

**Technical Specification Document**

**Sales Data Analysis and Data Modelling to support Business Requirements**

**Author - Shantanu Zodey**

**Date – 30/05/2025**

# Objective

The purpose of this project is to transform raw orders, customer, and shipping data from different formats (Excel, CSV, JSON) into a unified data model that supports advanced business reporting and analytics. The goal is to enable the business to gain valuable insights into customer behavior, product performance, shipping efficiency, and geographic trends, which will drive strategic decision-making.

By leveraging SQL-based data modeling, this project aims to ensure data consistency, integrity, and traceability, while preparing the dataset for scalable reporting across key metrics like total sales, customer transactions, product popularity, and delivery statuses.

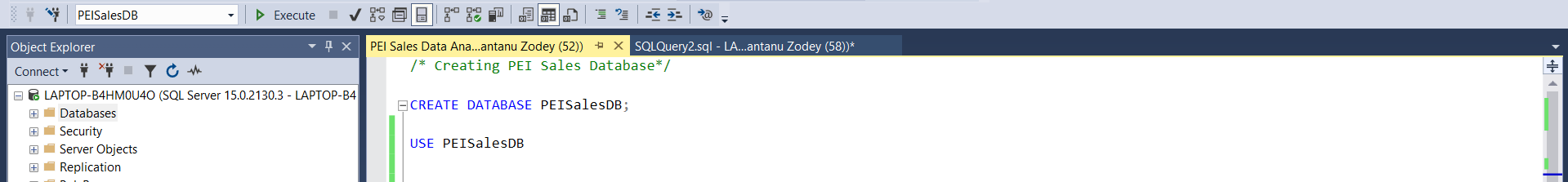
# Project Scope

This project includes the following activities:

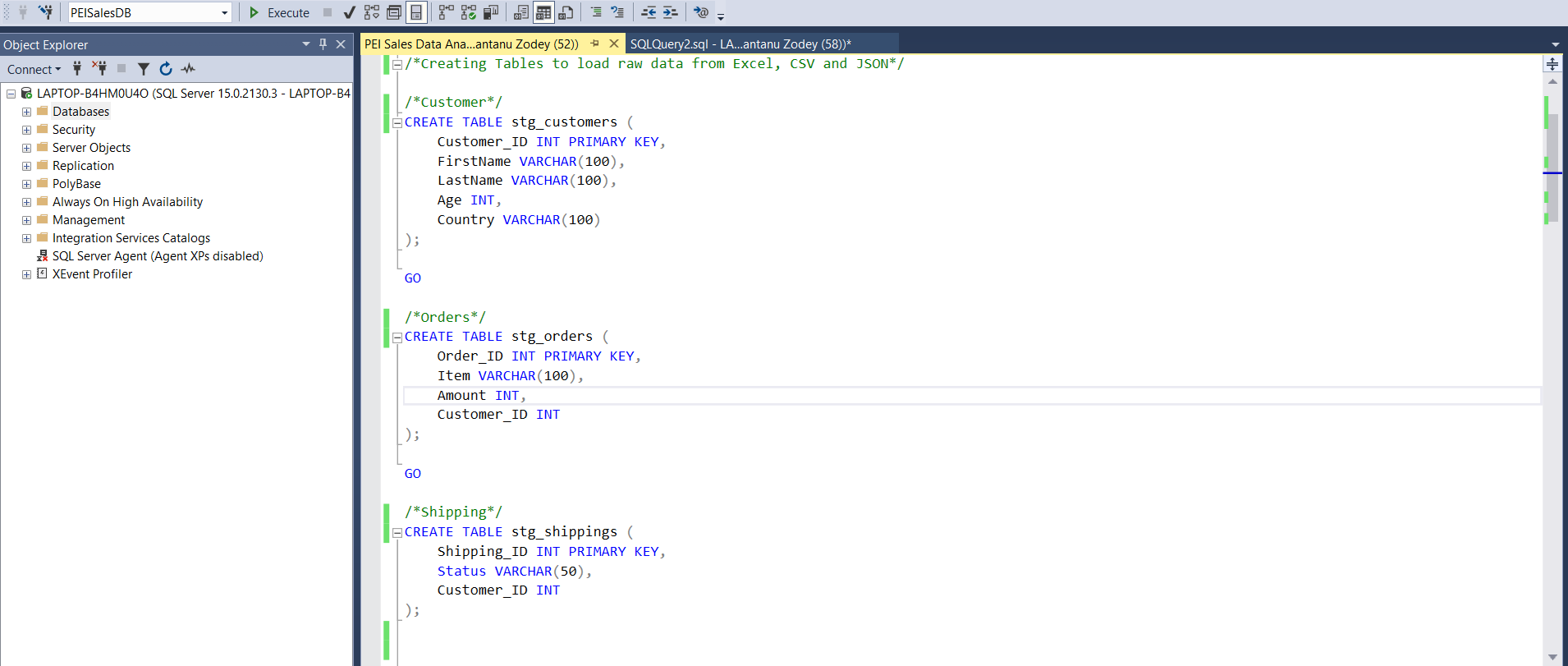
1. **Source Data Integration**
   * Ingesting data from three sources:
     + Customer (Excel)
     + Orders (CSV)
     + Shipping (JSON)
2. **Data Quality Checks**
   * Performing completeness, accuracy, and referential integrity checks using SQL.
3. **Data Modeling**
   * Designing a star schema using dimension and fact tables.
   * Resolving inconsistencies in item prices and categorize customers by age.
4. **ETL Logic Development**
   * Defining and implement transformations to populate the target model.
5. **Enable Reporting Requirements**
   * Structure data to support five business reporting use cases including:
     + Total spent per country with pending deliveries
     + Product performance by age category
     + Transaction volume and sales per customer
     + Country with lowest sales and transaction count

# Source Data Integration

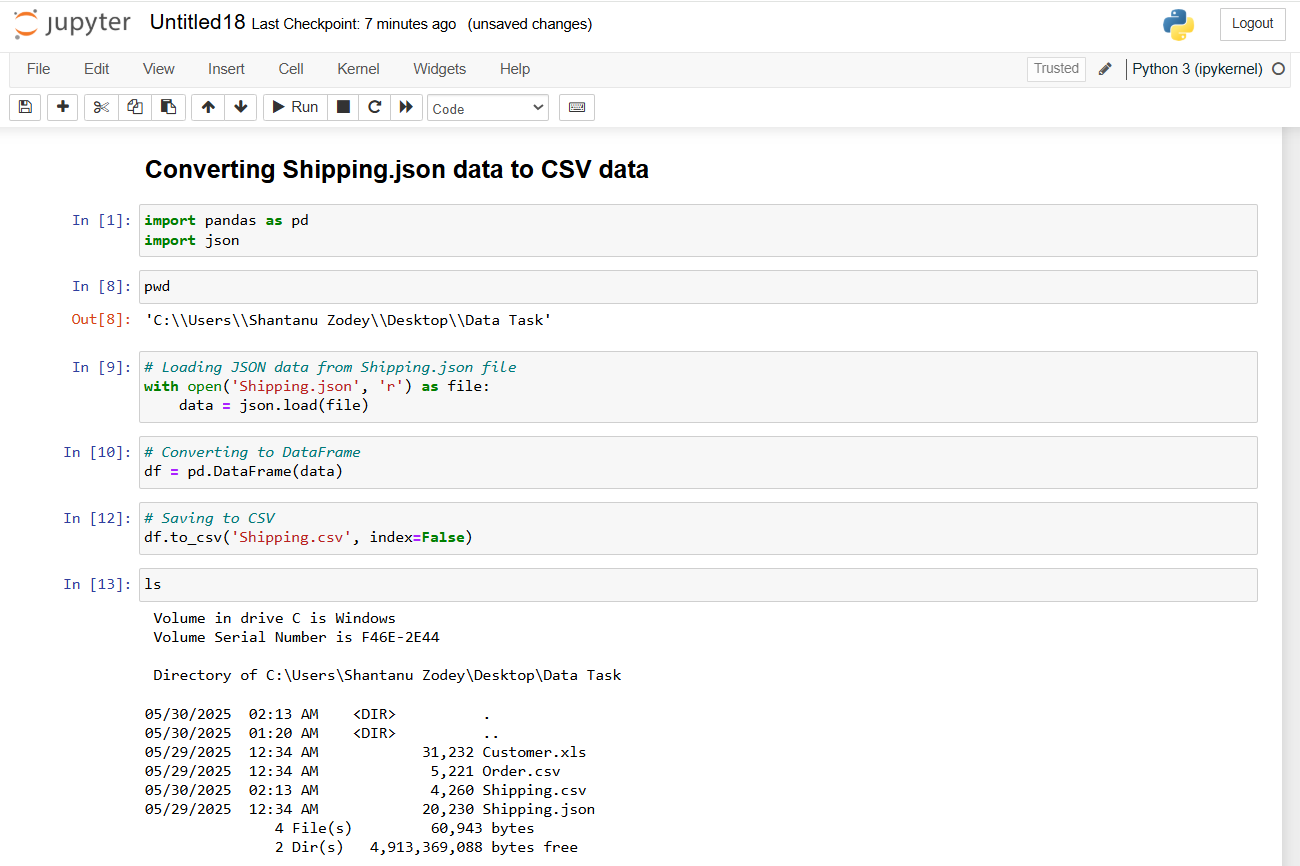
1. Creating Database PEISalesDB



1. Creating Staging Tables for Customers, Orders and Shipping.

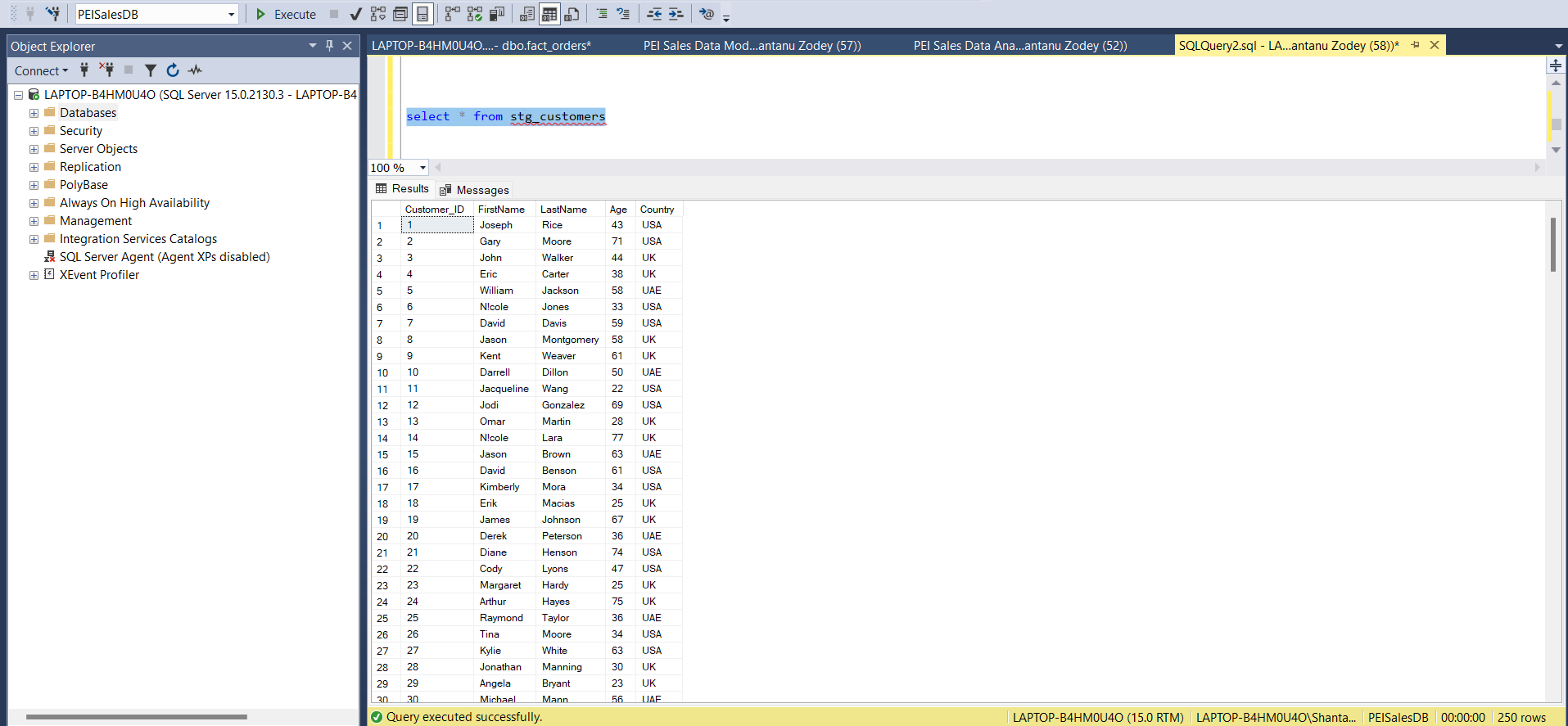


* The above staging tables will be used to load raw data and perform exploratory data analysis, data quality, data consistency and data accuracy checks.
* Once the staging tables are created, we loaded the data for Customer data from Excel and Orders data from CSV using Import Data functionality in Microsoft SQL Server. As Shipping data is in JSON format, Microsoft SQL Server does not support Import Data functionality for JSON file format. So, to convert JSON data into CSV data, we leveraged python and converted shipping.json data to shipping.csv file and then loaded the shipping data using Import Data functionality. Please find below snapshot.

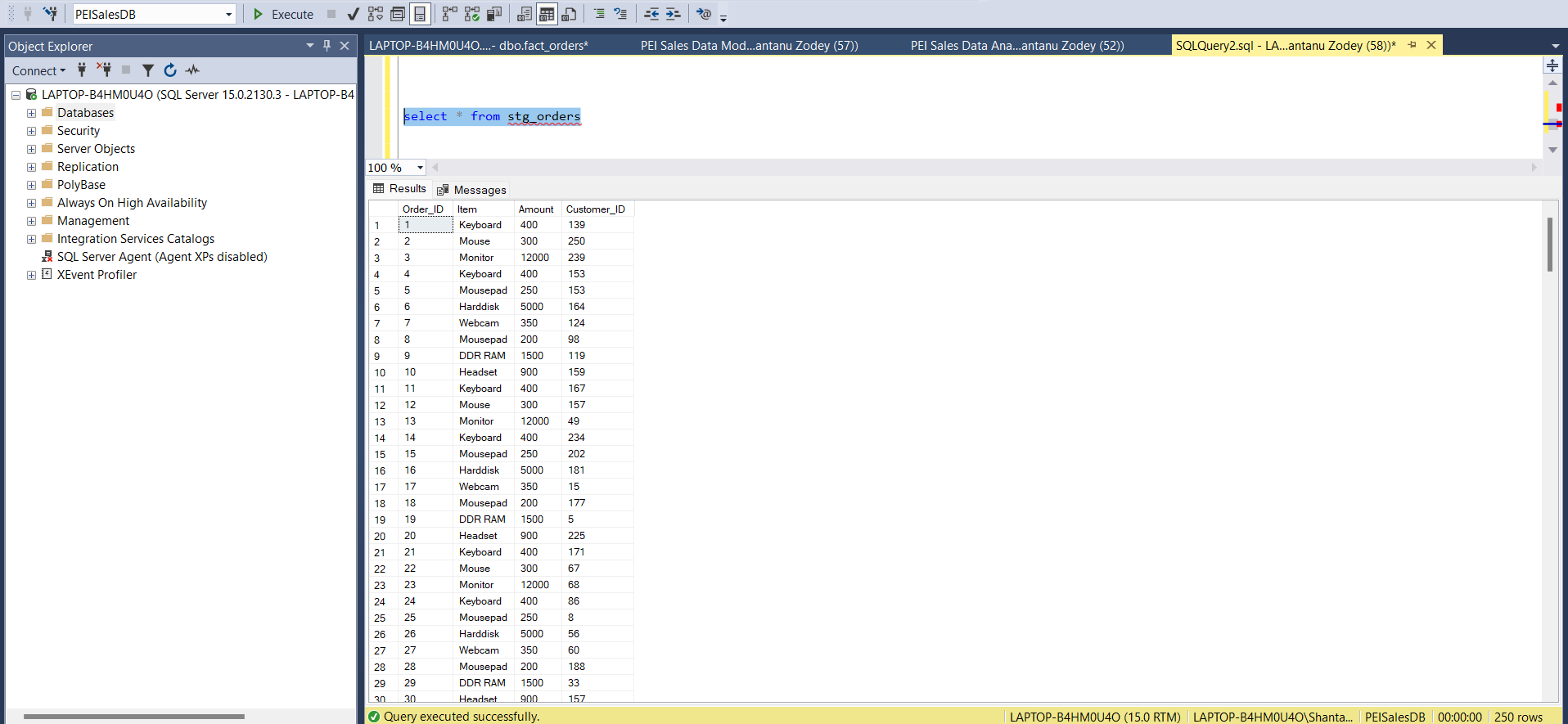


1. Validating all the records and counts to ensure all records are loaded into staging table.

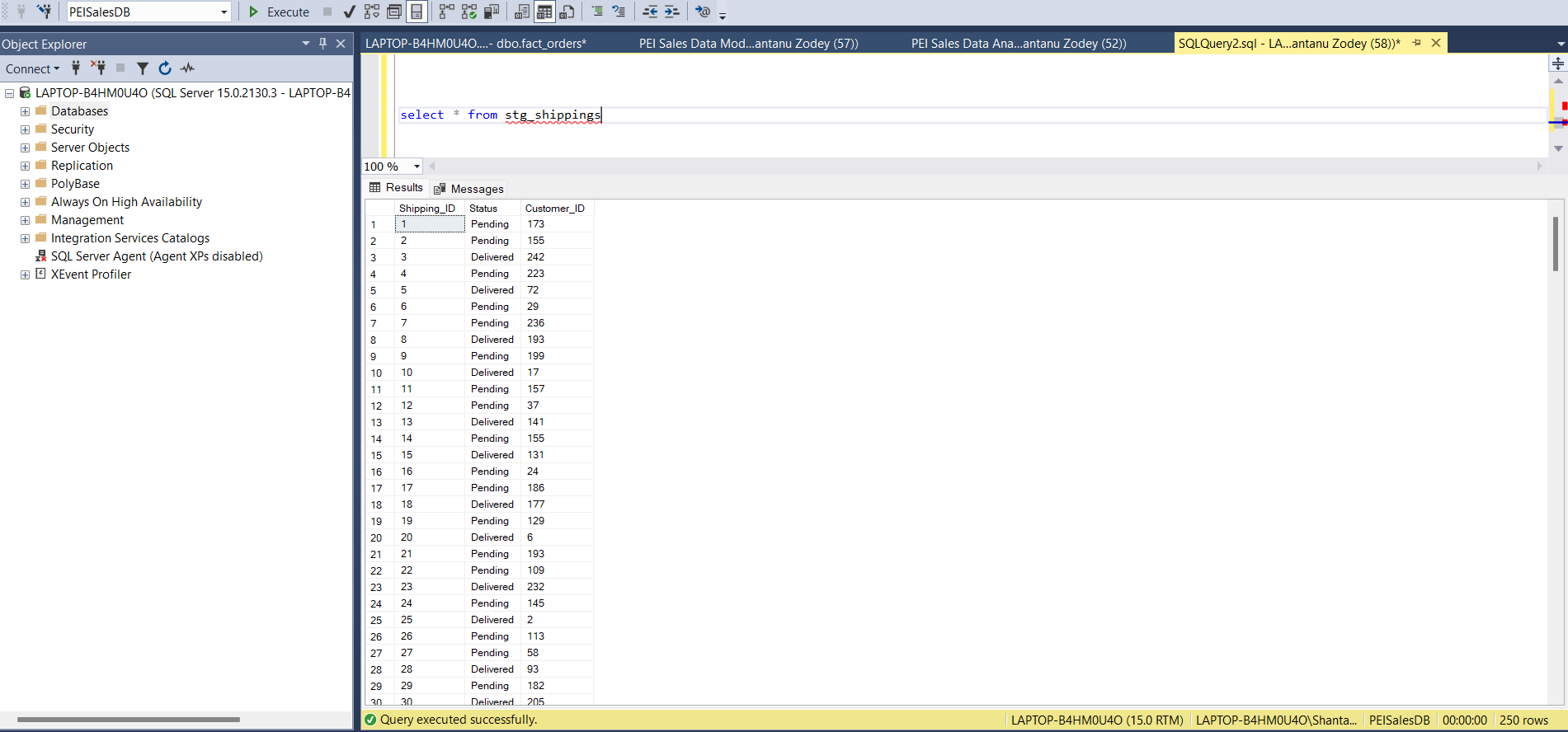
**Stg\_customer**



**stg\_orders**

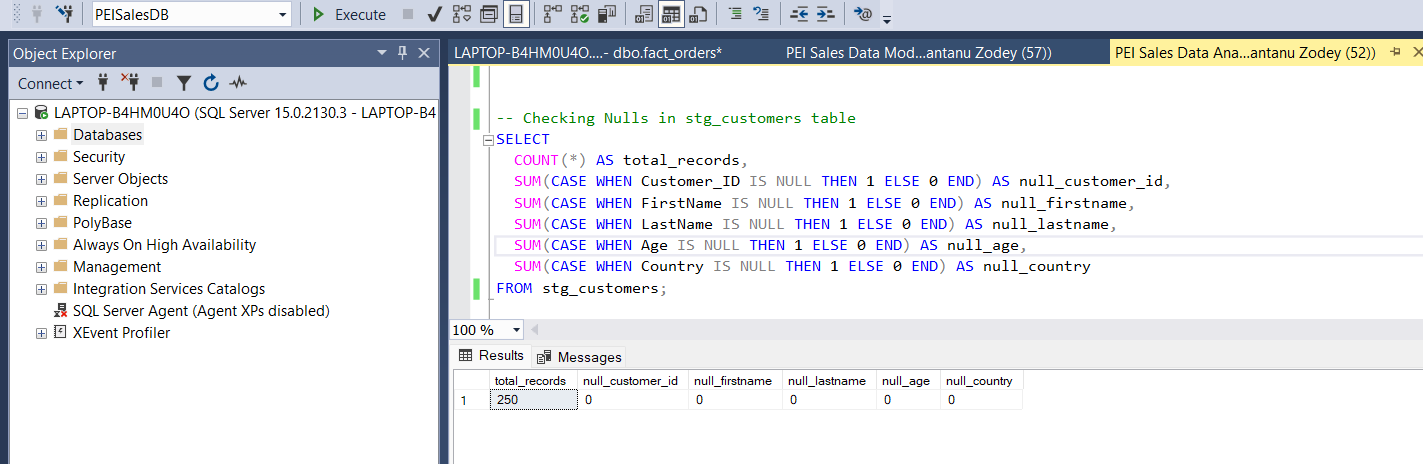


**stg\_shippings**

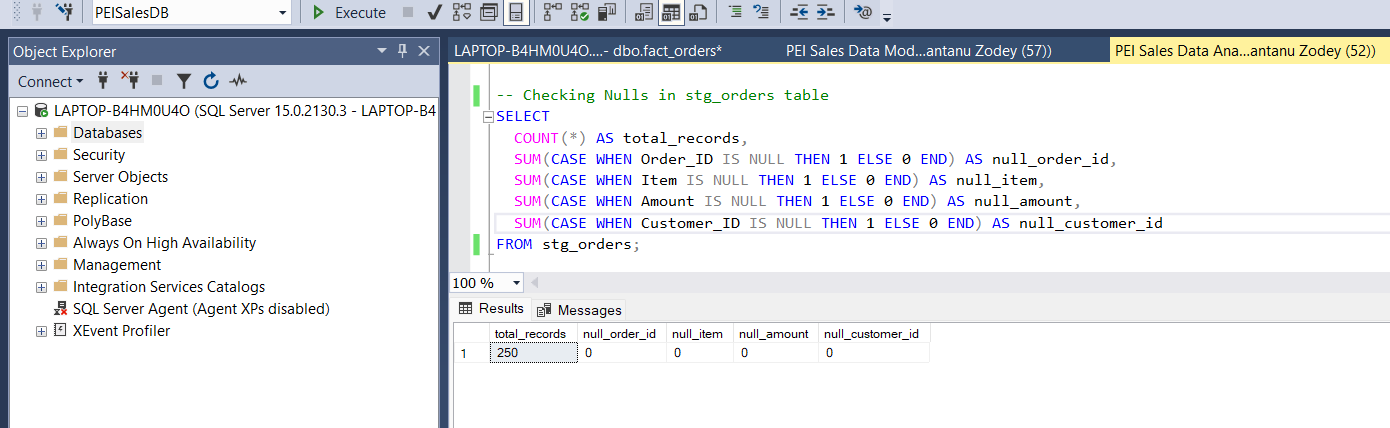


# Data Quality Checks

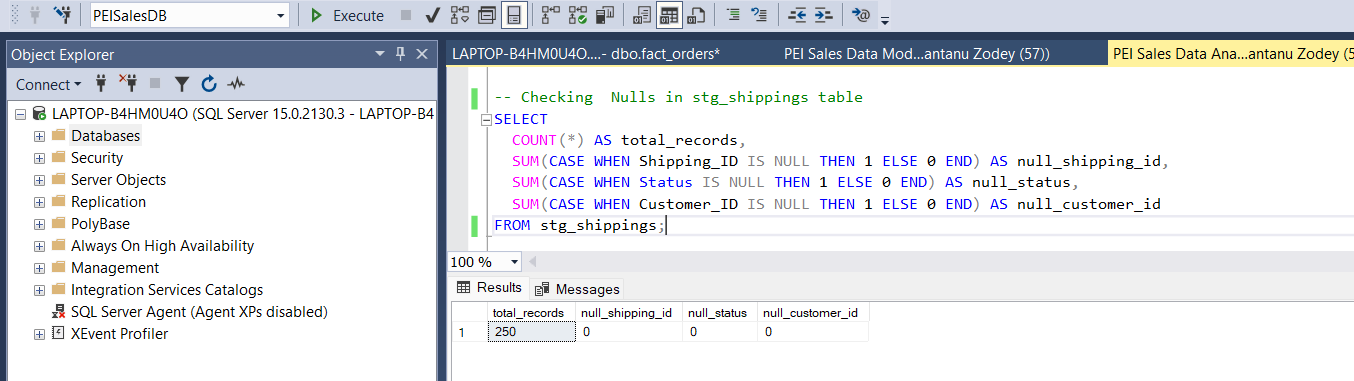
1. Checking Nulls in stg\_customer table



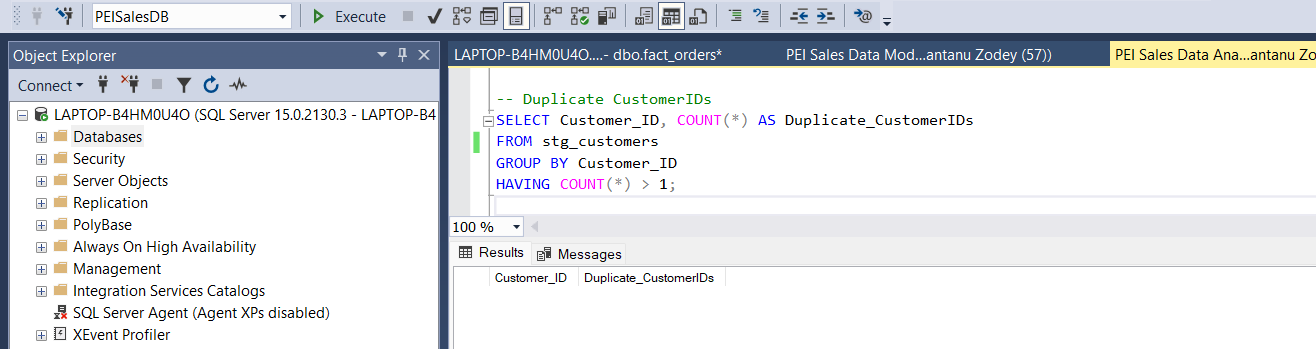
1. Checking Nulls in stg\_orders table



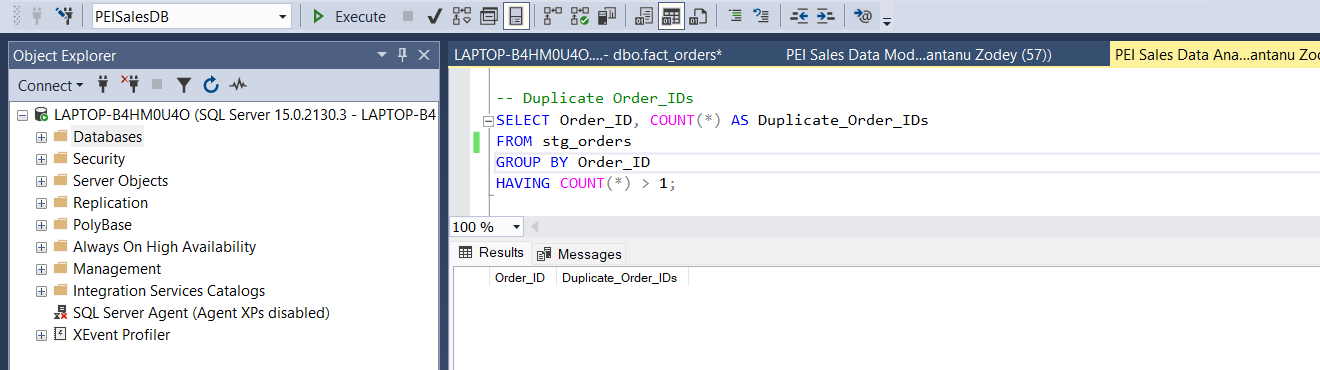
1. Checking Nulls in stg\_shipping table



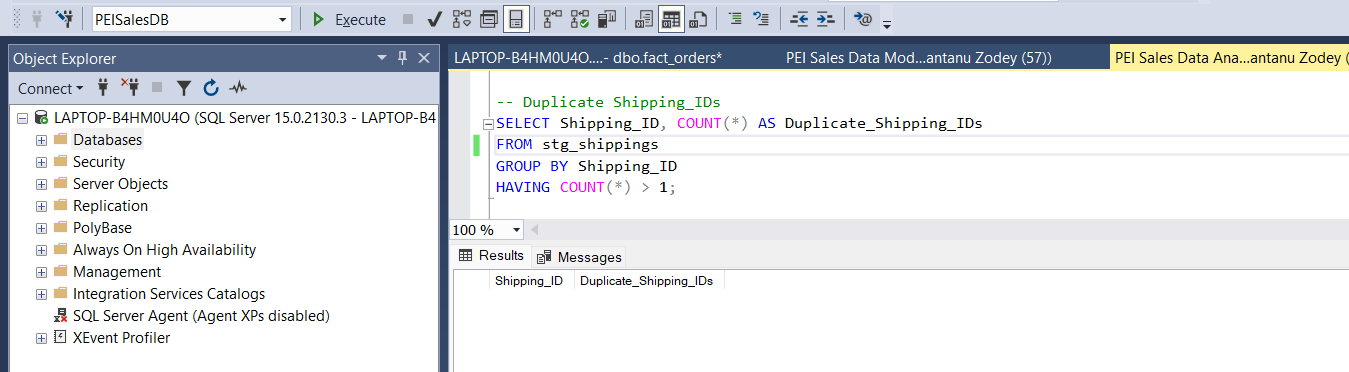
1. Checking Duplicate Customer IDs in stg\_customer table



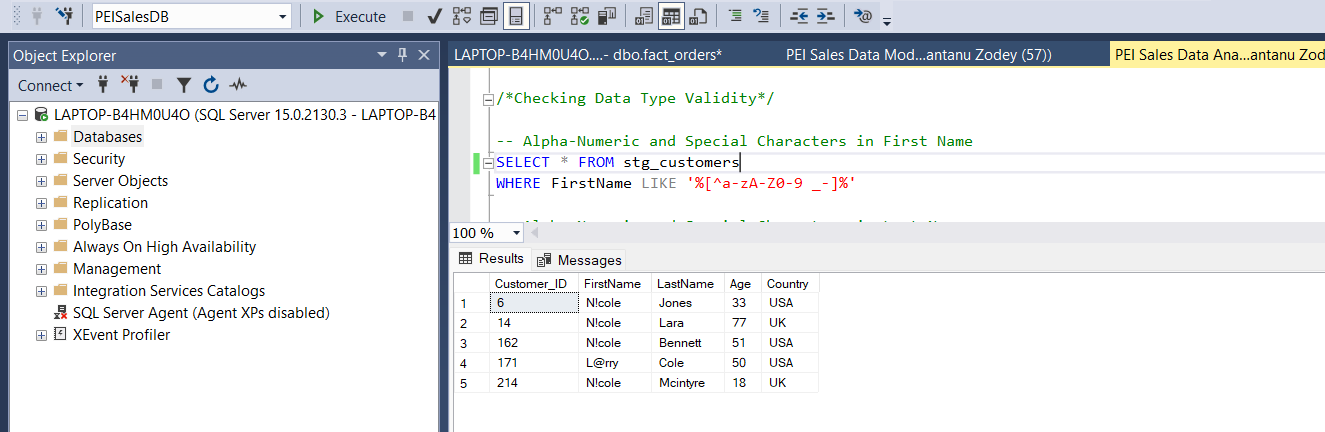
1. Checking Duplicate Order IDs in stg\_orders table



1. Checking Duplicate Shipping IDs in stg\_shipping table

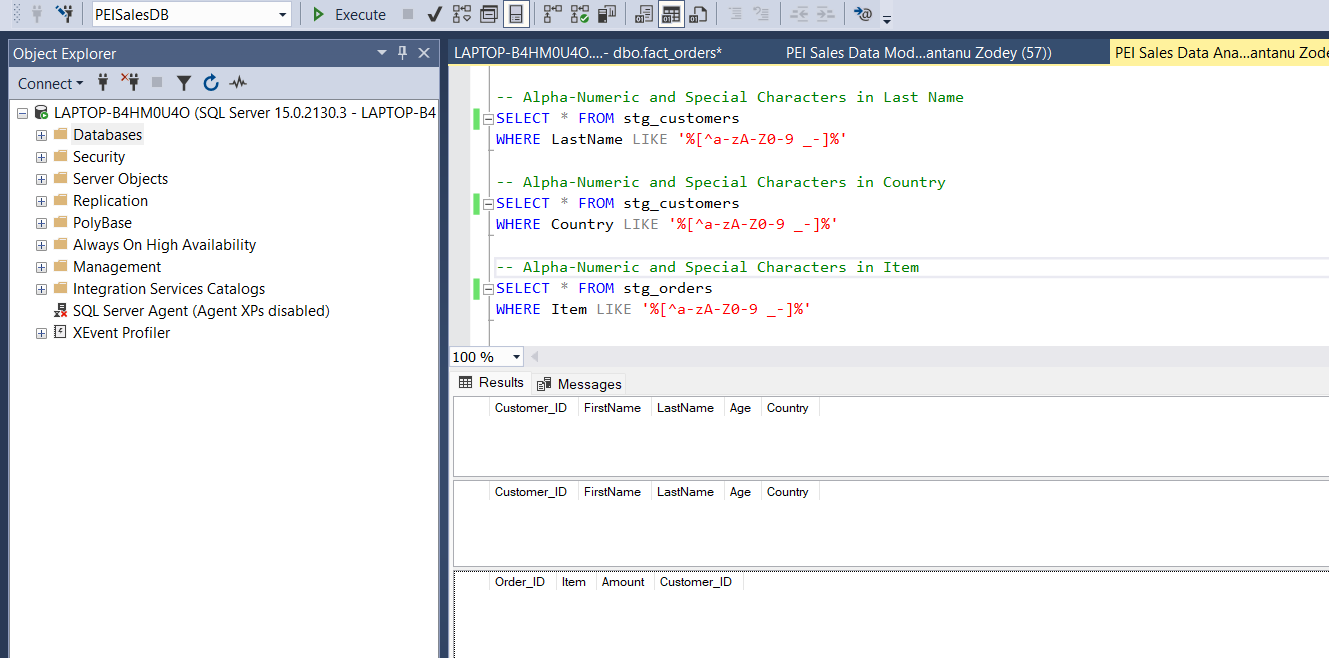


1. Checking Alpha-Numeric and Special characters in First Name

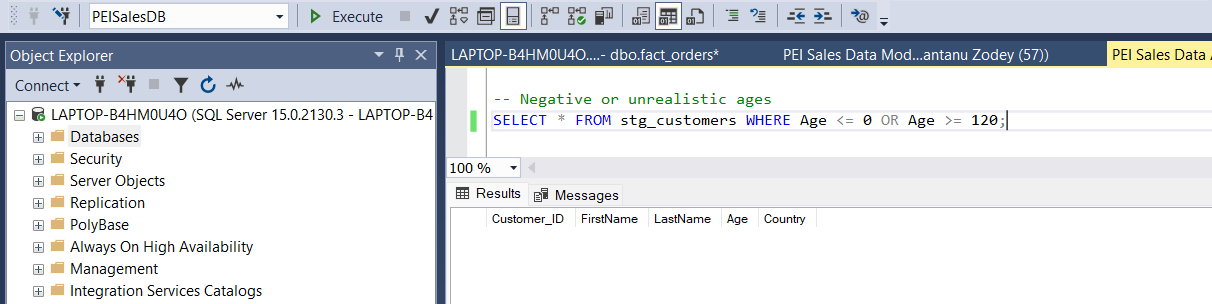


In the above screenshot, there are few records in First Name that contains special characters. This column needs transformation in order to get the clean and accurate data.

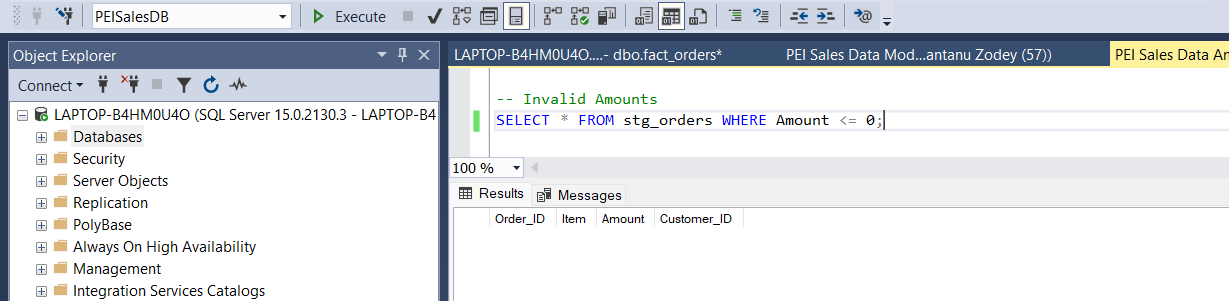
1. Checking Alpha-Numeric and Special characters in Last Name, Country and Item



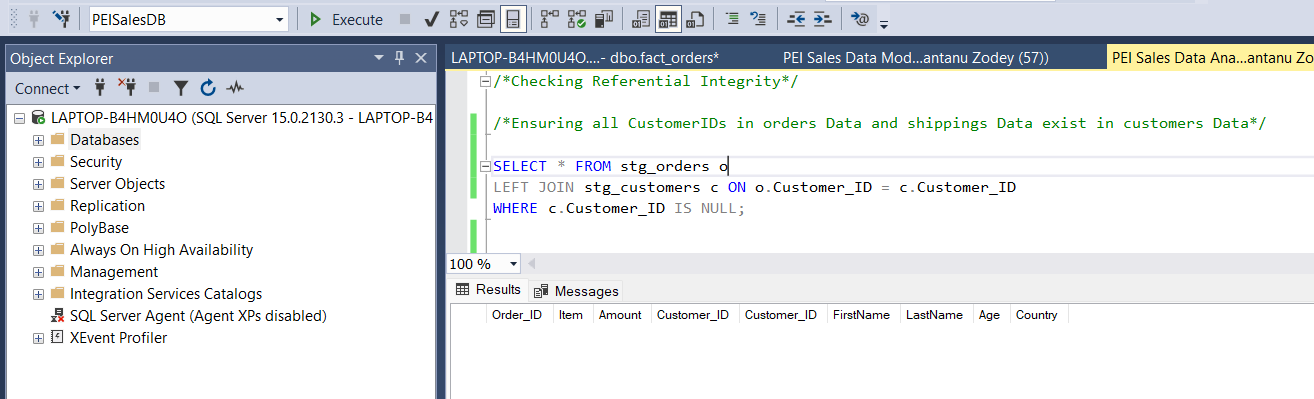
1. Checking Negative or Unrealistic Ages.



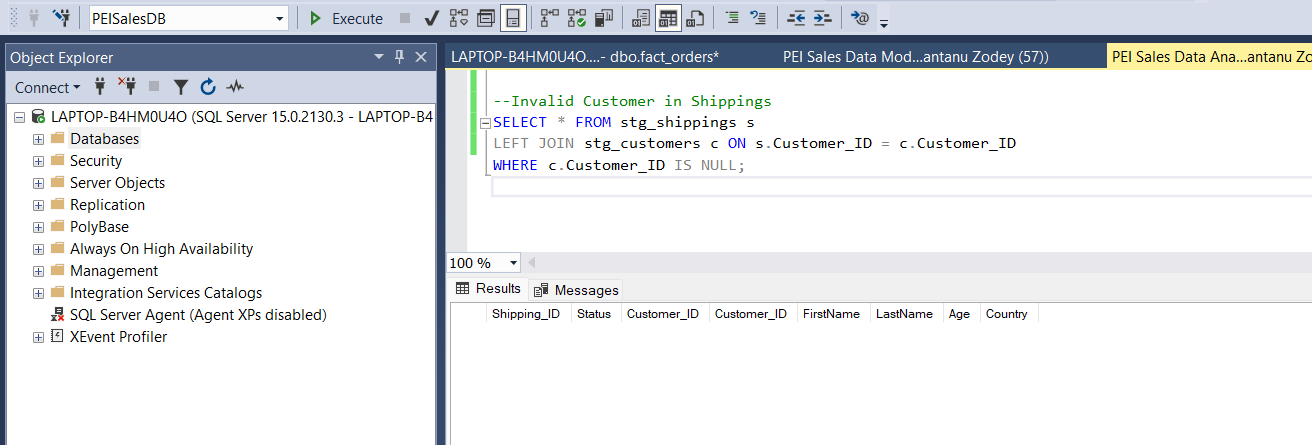
1. Checking Invalid Amounts



1. Referential Integrity Checks for valid customers present in stg\_orders



1. Referential Integrity Checks for valid customers present in stg\_shippings



**Findings for Data Quality checks:**

* First Name column in stg\_customer table contains special characters
* Duplicate Items with different amounts.
* All datasets are joinable via CustomerID.
* No nulls found in critical fields.
* No duplicates found in critical fields.

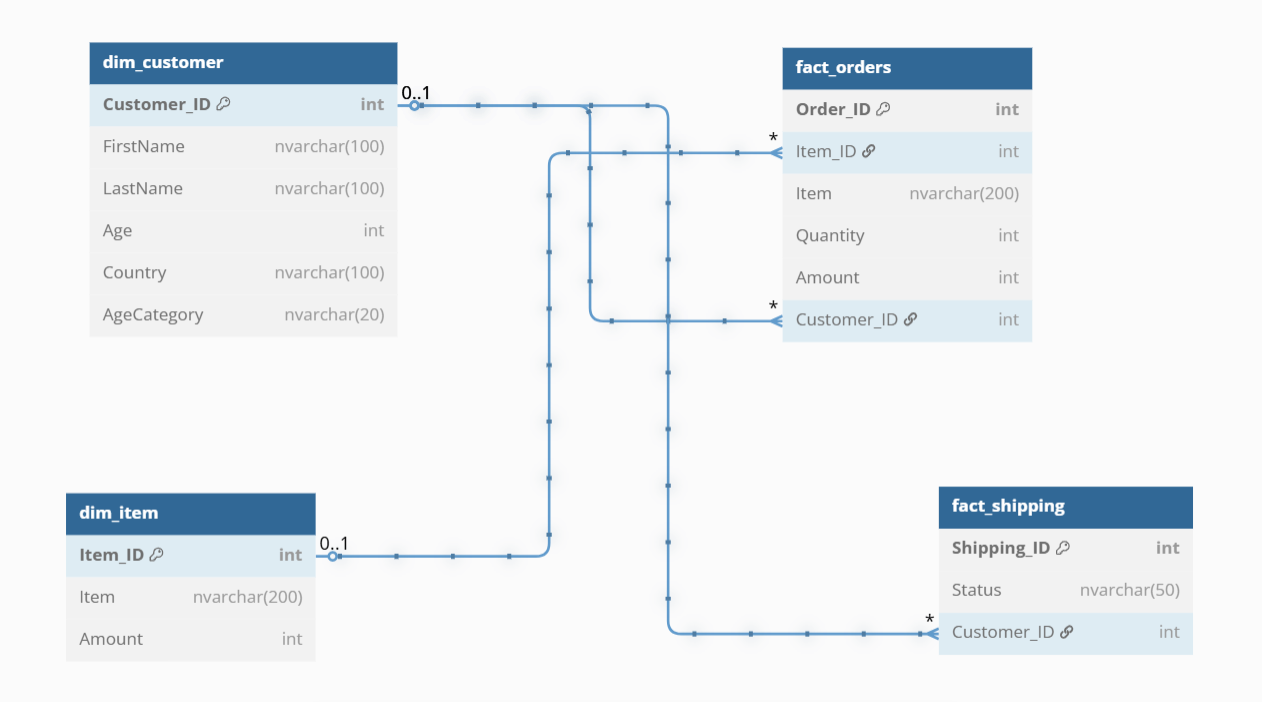
# Data Modelling

To fulfill the Business reporting requirement, we designed a **Star Schema** with:

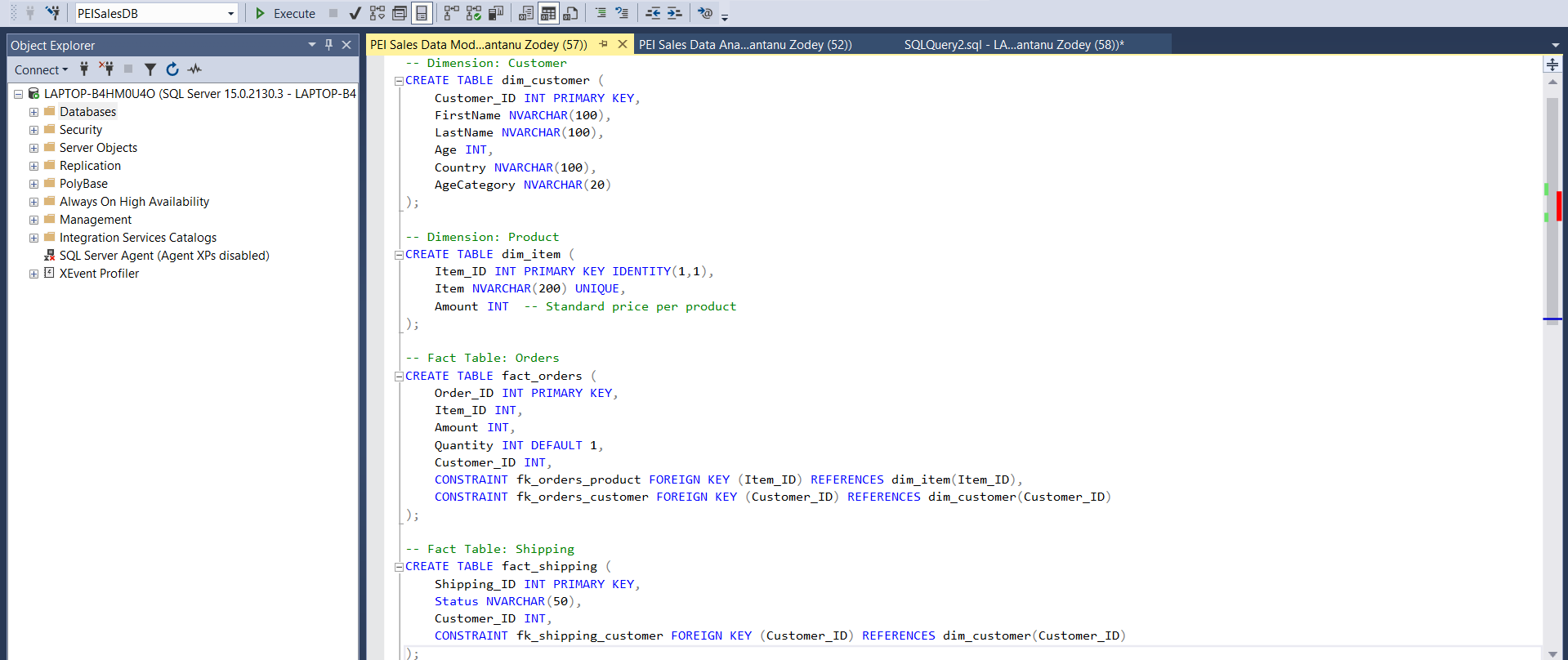
* **2 Dimension Tables**: dim\_customer, dim\_product
* **2 Fact Tables**: fact\_orders, fact\_shipping

Please find below snapshot.

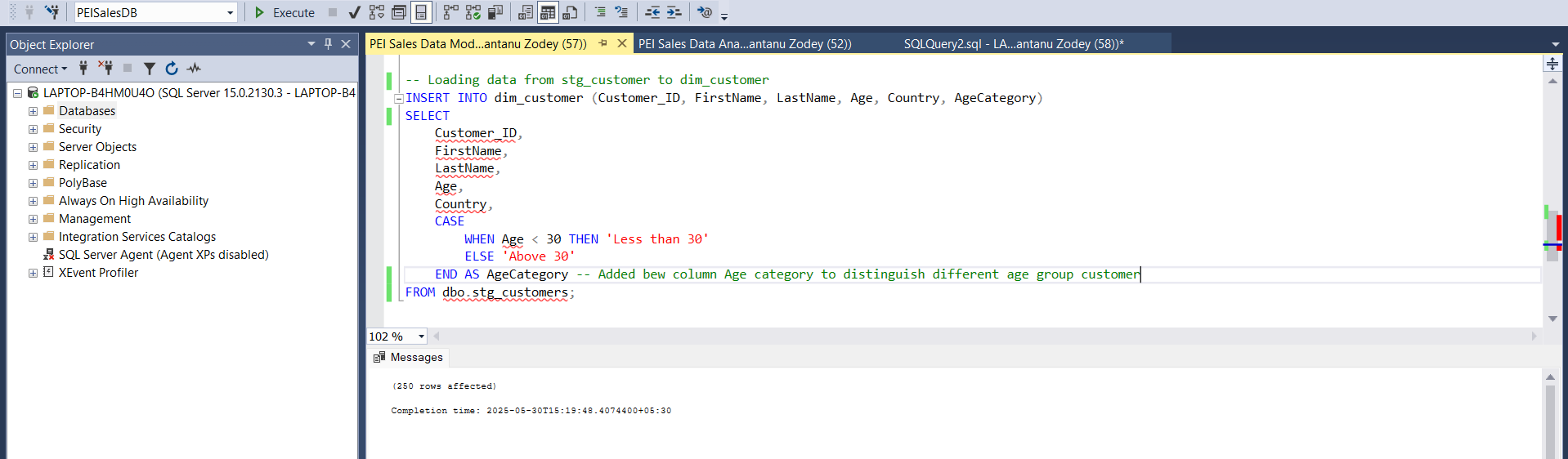
## Domain Model (ERD)

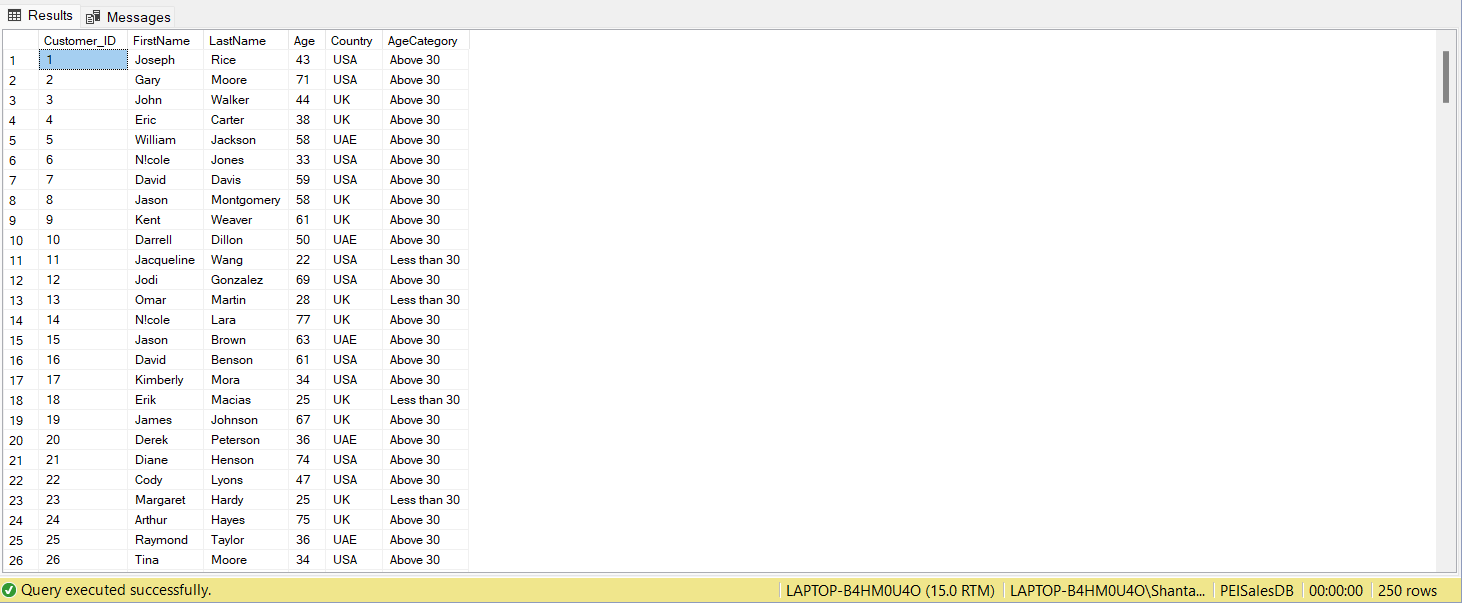


Below are the table schema’s that has been created in Database.

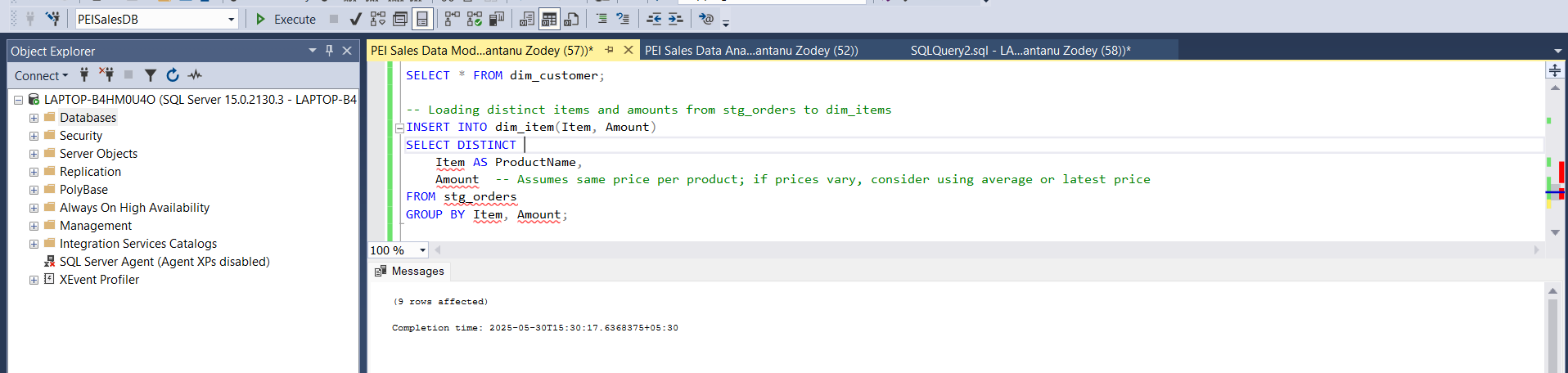


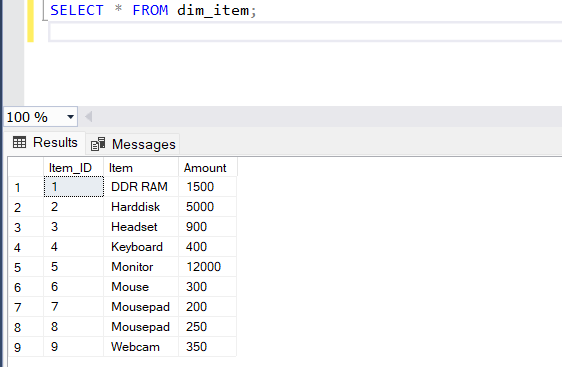
Loading data from stg\_customer to dim\_customer.



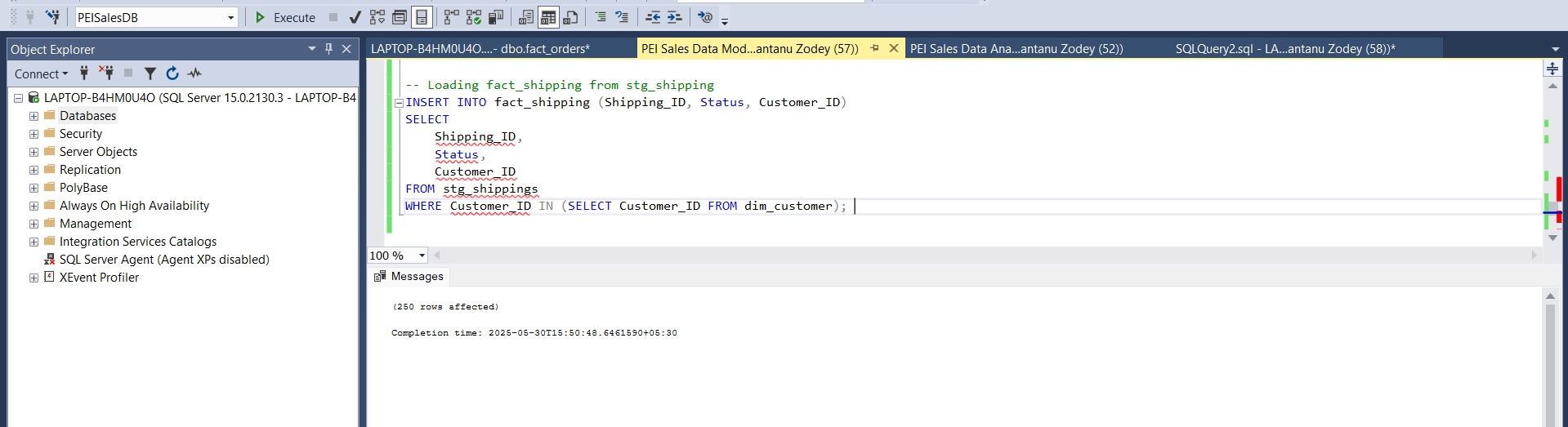


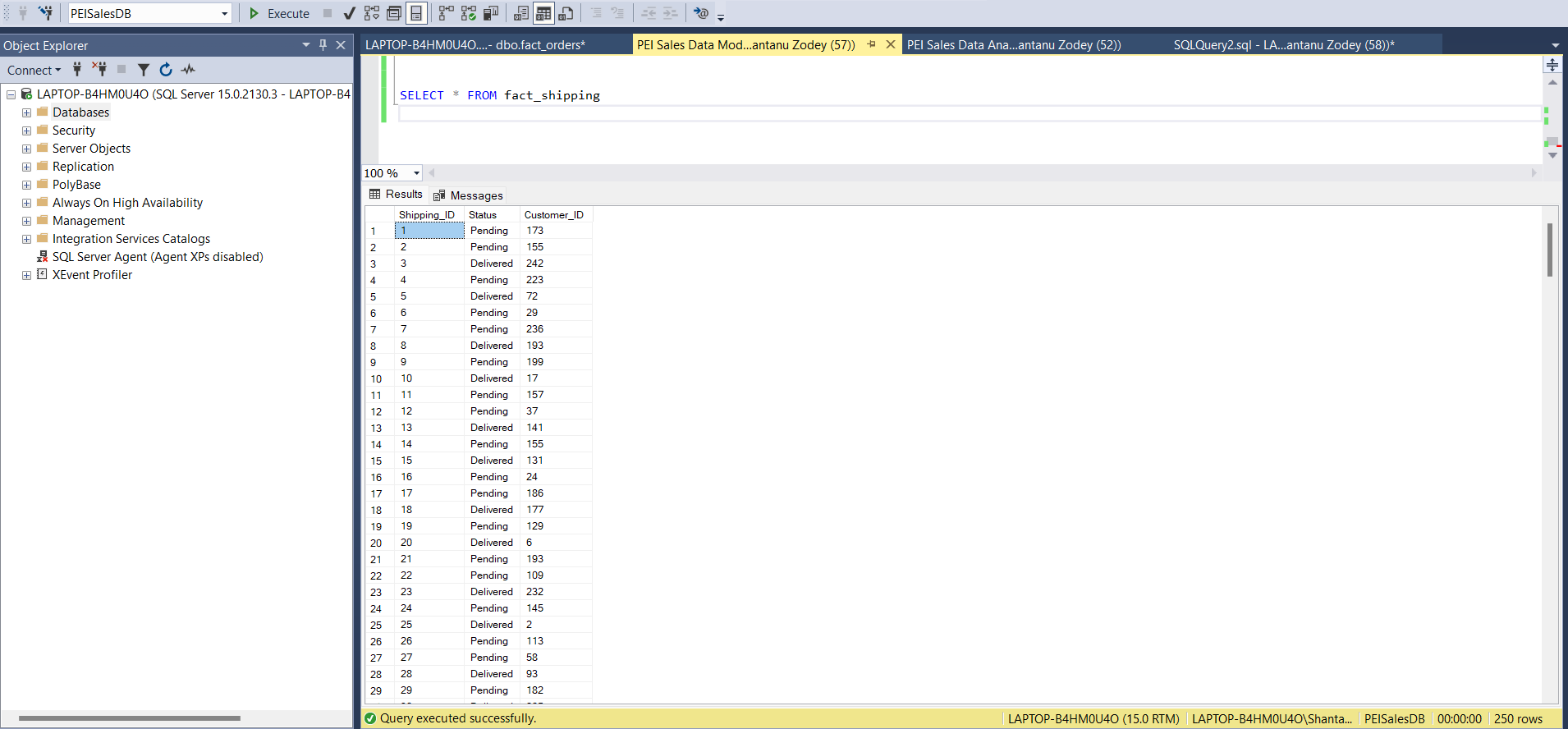
Loading Distinct Items and Amounts from stg\_orders into dim\_items.





Loading data from stg\_shipping to fact\_shipping.





## Loading fact\_order table.

To populate the fact\_orders table for transactional reporting. A Data engineer should create a table in such a way that it captures individual product purchase events linked to customer and item dimensions.

In a way Business stakeholders can analyze total sales, customer purchasing behavior, and product performance across various dimensions such as country, age category, and product type.

**Business Context**

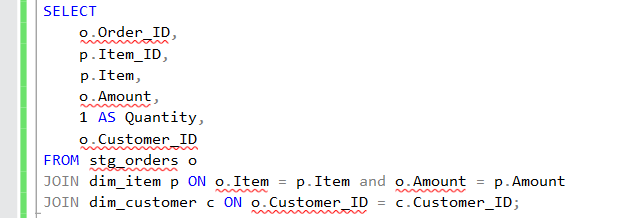
The fact\_orders table will store transactional-level data based on raw order records from the sales team. Each record in the raw orders CSV contains an Order\_ID, Item, Amount, and CustomerID. The goal is to link this to:

* A **product dimension (dim\_product)** for consistent product representation.
* A **customer dimension (dim\_customer)** to associate orders with customer demographics.

This table supports reporting on:

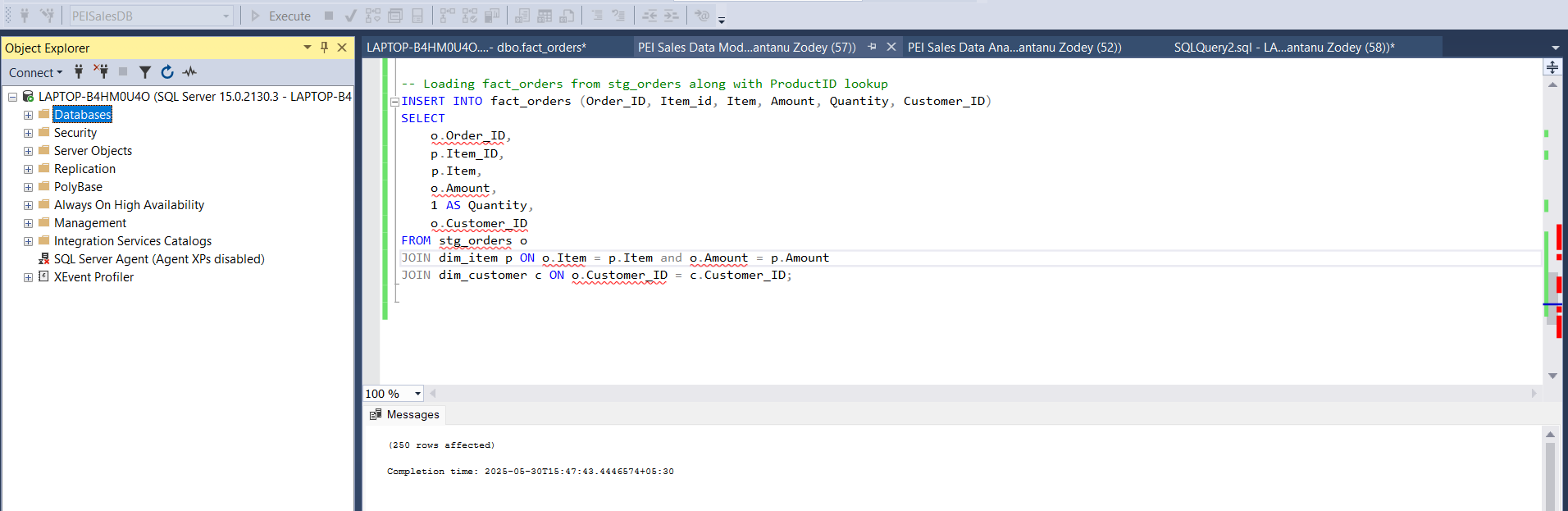
* Total sales by customer
* Product purchases by age category
* Quantity sold per item

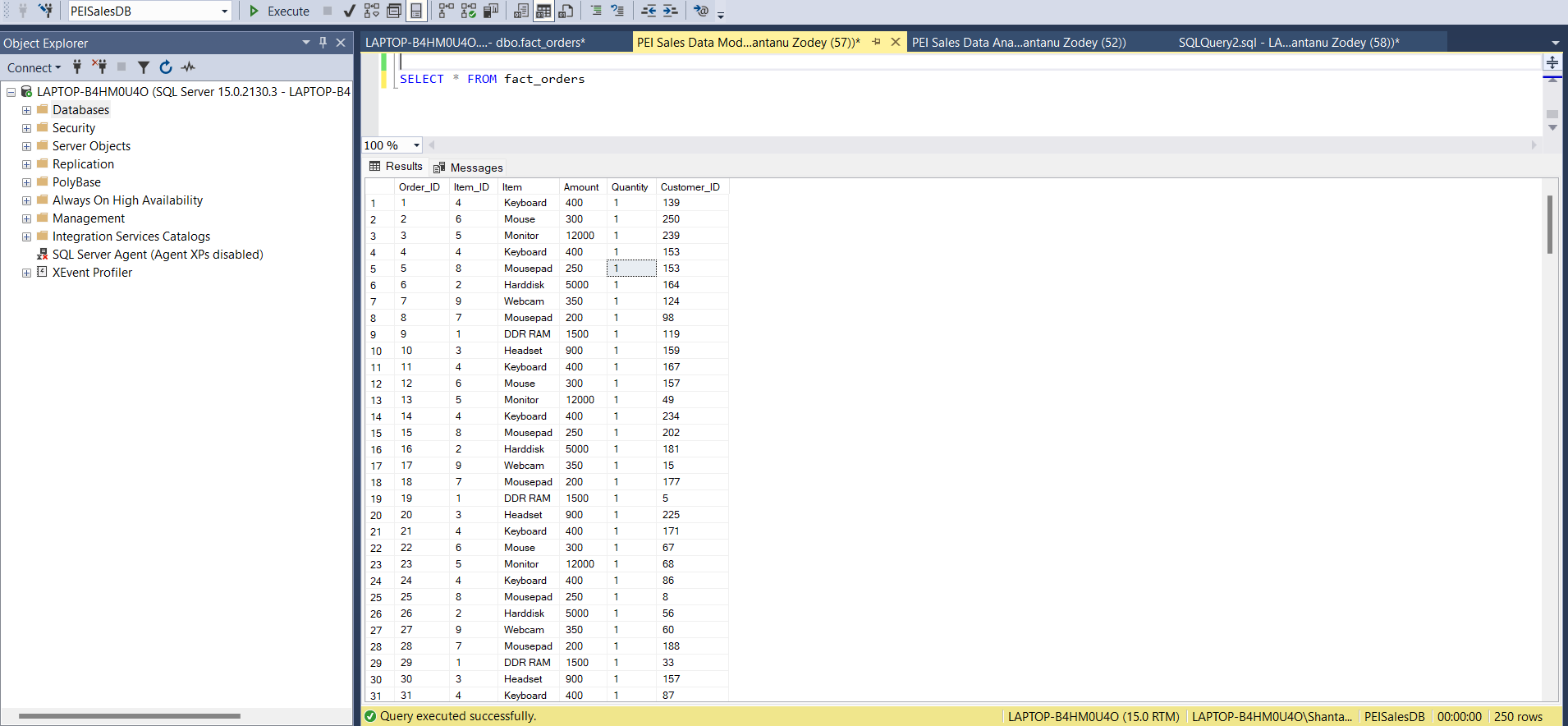
**Transformations**

****

* Join with dim\_item on Item to obtain Item\_ID.
* Validate Customer\_ID exists in dim\_customer.
* Derive Quantity: Hardcoded to 1 per transaction.
* Exclude null or malformed records in Order\_ID, Item, Amount, or CustomerID.

**Implementation**



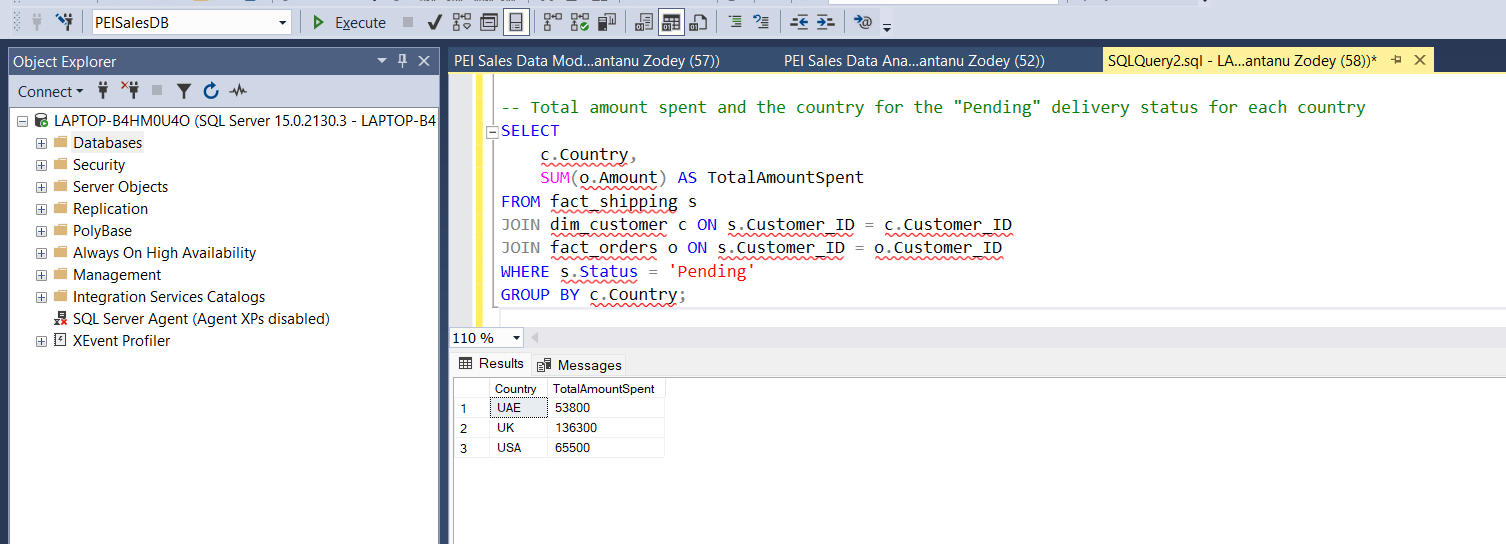


**QA Test Criteria**

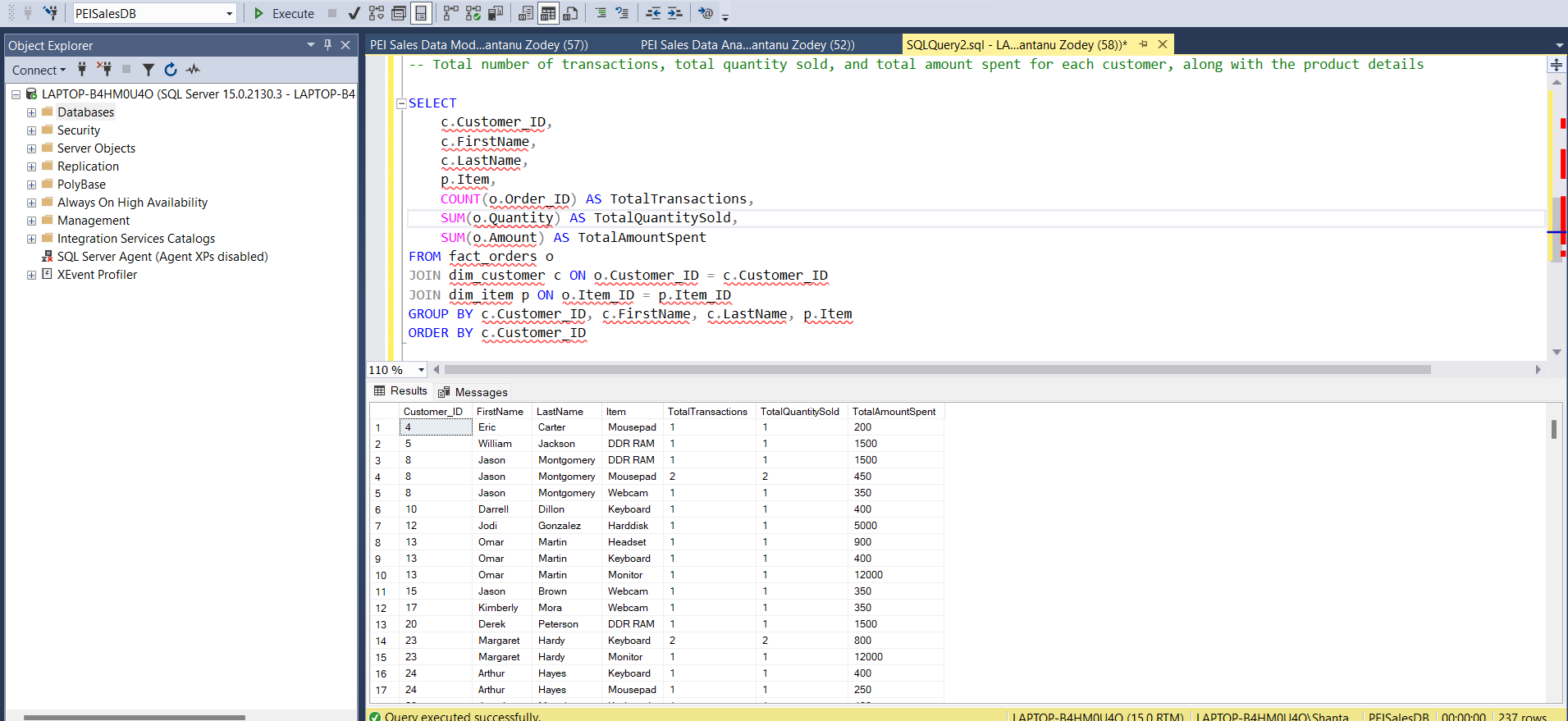
| **Test Case #** | **Description** | **Expected Outcome** |
| --- | --- | --- |
| TC1 | Validate Order\_ID uniqueness | No duplicates in fact\_orders.Order\_ID |
| TC2 | Verify Item\_ID and Customer\_ID exist in dimension | All values must match dimension records |
| TC3 | Validate Amount is not null and > 0 | All records must pass |
| TC4 | Ensure Quantity is always 1 | No deviation |
| TC5 | Join test with dim\_Item and dim\_customer | Ensure joins produce expected results |

# Enable Reporting Requirements

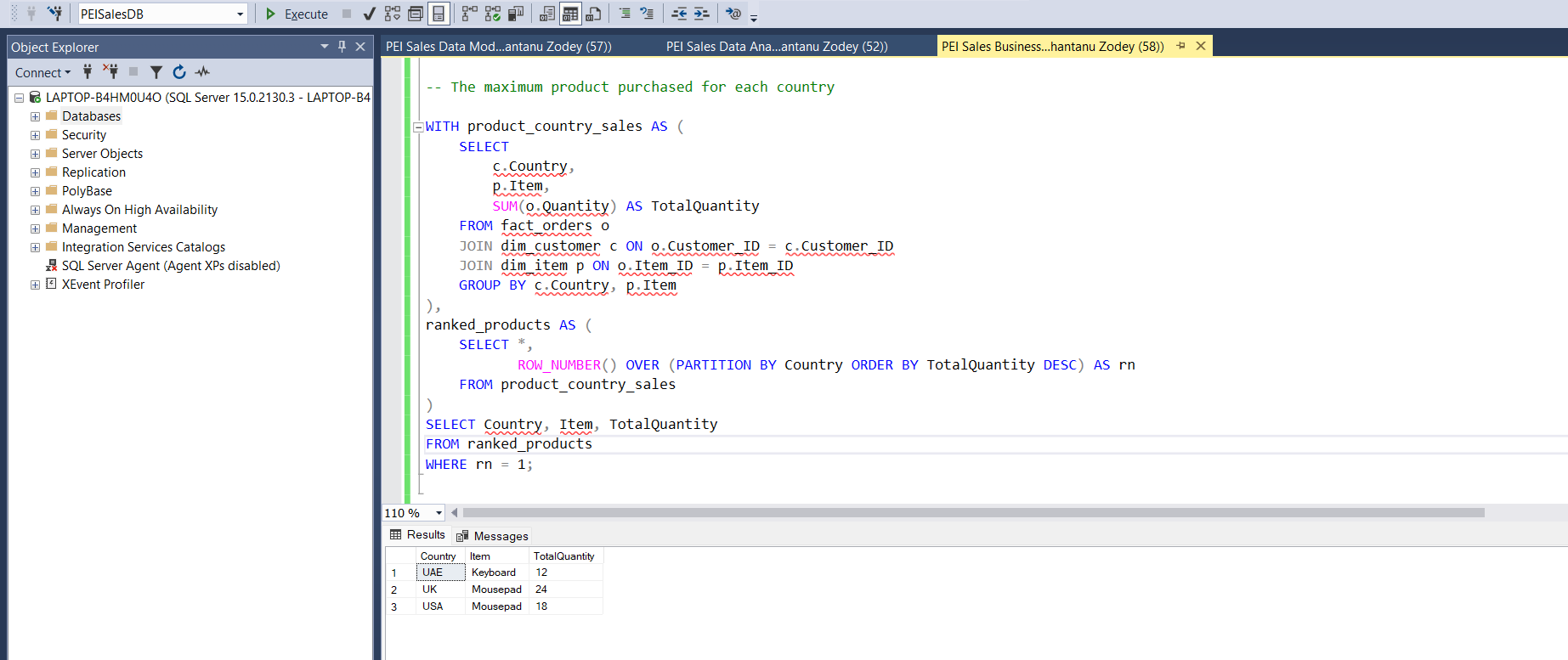
1. Total amount spent and the country for the "Pending" delivery status for each country

****

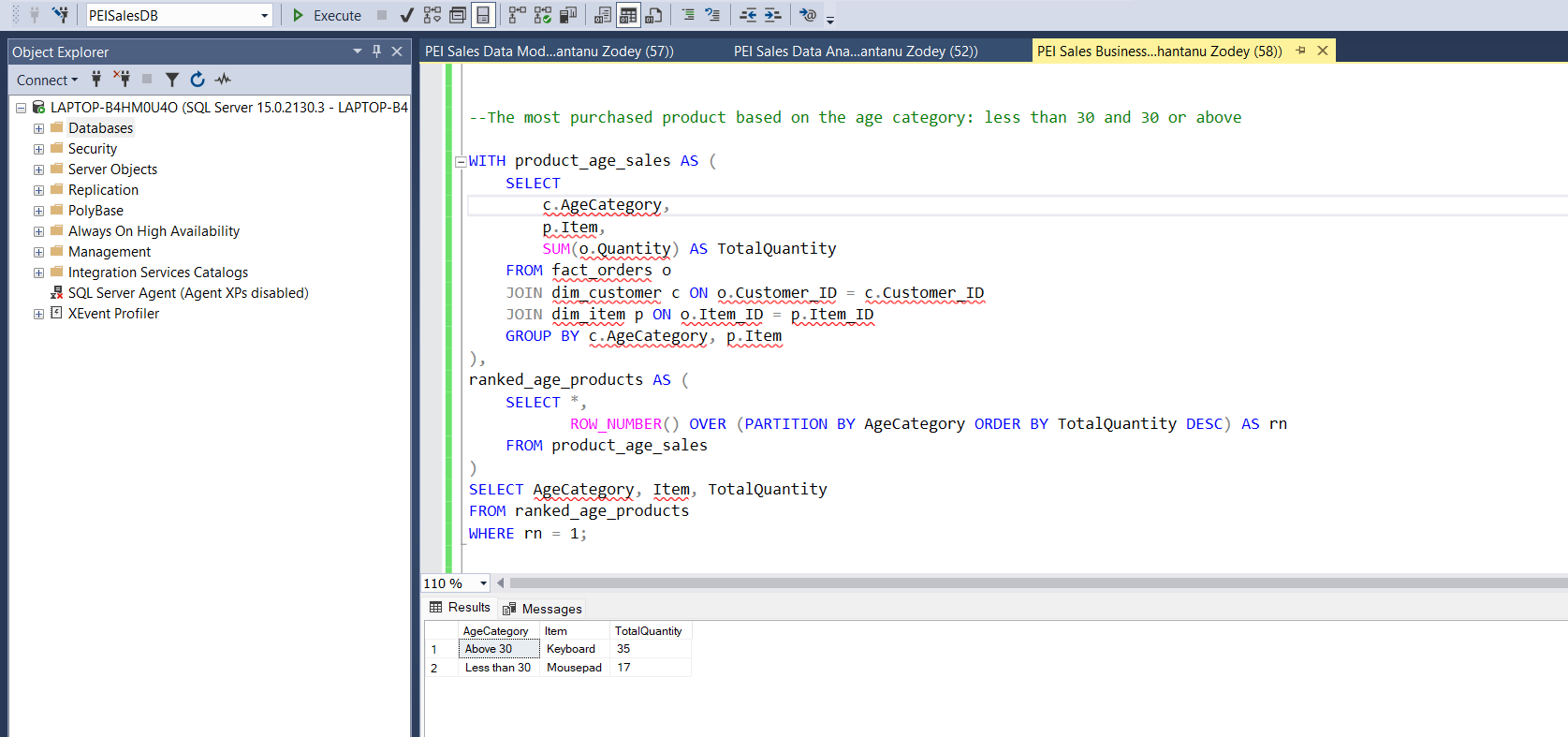
1. Total number of transactions, total quantity sold, and total amount spent for each customer, along with the product details



1. The maximum product purchased for each country



1. The most purchased product based on the age category: less than 30 and 30 or above



1. The country that had the minimum transactions and sales amount

