3.4 Return Cube:

Product Name	Price	Brand Name	Category Name	Year	Month	Day	Number of Returns	Distinct Returned Products
Adjustable Bar Stools	149.99	Nike	Fashion	2022	11	9	1	1
Adjustable Bar Stools	149.99	Nike	Fashion	2022	7	2	1	1
Adjustable Bar Stools	149.99	Nike	Fashion	2022	9	15	1	1
Adjustable Bar Stools	149.99	Nike	Fashion	2023	3	3	1	1
Adjustable Bar Stools	149.99	Nike	Fashion	2023	3	9	1	1
Adjustable Bar Stools	149.99	Nike	Fashion	2023	5	18	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2022	2	2	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2022	4	29	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2022	6	5	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2022	9	26	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2023	7	5	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2023	8	16	1	1
Adjustable Dog Harness	24.99	Nike	Fashion	2023	9	4	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2022	6	17	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2022	7	18	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2022	7	5	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2022	9	24	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2023	1	26	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2023	5	16	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2023	8	11	1	1
Adjustable Dumbbells	249.99	Asus	Fashion	2023	8	26	1	1
Adjustable Dumbbells	249.99	Microsoft	Fashion	2022	5	28	1	1
Adjustable Dumbbells	249.99	Microsoft	Fashion	2023	3	4	1	1
Adjustable Garden Rake	22.99	Apple	Home & Kitchen	2022	12	4	1	1
Adjustable Garden Rake	22.99	Apple	Home & Kitchen	2022	2	2	1	1
Adjustable Garden Rake	22.99	Apple	Home & Kitchen	2023	8	16	1	1

Figure(16)

4.3.5 CartActivity Cube:

Cust Name	Year	Month	Day	Country	Category Name	Cart ID	Distinct Count	Product SK	Distinct Count	Quantity
Abby Crowl	2025	5	15	Greece	Books	1	9			443
Abby Crowl	2025	5	15	Greece	Electronics	1	13			396
Abby Crowl	2025	5	15	Greece	Fashion	1	12			824
Abby Crowl	2025	5	15	Greece	Health & Beauty	1	18			873
Abby Crowl	2025	5	15	Greece	Home & Kitchen	1	13			854
Abby Crowl	2025	5	15	Greece	Sports & Outd...	1	3			96
Abby Lorme	2025	4	8	Canada	Books	1	4			258
Abby Lorme	2025	4	8	Canada	Electronics	1	11			496
Abby Lorme	2025	4	8	Canada	Fashion	1	3			268
Abby Lorme	2025	4	8	Canada	Health & Beauty	1	4			279
Abby Lorme	2025	4	8	Canada	Home & Kitchen	1	4			223
Abby Lorme	2025	4	8	Canada	Sports & Outd...	1	4			179
Abigail Ky...	2025	3	20	Indonesia	Books	1	10			302
Abigail Ky...	2025	3	20	Indonesia	Electronics	1	9			536
Abigail Ky...	2025	3	20	Indonesia	Fashion	1	13			599
Abigail Ky...	2025	3	20	Indonesia	Health & Beauty	1	9			371
Abigail Ky...	2025	3	20	Indonesia	Home & Kitchen	1	14			753
Abigail Ky...	2025	3	20	Indonesia	Sports & Outd...	1	4			142
Abigail Doo...	2025	2	13	Nepal	Books	1	10			573
Abigail Doo...	2025	2	13	Nepal	Electronics	1	5			329
Abigail Doo...	2025	2	13	Nepal	Fashion	1	6			344
Abigail Doo...	2025	2	13	Nepal	Health & Beauty	1	2			171

Figure(17)

Chapter05.

Reporting With(SSRS)

5.1 Overview of SSRS:

SQL Server Reporting Services (SSRS) is a comprehensive, server-based reporting platform developed by Microsoft. It enables the creation, management, and distribution of a wide variety of interactive and paginated reports. SSRS provides a complete suite of tools for designing visually rich reports, scheduling automated delivery, and integrating reporting capabilities into applications. It supports multiple data sources, allowing the generation of charts, tables, KPIs, and dashboards. Reports can be delivered in various formats such as PDF, Excel, and Word, and accessed via a web portal, email subscriptions, or embedded directly into custom solutions. This makes SSRS a powerful and flexible component within the Microsoft Business Intelligence stack.

5.2 Developed Reports:

The developed reports in SSRS are designed to provide comprehensive insights into business data by transforming raw information into meaningful visualizations. These reports may include interactive dashboards, detailed tabular reports, and visual charts tailored to meet specific business requirements. Each report is connected to its underlying data source, ensuring that information is accurate and up to date. Features such as parameters, drill-down functionality, and conditional formatting are implemented to enhance user interaction and usability. The reports are also optimized for export in multiple formats, enabling stakeholders to access and share insights efficiently.

5.2.1 Selling Product Report:

Selling Product in Countries per Category					
Health & Beauty					
Country	Sub-Cell Name	Product Name	Avg Price	Total Quantity	Total Revenue
France			880 \$	4780	4,201,600 \$
			880.00 \$	51180	4,504,800 \$
			1,380 \$	44645	61,601,800 \$
			880.00 \$	40000	3,504,000 \$
			880.00 \$	46404	4,083,168 \$
Germany	Essence	Whisper Security	1,400.00 \$	84100	1,177,400 \$
	Essence	Whisper Security	1,400.00 \$	84100	1,177,400 \$
	Essence	Whisper Security	1,400.00 \$	84100	1,177,400 \$
	Essence	Whisper Security	1,400.00 \$	84100	1,177,400 \$
	Essence	Whisper Security	1,400.00 \$	84100	1,177,400 \$
Indonesia			880 \$	1880	1,654,400 \$
			880.00 \$	21000	1,848,000 \$
			1,380 \$	13485	18,609,000 \$
			880.00 \$	34170	3,006,720 \$
			880.00 \$	46404	4,083,168 \$
Japan			880 \$	4780	4,201,600 \$
			880.00 \$	51180	4,504,800 \$
			1,380 \$	44645	61,601,800 \$
			880.00 \$	40000	3,504,000 \$
			880.00 \$	46404	4,083,168 \$

Figure(18)

Selling Product in Countries per Category					
Home & Kitchen					
Country	Sub-Cell Name	Product Name	Avg Price	Total Quantity	Total Revenue
China			880 \$	42844	3,769,776 \$
			180.00 \$	50000	9,000,000 \$
	Furniture	Digital Window	190.00 \$	87700	1,667,300 \$
	Decor	Old Learning	120.00 \$	81004	9,720,480 \$
	Decor	Old Learning	120.00 \$	81004	9,720,480 \$
Indonesia			880 \$	1880	1,654,400 \$
			180.00 \$	51004	9,180,720 \$
			1,380 \$	44645	61,601,800 \$
			880.00 \$	40000	3,504,000 \$
			880.00 \$	46404	4,083,168 \$


Figure(19)

5.2.2 Cart Report:

Cart Activity Overview

Figure(20)

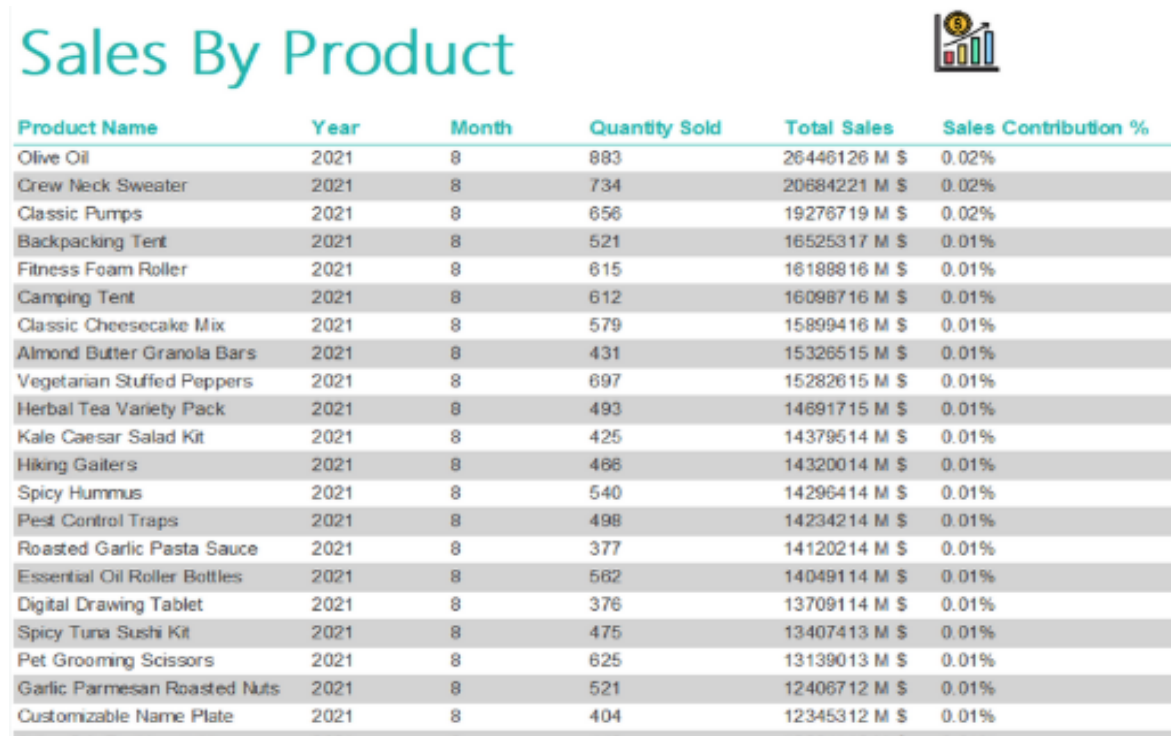
5.2.3 Customer Report



Country	Total Customers	÷ Total Sales	÷ Top Customer
China	418	236112015.16	Bayard Loosley
Indonesia	271	148839685.70	Mil Cowdry
Philippines	126	71380805.17	Nikki Capelow
Russia	117	66916162.91	Tiler Torricina
Brazil	81	48842685.47	Marice Willmott
Poland	75	43938365.34	Julio Benediktsson
France	69	39088665.14	Max MacCall
Portugal	66	34700560.33	Arkuene Daintrey
Sweden	60	31853957.52	Inesita Lempel
United States	42	25480074.12	Janetta Zanioletti
Peru	43	23823501.13	Shurwood Sofe
Czech Republic	33	21884733.70	Jephthah Greder
Canada	41	21722028.80	Philbert Shimmin
Ukraine	37	21108570.50	Miner Grishunin
Thailand	37	20418800.68	Elcia Ovell
Japan	36	19378309.23	Danyotte Spata
Greece	34	19224862.32	Clemmy Ulyatt
Argentina	31	17421448.33	Gabby Tredger
Mexico	28	15058760.32	Bekki Polondino
Colombia	25	13431435.21	Ingiebert Berger
Vietnam	22	13323904.28	Melonie Dibley
Nigeria	25	10884803.48	Kennett O'Carroll

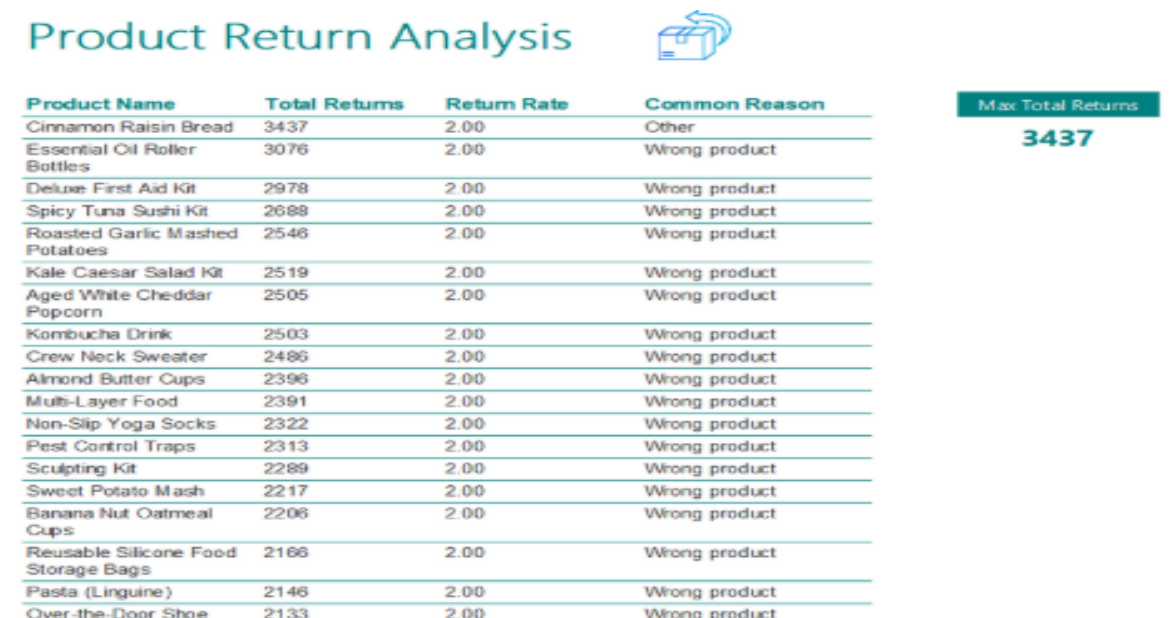
Figure(21)

5.2.4 Product Report



Figure(22)

5.2.5 Return Report



Figure(23)

Chapter 06.

Visualization

6.1 Power BI:

Power BI is a business analytics service by Microsoft that enables users to connect to various data sources, transform raw data, and create interactive reports and dashboards. It provides powerful data visualization capabilities, allowing organizations to gain actionable insights and make data-driven decisions. Power BI supports real-time data monitoring, integrates seamlessly with other Microsoft tools, and offers cloud-based as well as on-premises deployment options, making it a flexible solution for diverse business intelligence needs.



Figure(24)



Figure(25)



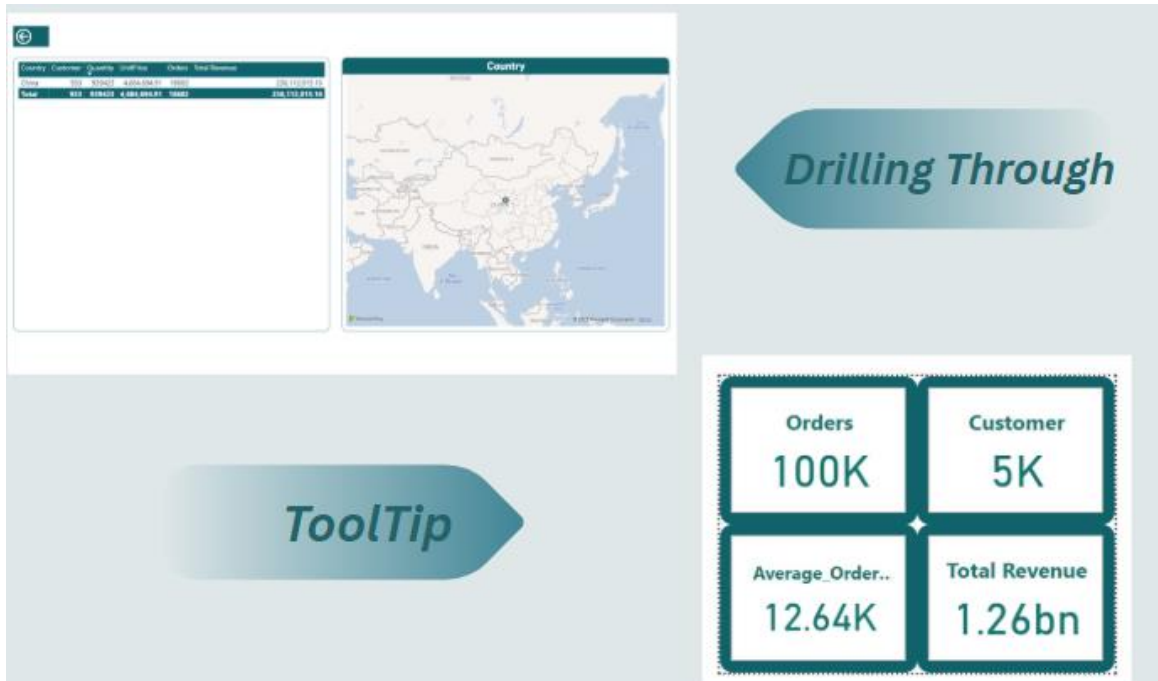
Figure(26)



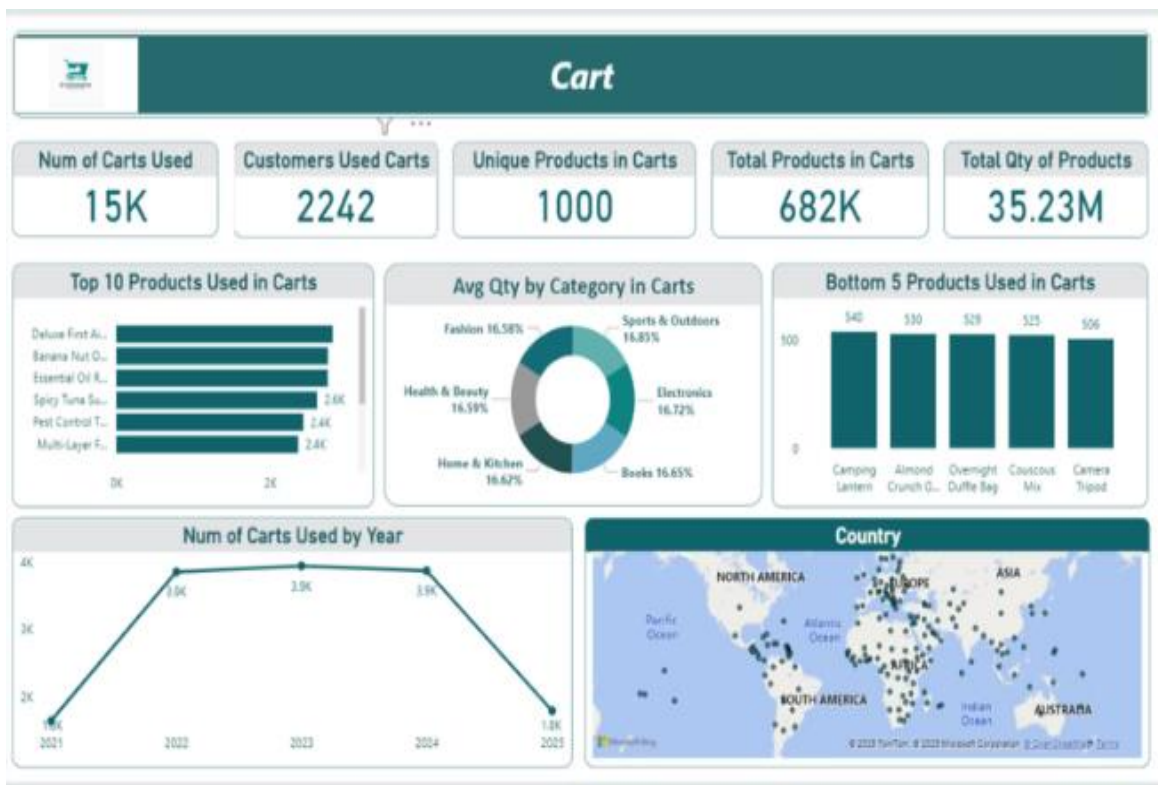
Figure(27)



Figure(28)



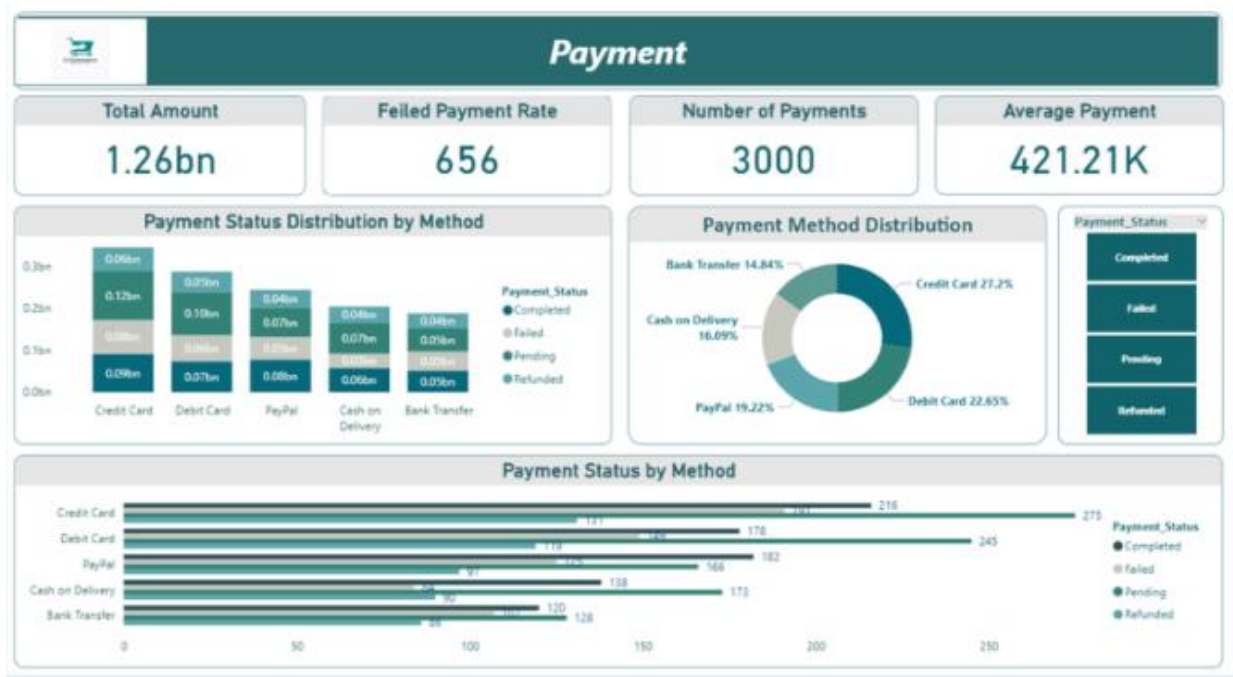
Figure(29)



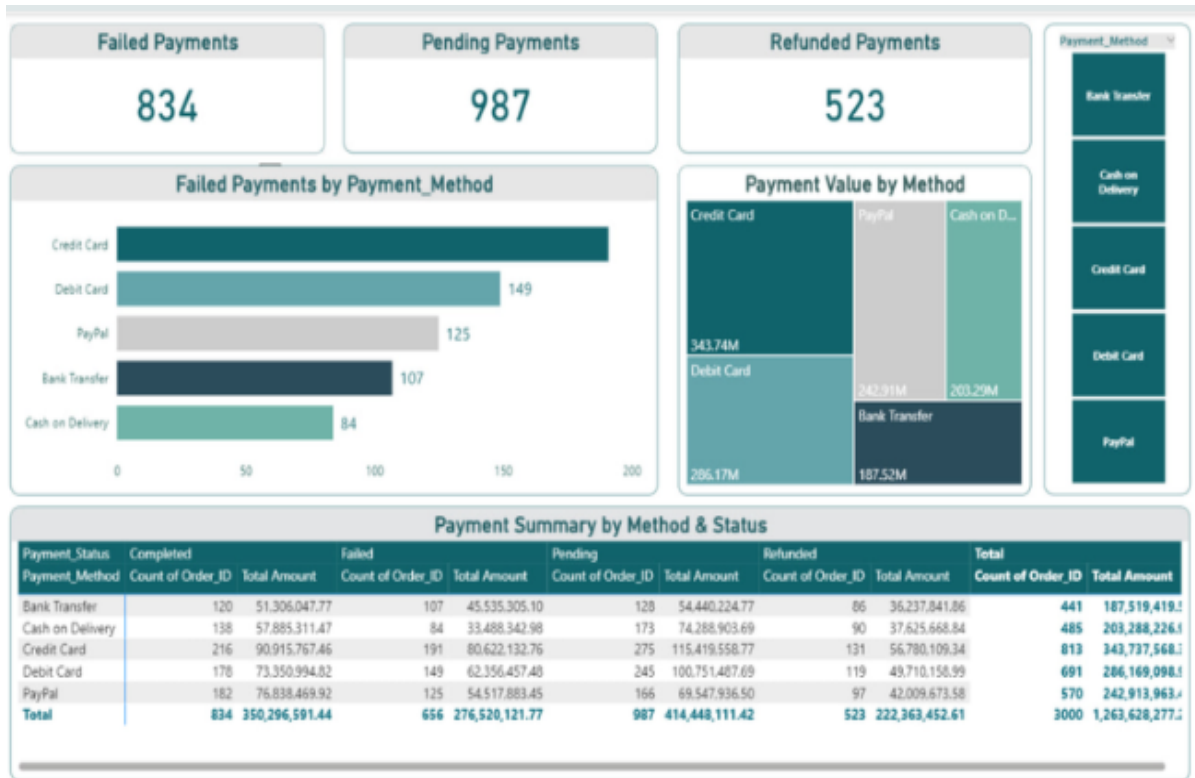
Figure(30)



Figure(31)



Figure(32)



Figure(33)



Figure(34)

```
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
```

```
df = pd.read_csv("/cust_segmentation.csv")
df['orderDate'] = pd.to_datetime(df['orderDate'])
```

```
df.shape
df.head(10)
```

```
snapshot_date = df['orderDate'].max() +
pd.Timedelta(days=1)
```

```
rfm = df.groupby('CustomerSK').agg({
    'orderDate': lambda x: (snapshot_date -
x.max()).days,
    'Order_ID': 'nunique',
    'Total_Amount': 'sum'
}).reset_index()

rfm.columns = ['CustomerSK', 'Recency', 'Frequency',
'Monetary']
```

```
scaler = StandardScaler()
```

```
rfm_scaled = scaler.fit_transform(rfm[['Recency',  
'Frequency', 'Monetary']])
```

```
kmeans = KMeans(n_clusters=4, random_state=42)  
rfm['Segment'] = kmeans.fit_predict(rfm_scaled)
```

```
cluster_summary = rfm.groupby('Segment')[['Recency',  
'Frequency', 'Monetary']].mean()  
print(cluster_summary)
```

```
segment_summary = rfm.groupby('Segment').agg({  
    'Recency': 'mean',  
    'Frequency': 'mean',  
    'Monetary': 'mean',  
    'CustomerSK': 'count'  
}).rename(columns={'CustomerSK': 'Count'})  
print(segment_summary)
```

```
rfm['Cluster_Name'] = rfm['Segment'].map({  
    0: 'Inactive Customers',  
    1: 'VIP Customers',  
    2: 'Loyal Customers',  
    3: 'Potential Returning Customers'  
})
```



```
print(rfm[['CustomerSK', 'Segment',
'Cluster_Name']].head())
```

```
rfm.to_csv("rfm_segments_named.csv", index=False)
```

```
df = pd.read_csv("rfm_segments_named.csv")
df.head(10)
```



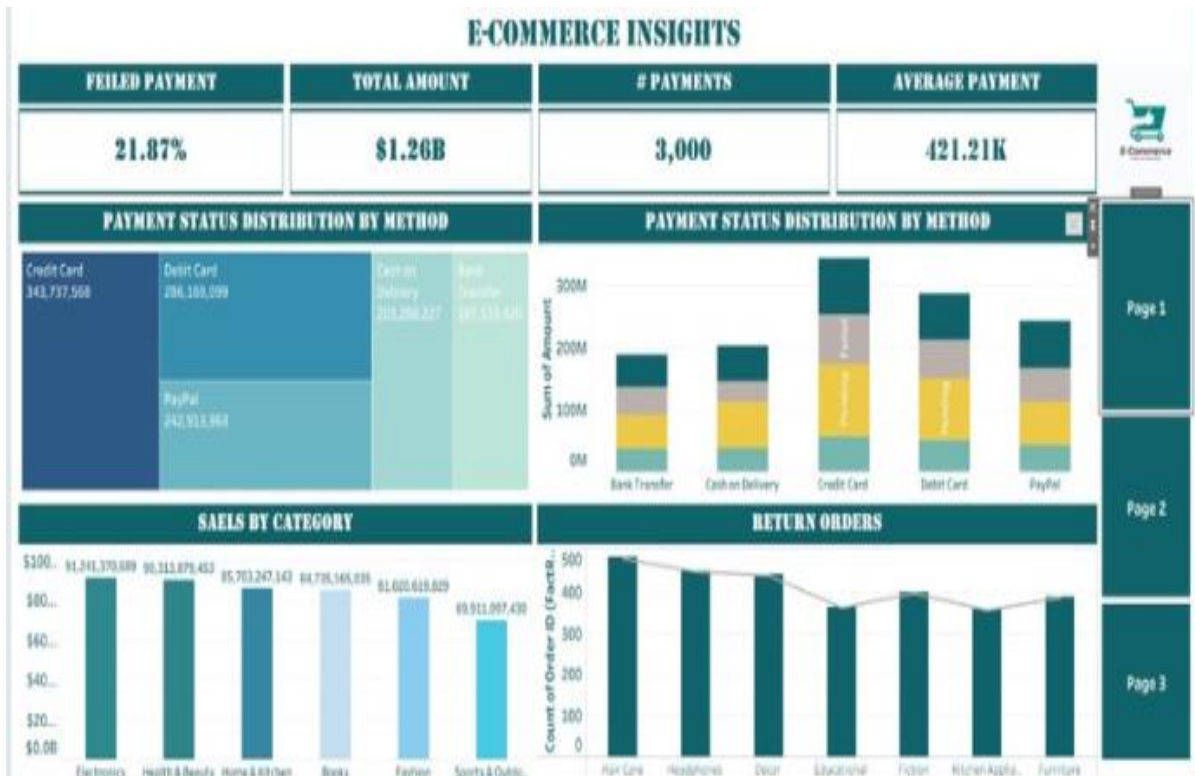
Figure(35)

6.2 Tableau:

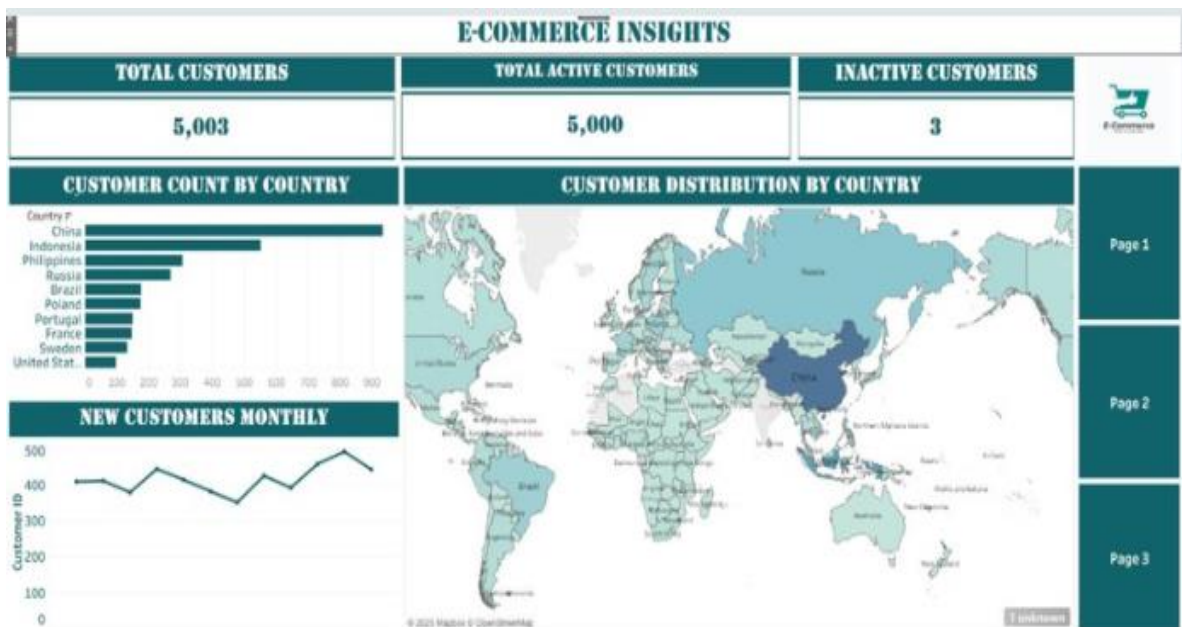
Tableau is a leading data visualization and business intelligence tool that allows users to connect to a wide variety of data sources, transform data, and create interactive dashboards and reports. It offers intuitive drag-and-drop functionality, enabling both technical and non-technical users to build insightful visualizations without extensive coding knowledge. Tableau supports real-time data analysis, advanced analytics, and integration with various platforms, making it a powerful solution for uncovering trends, patterns, and actionable insights to support informed business decisions.



Figure(36)



Figure(37)



Figure(38)

Chapter07:

Conclusion

8.1 Future scope :

The future scope of the E-commerce Analysis project includes expanding the dataset to cover multiple years and additional product categories to gain a more comprehensive understanding of customer behavior. Integrating advanced analytics techniques such as predictive modeling and machine learning can enhance sales forecasting and personalized marketing strategies. Real-time data integration from multiple sources, including social media and customer feedback platforms, can further improve the accuracy of insights. Additionally, developing automated reporting dashboards will enable stakeholders to monitor key performance indicators continuously and make faster, data-driven decisions.

8.2 Conclusion

The E-commerce Analysis project has successfully provided valuable insights into sales performance, customer behavior, and product trends. By transforming raw transaction data into meaningful reports and visualizations, the project has enabled stakeholders to identify growth opportunities and address performance gaps. The integration of data analytics tools has streamlined decision-making processes, while visual dashboards have improved data accessibility for both technical and non-technical users. Overall, this project serves as a strong foundation for implementing advanced analytics in the future to further optimize business performance and customer engagement.

Reference

[1] Investopedia. (2024, September 24). E-commerce Defined: Types, History, and Examples. Retrieved from <https://www.investopedia.com/terms/e/ecommerce.asp>

[2] Scandiweb. (2022, September 18). What Is a Data Warehouse and How Can You Use It for eCommerce?. Retrieved from <https://scandiweb.com/blog/data-warehouses-for-ecommerce-definition-examples-benefits/>

[3] ThoughtSpot. (2023, February 9). What is Dimensional Data Modeling? Examples, Tips, and More. Retrieved from <https://www.thoughtspot.com/data-trends/data-modeling/dimensional-data-modeling>

[4] Microsoft Learn. (2025, June 16). SQL Server Reporting Services (SSRS). Retrieved from <https://learn.microsoft.com/enus/sql/reporting-services/create-deploy-and-manage-mobile-and-paginated-reports?view=sql-server-ver16>

[5] Microsoft Power BI. (n.d.). E-Commerce Dashboard. Retrieved from <https://app.powerbi.com/view?r=eyJrIjoiaW50ZW50ZjctZWVjNS00MGM0LWVlNTAtYzAzYWE0OGNmOWUwliwidCI6ImNiNDg0NDZILTkwZTYtNGJmMS00MjViLTQwZTQ>

[6] Tableau. (n.d.). 5 best practices for analyzing e-commerce data – and the 7 mistakes to avoid. Retrieved from <https://www.tableau.com/learn/whitepapers/5-best-practices-analyzing-ecommerce-data>