Sri Lanka Institute of Information Technology

Data Warehousing and Business Intelligence - IT3021

B.Sc. (Hons) in Information Technology

Data Science Specialization

2025

DWBI Assignment 01

Name: S.M.S.G.S.D.SAMPATH

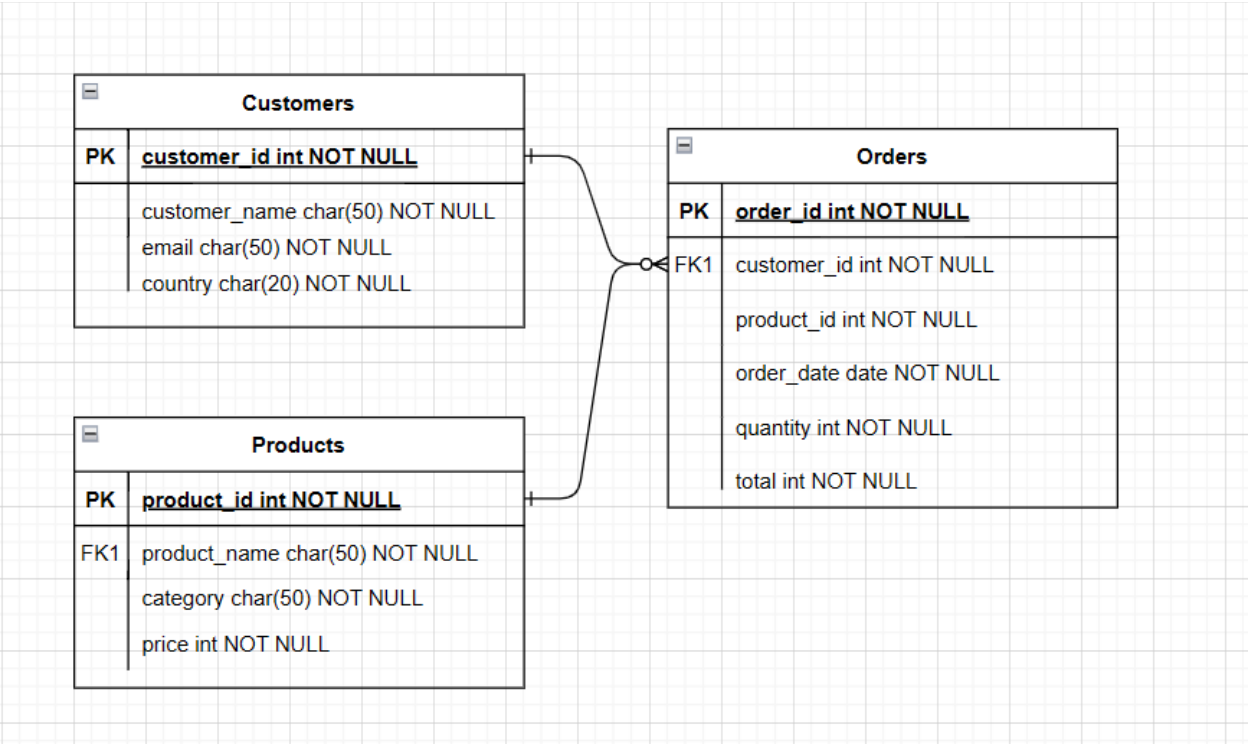
IT Number: IT22310750

Step 1: Data Set Selection

OLTP data set chosen : Online Electronics Store

Table	Description
customers	Customer info (name, country, email)
products	Product info (category, price, stock)
orders	Orders placed by customers

I have chosen a data set that represents an electronics shop that sells items like mobile phones, mobile and PC accessories, and other gadgets online. It consists of customer information, product details, and order transactions. Customers place orders through an online portal (website), and these orders are delivered within a few days. This data set is not OLAP as it is transaction-oriented (customers, products, orders) and is not designed for analytical querying. It is suitable because it can simulate a real-world sales scenario. It also allows to demonstrate multiple data sources (CSV and SQL). It consists of enough data to design a data warehouse with dimensions and facts.



ER Diagram for chosen data set

Step 2: Preparation of Data sources

Data sources

Two CSV files are used to represent master data (customers and products) and a database table for transactional data (orders).

customers.csv – CSV file containing customer information.

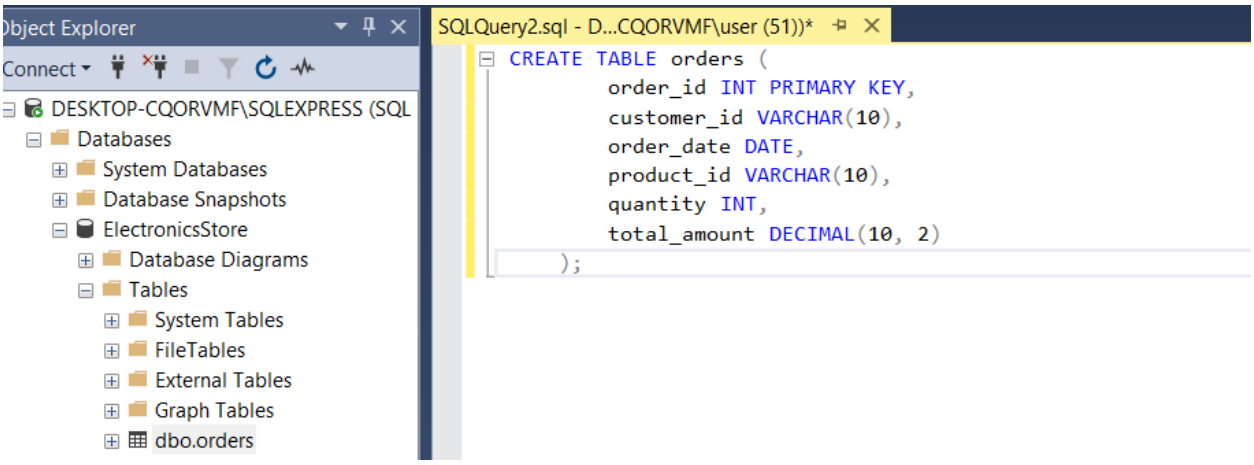
```
customer_id,name,email,country
C001,Amal Bandara,amal@email.com,Sri Lanka
C002,Nirosha Kumari,nirosha@email.com,Sri Lanka
C003,Shalini Perera,shalini@email.com,Sri Lanka
C004,Chamath Rajanayaka,chamath@email.com,Sri Lanka
C005,Kamesh Fernando,kamesh@email.com,Sri Lanka
C006,Lionel Rajasinghe,lionel@email.com,Sri Lanka
C007,Kumudu Wijerathne,kumudu@email.com,Sri Lanka
C008,Sahansa Attanayaka,sahansa@email.com,Sri Lanka
C009,Janashi Hope,janashi@email.com,Sri Lanka
```

products.csv – CSV file containing product information

```
product_id,name,category,price
P001,Fantech Mouse,PC Accessories,4250
P002,MSI Headset,PC Accessories,15000
P003,Table Lamp,Mobile Accessories,1500
P004,Mouse Pad,PC Accessories,999
P005,Phone Holder,Mobile Accessories,3000
P006,Mobile Screen Protector,Mobile Accessories,850
P007,"ASUS 24"" Monitor",PC Accessories,30000
P008,"Dahua 24"" Monitor",PC Accessories,34000
P009,Logitech Headset,PC Accessories,12500
```

orders table – SQL server table that stores order data

The following sql script was used to create the orders table



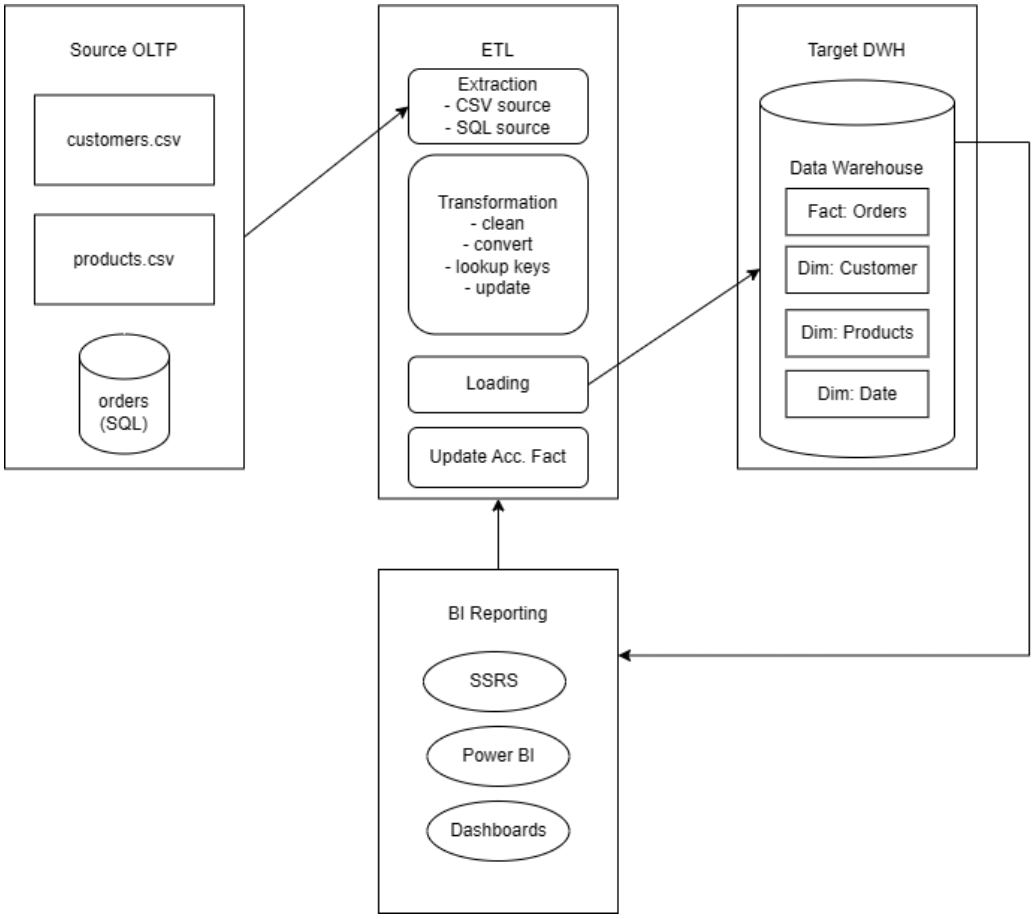
And the order data was inserted and stored there.

100 %

Results | Messages

	order_id	customer_id	order_date	product_id	quantity	total_amount
1	1	C013	2024-05-31	P001	3	12750.00
2	2	C040	2024-03-08	P003	2	3000.00
3	3	C021	2024-03-05	P023	3	15600.00
4	4	C001	2024-10-22	P004	1	999.00
5	5	C025	2024-09-28	P011	3	19500.00
6	6	C023	2024-11-01	P013	3	28500.00
7	7	C056	2024-07-29	P021	3	9600.00
8	8	C061	2024-09-24	P009	2	25000.00
9	9	C033	2024-01-28	P014	3	1800.00
10	10	C072	2024-11-10	P022	2	36000.00
11	11	C026	2024-02-05	P023	3	15600.00
12	12	C060	2024-07-04	P009	1	12500.00
13	13	C042	2024-05-13	P011	1	6500.00
14	14	C023	2024-12-28	P006	2	1700.00
15	15	C073	2024-06-15	P019	5	55000.00
16	16	C017	2024-12-16	P021	1	3200.00
17	17	C028	2024-01-23	P017	3	66000.00
18	18	C009	2024-05-24	P014	2	1200.00

Step 3: Solution Architecture



Source OLTP (Online Transaction Processing)

Represents the operational data sources that provide the raw data for the data warehouse, consisting of the systems and files used for day-to-day business operations (customers.csv, products.csv and orders (SQL)) This is what provides the initial data that will be extracted, transformed, and loaded into the data warehouse for analysis and reporting.

ETL (Extract, Transform, Load)

The core process responsible for moving data from the Source OLTP to the Target DWH. It involves a series of steps to including retrieval of data from the various source systems, cleaning, transforming, and integrating the extracted data to make it suitable for the data warehouse.

Target DWH (Data Warehouse)

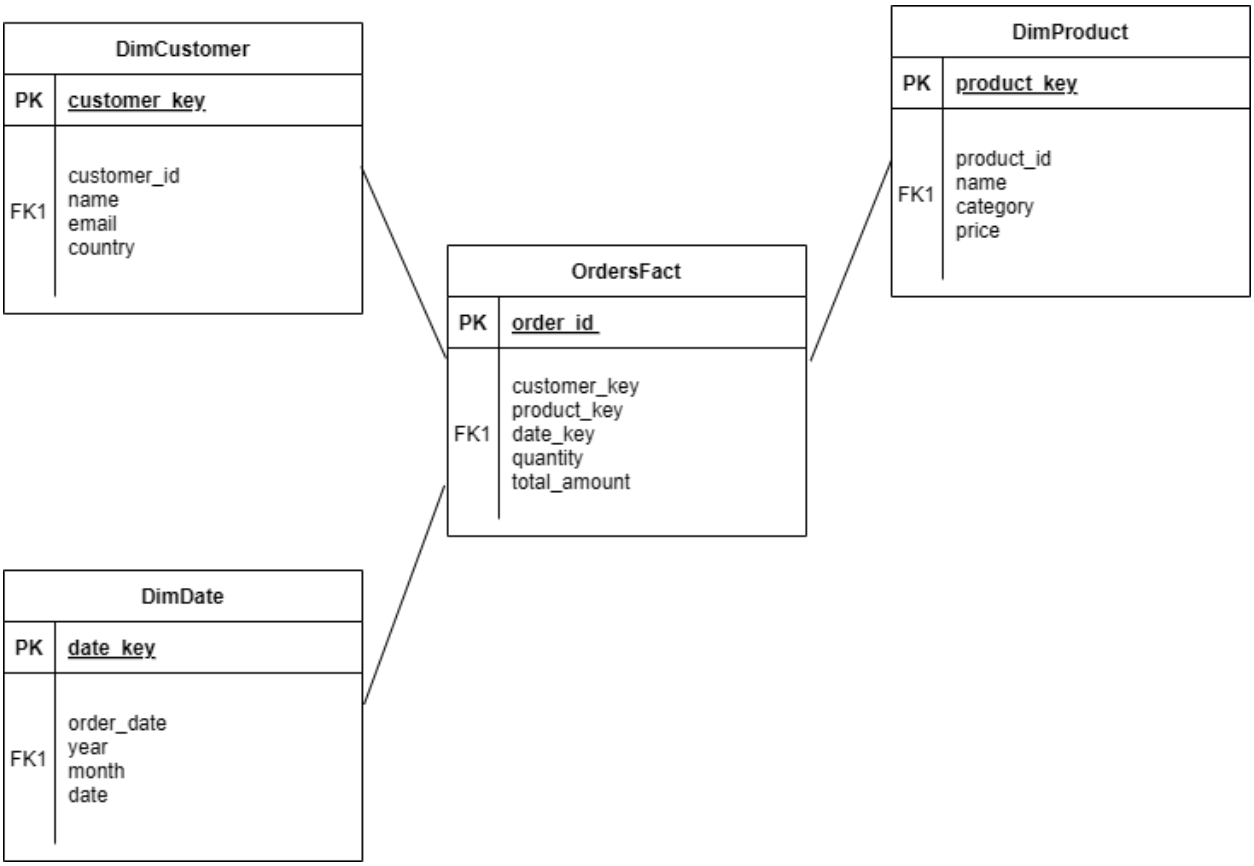
It is the central repository for integrated data, designed to support analytical queries and reporting. It comprises of a data warehouse which is the overall database that stores the analytical data, Fact: Orders: the fact table that stores transactional data about orders, Dim: Customer: a dimension table containing customer information, Dim: Product: a dimension table containing product details and Dim: Date: a dimension table providing time-related attributes for analyzing trends.

BI Reporting (Business Intelligence Reporting)

This is the component that provides tools and technologies accessing and analyzing the data stored inside the data warehouse. It uses services such as SSRS (SQL Server Reporting Services), Power BI (A business analytics service for interactive visualizations and dashboards) and dashboards (visually displays key performance indicators and other relevant metrics).

Step 4: Data Warehouse Design & Development

Dimensional Model (Star Schema)



Dimensions:

- DimCustomer: Stores customer details.
- DimProduct: Stores product information.
- DimDate: Stores date and time attributes for analyzing trends over time.

Fact Table:

- OrdersFact: Stores order transaction facts, linked to dimensions using foreign keys.

Slowly Changing Dimension (SCD):

- I will be implementing Type 1 SCD for the DimProduct table. If a product's price changes, the existing record will be updated.

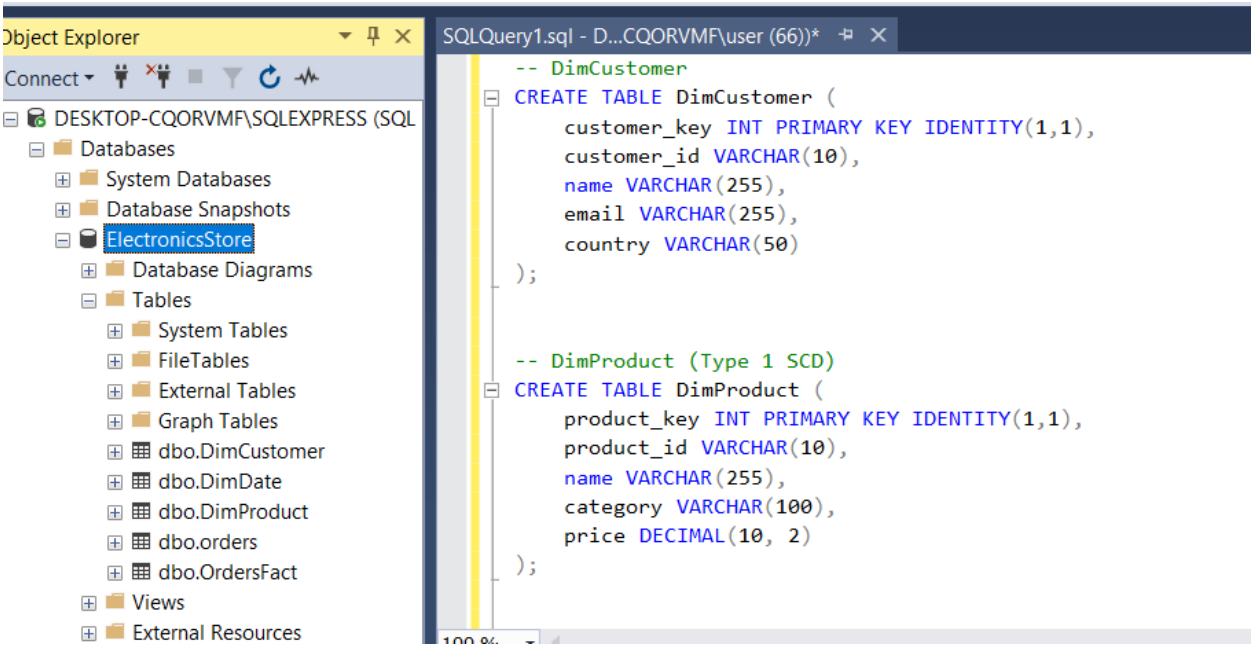
The following SQL script was used to implement the data warehouse schema in SQL

```
-- DimCustomer
CREATE TABLE DimCustomer (
      customer_key INT PRIMARY KEY IDENTITY(1,1),
      customer_id VARCHAR(10),
      name VARCHAR(255),
      email VARCHAR(255),
      country VARCHAR(50)
);

-- DimProduct (Type 1 SCD)
CREATE TABLE DimProduct (
      product_key INT PRIMARY KEY IDENTITY(1,1),
      product_id VARCHAR(10),
      name VARCHAR(255),
      category VARCHAR(100),
      price DECIMAL(10, 2)
);

-- DimDate
CREATE TABLE DimDate (
      date_key INT PRIMARY KEY,
      order_date DATE,
      year INT,
      month INT,
      day INT
);

-- OrdersFact
CREATE TABLE OrdersFact (
      order_id INT PRIMARY KEY,
      customer_key INT FOREIGN KEY REFERENCES
      DimCustomer(customer_key),
      product_key INT FOREIGN KEY REFERENCES DimProduct(product_key),
      date_key INT FOREIGN KEY REFERENCES DimDate(date_key),
      quantity INT, total_amount DECIMAL(10, 2)
);
```

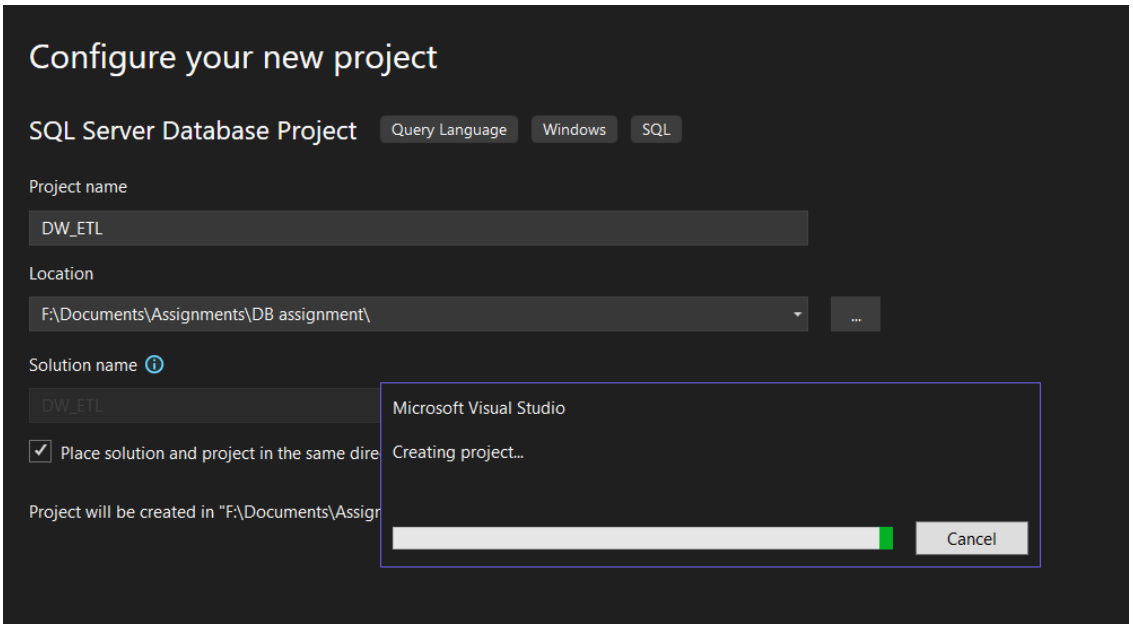


Assumptions:

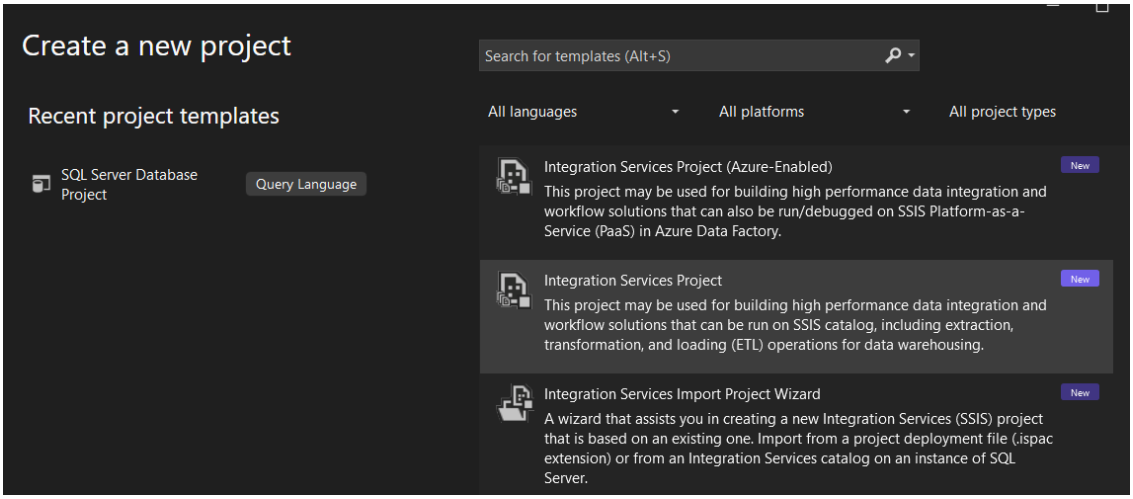
- I have used a simple star schema for ease of demonstration.
- The order_id is assumed to be unique across the source system.
- I've chosen Type 1 SCD for DimProduct for simplicity. In a real-world scenario, Type 2 maybe used to keep historical price changes.

Step 5: ETL Development

I have used Visual Studio to develop the ETL



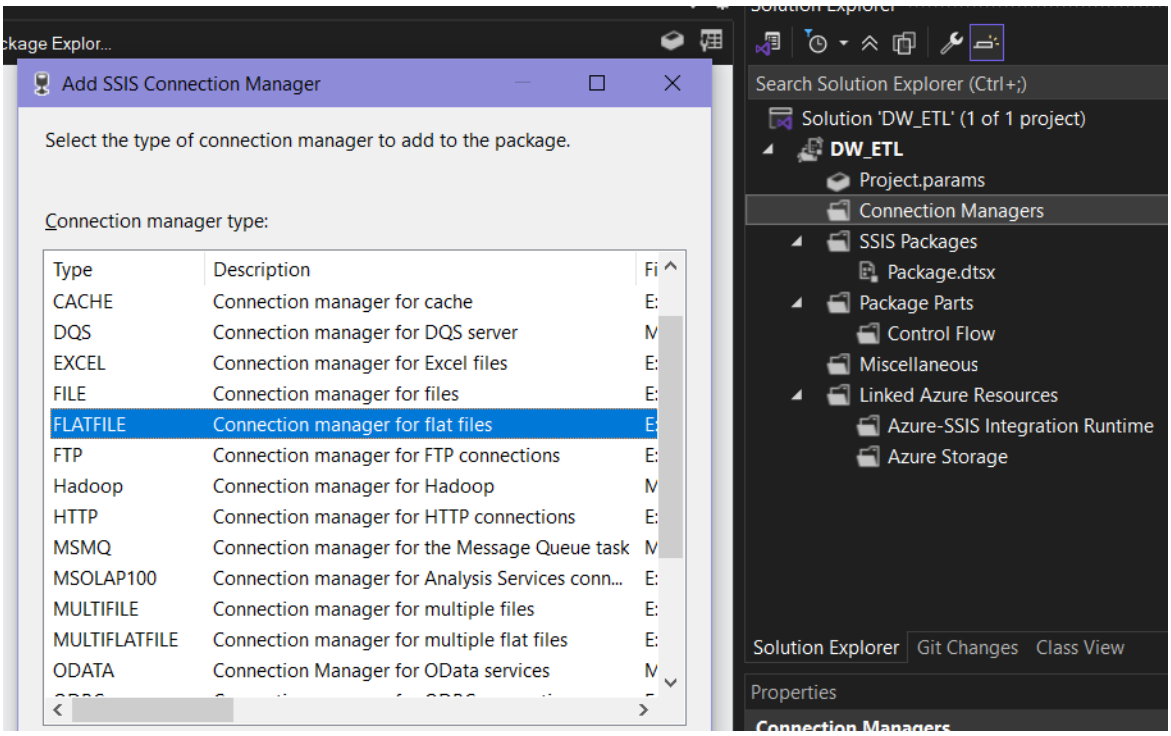
An **Integration Services Project** needs to be created in order to develop ETL.



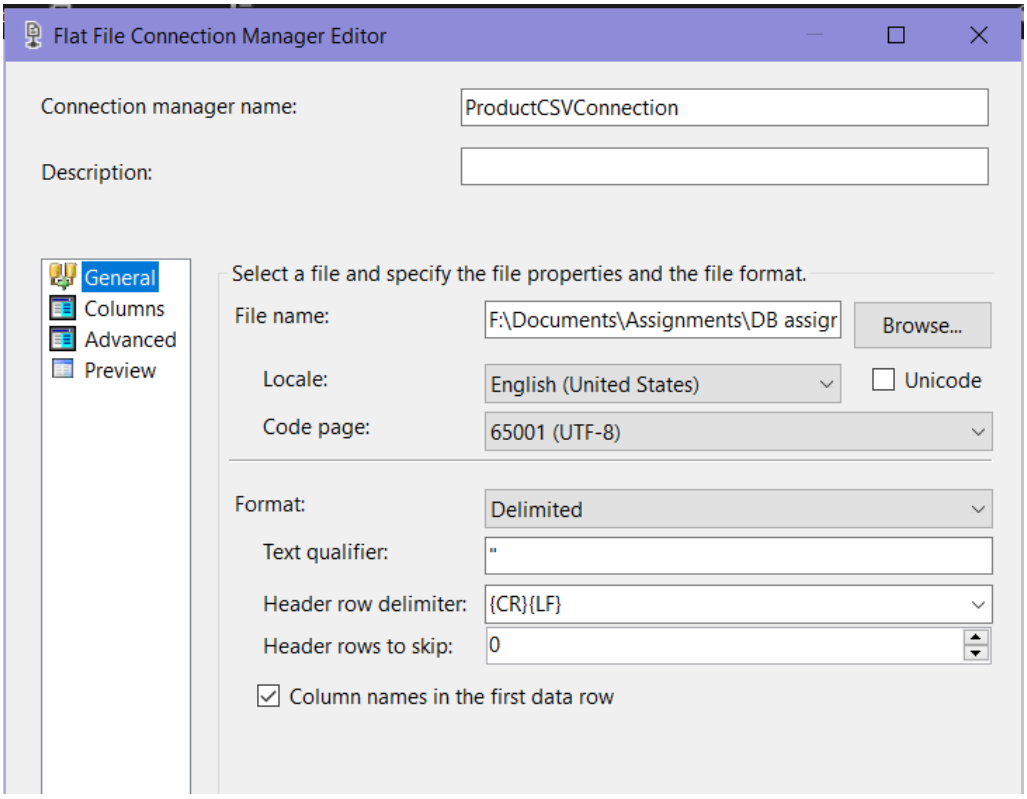
Set Up Connections

Once the project is created, the **ProductCSVconnection** needs to be added by right-clicking on the 'Connection Managers' option in the created project.

The FLATFILE option should be selected.



The preferred name should be given to it along with relevant settings as shown below.

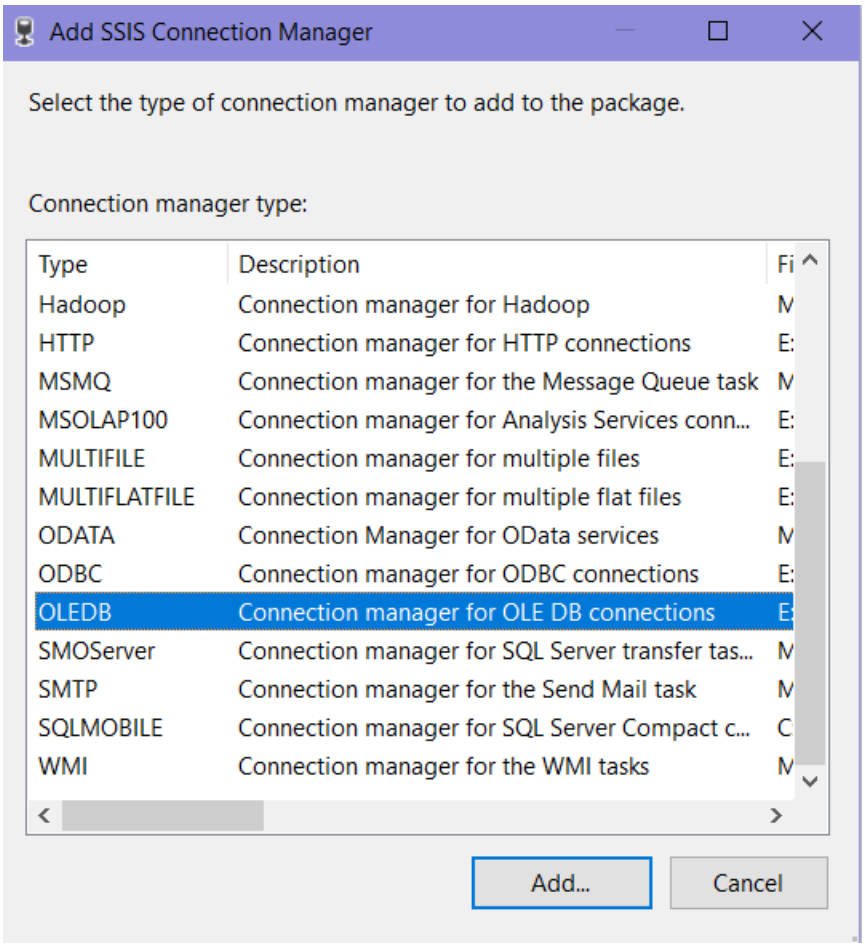


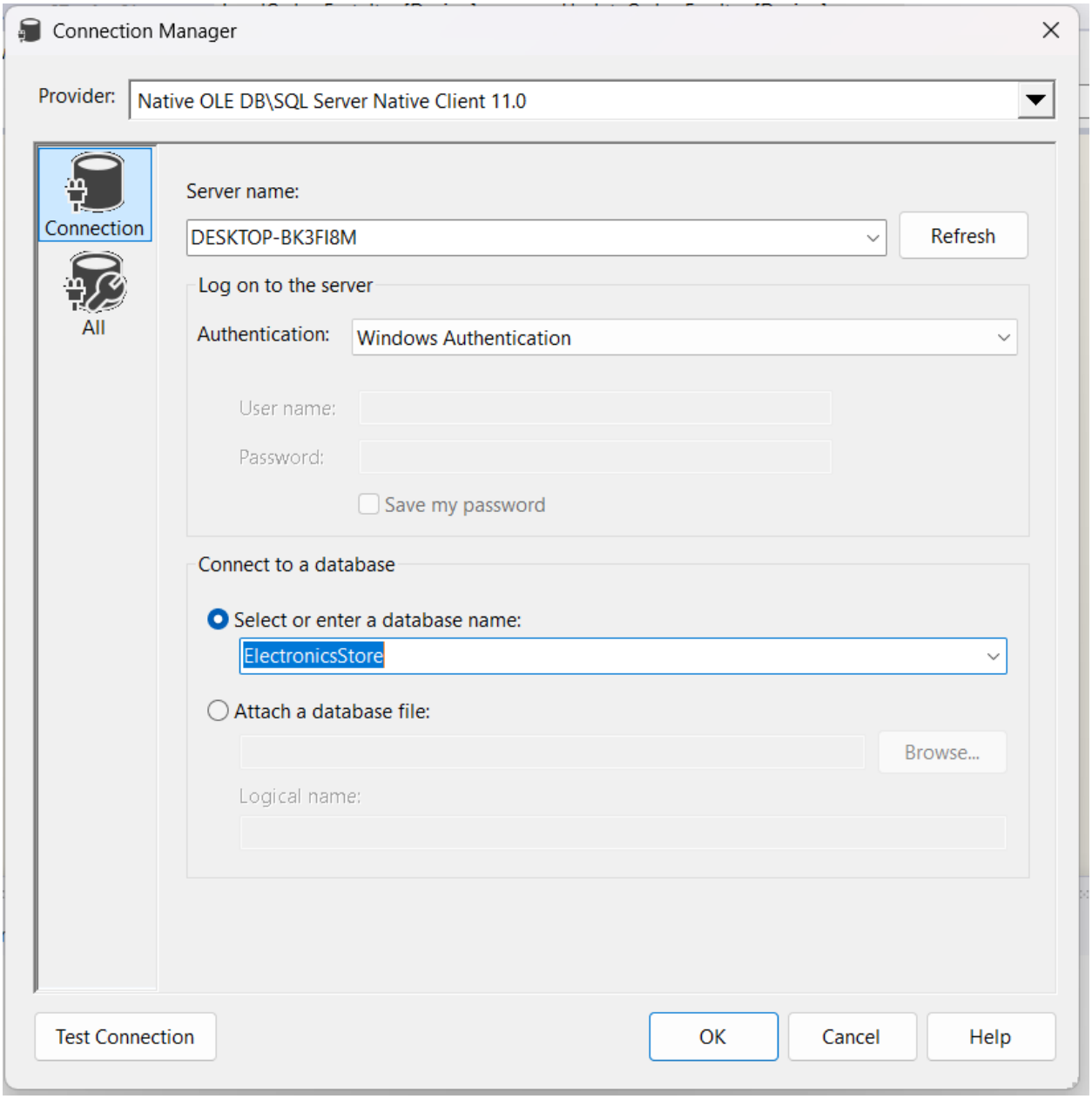
Similarly another Flat File Connection Manager should be created for the customers.csv.

Creating an OLE DB Connection Manager

A new OLE DB Connection Manager was created for the SQL server with the following configs and steps.

- Server name: DESKTOP-BK3FI8M
- Database name: ElectronicsStore
- Authentication: Use Windows Authentication





Once all three connection managers are configured, the SSIS packages need to be created.

Creating SSIS Packages

The following three packages were created.

- LoadDimensions.dtsx: For loading DimCustomer, DimProduct, and DimDate.

- LoadOrdersFact.dtsx: For loading the OrdersFact table.
- UpdateOrdersFact.dtsx: For updating the accumulating fact table columns.

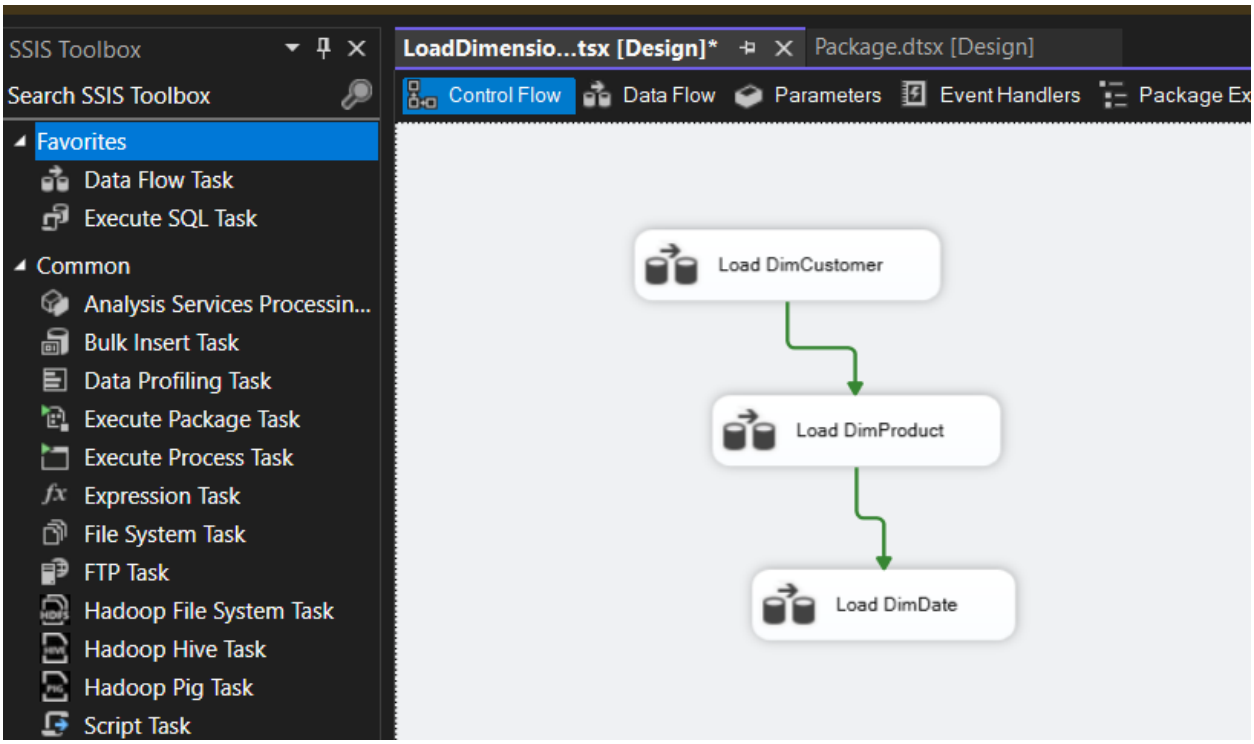
LoadDimensions.dtsx Package

Control Flow:

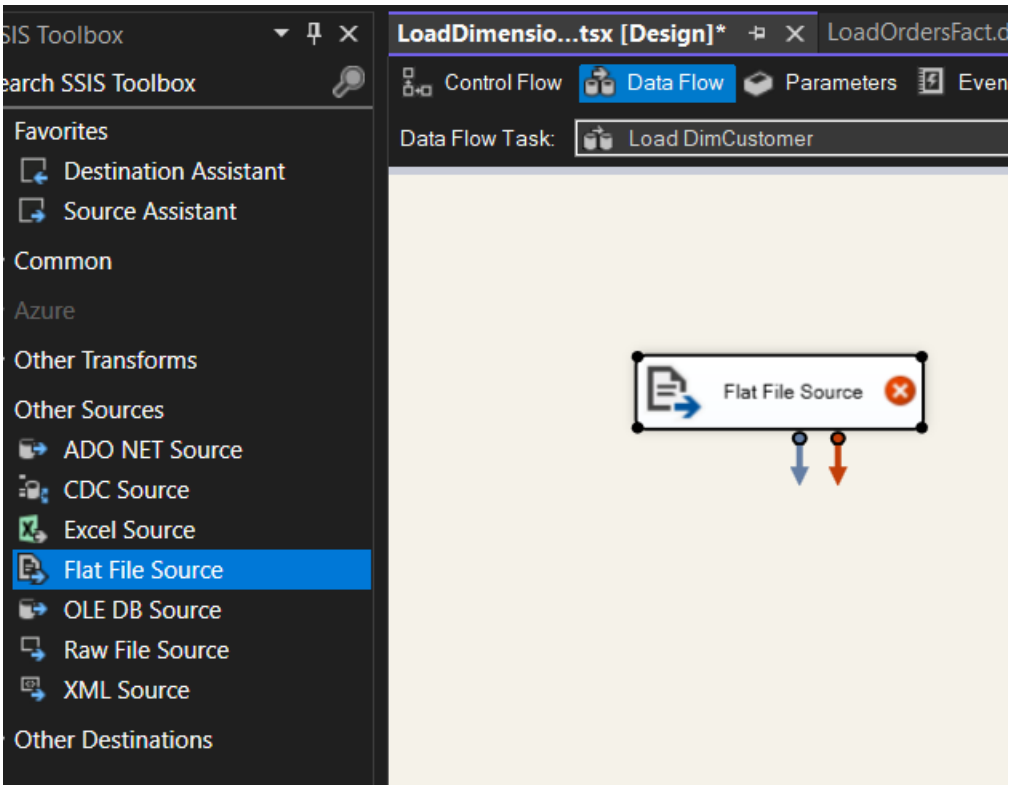
Data Flow Tasks:

- Data Flow Task: Load DimCustomer
- Data Flow Task 1: Load DimProduct
- Data Flow Task 2: Load DimDate

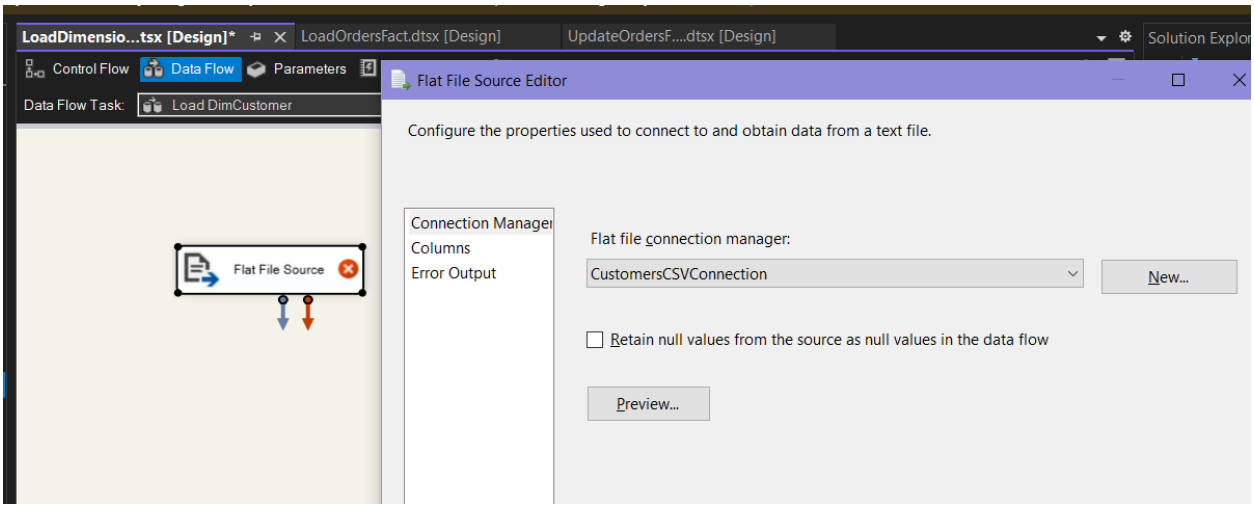
The green arrow was connected from the first task to the second, and then from the second to the third, defining the sequence.



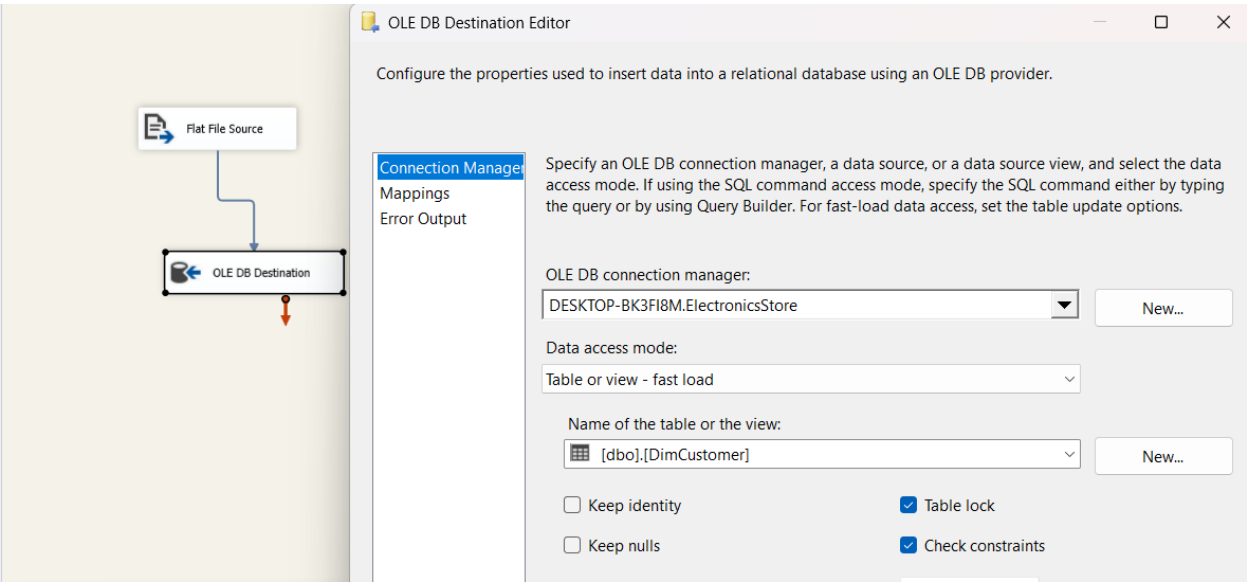
✓ Double click on Load DimCustomer to configure its data flow as below.

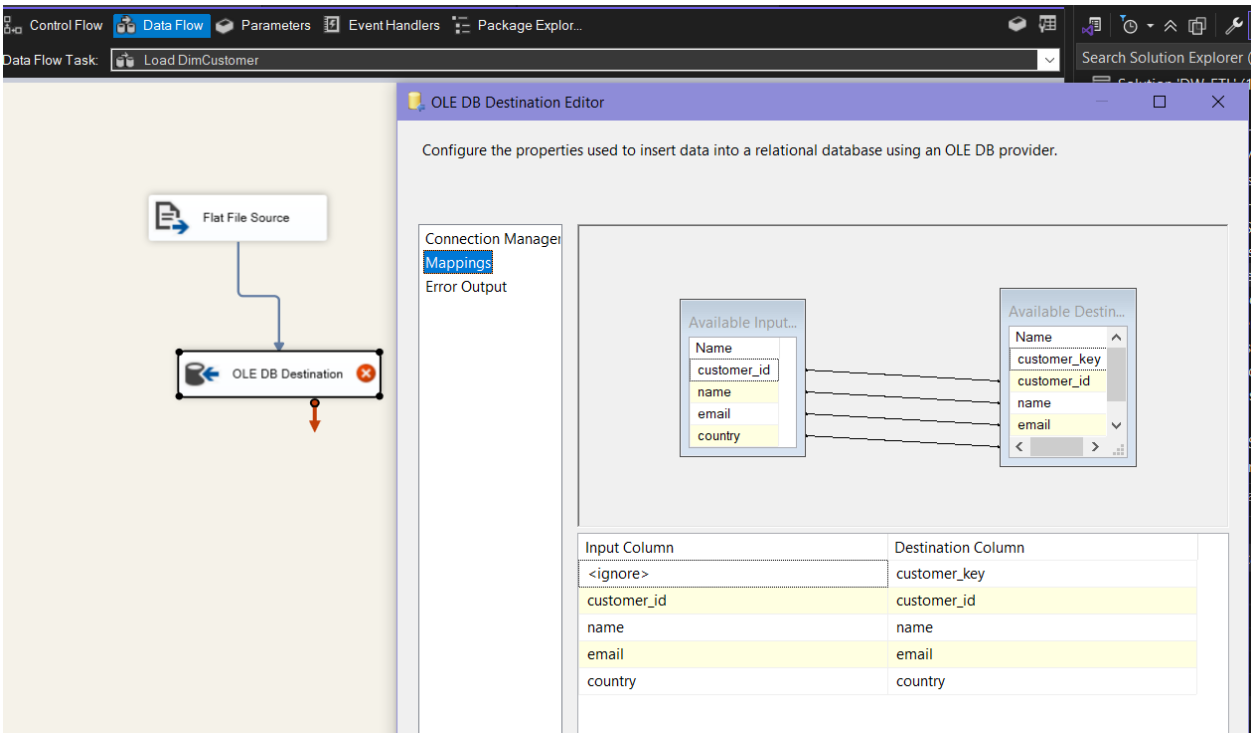


The Flat File Source should be configured to load data from the relevant file source as shown below. The settings pop up can be viewed by right-clicking on the node.

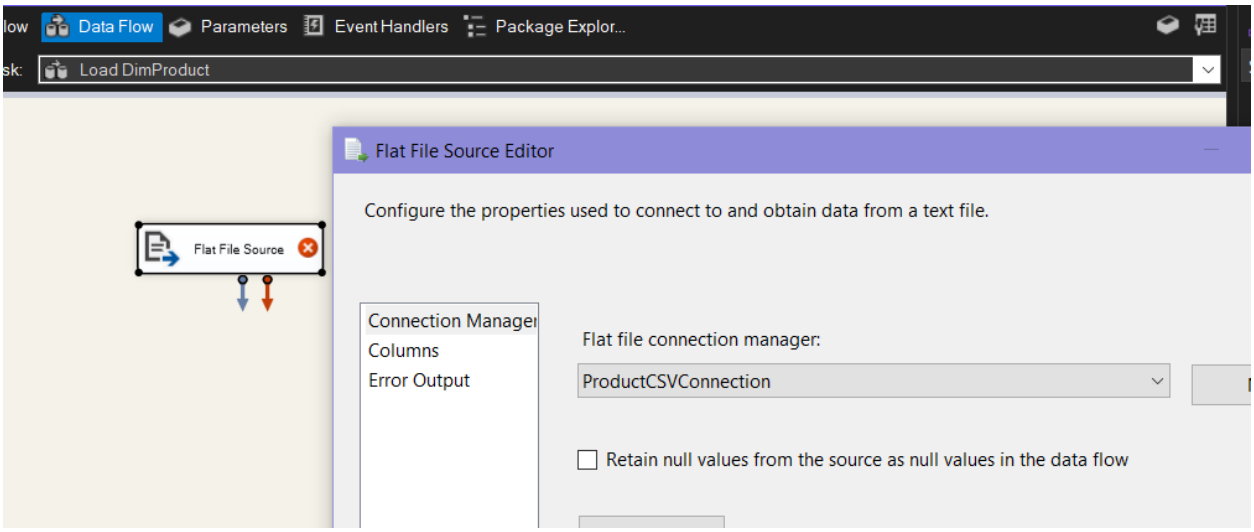


The relevant table should be selected as shown below and the columns mapped as required.

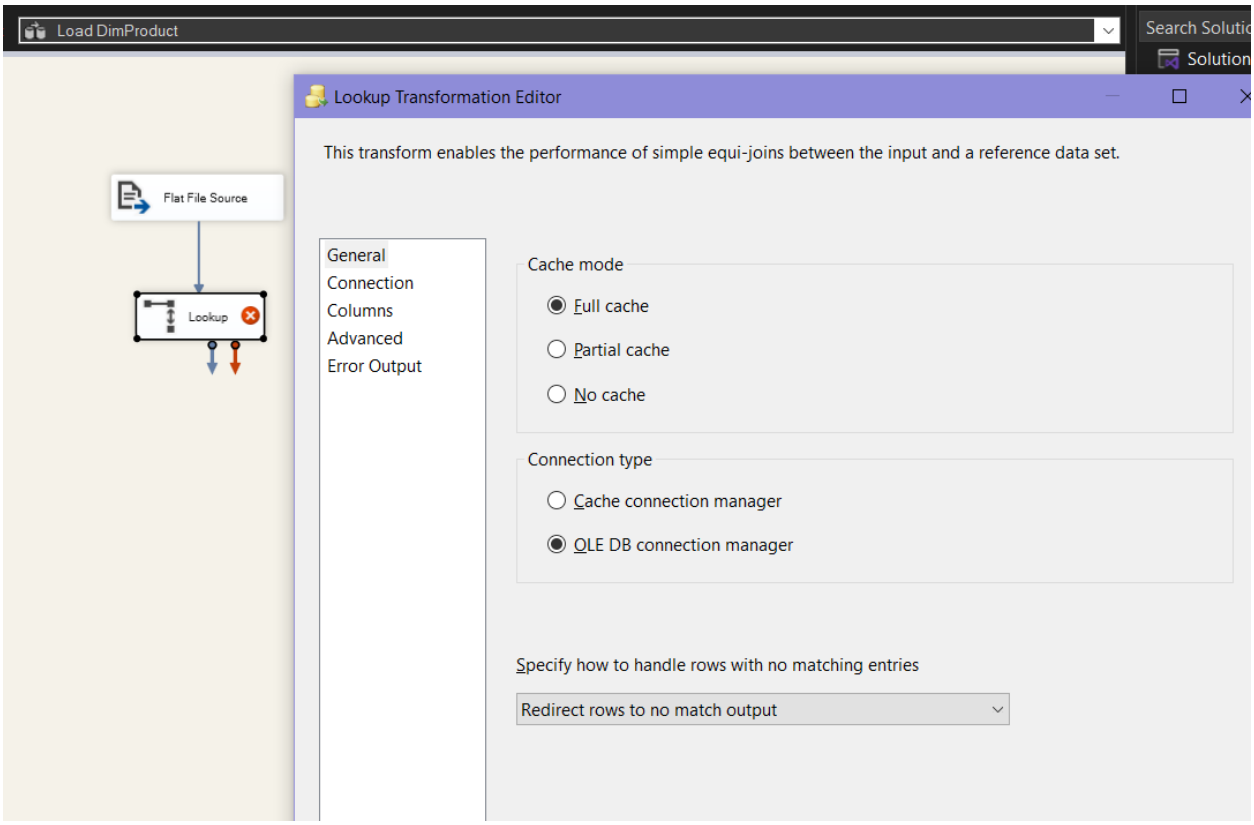


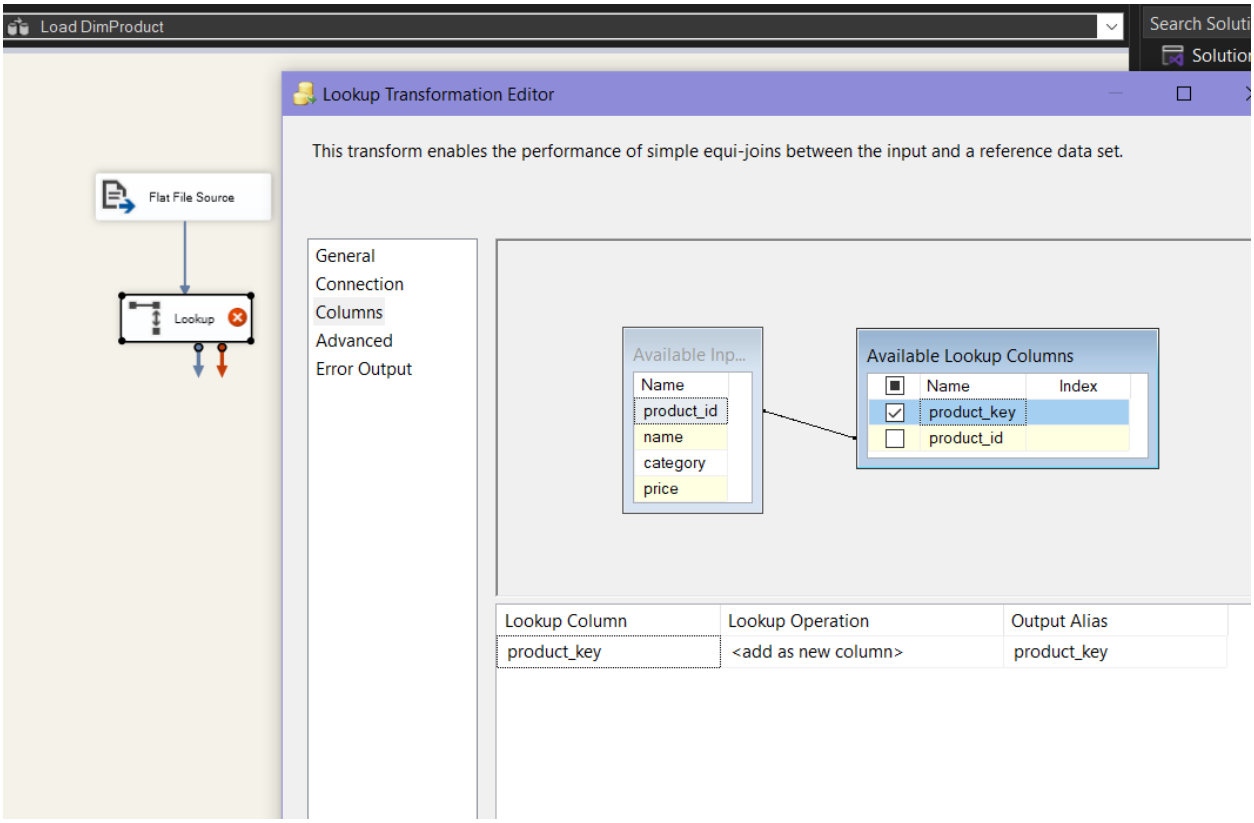


✓ Similarly, the data flow for Load DimProduct should be configured as follows.

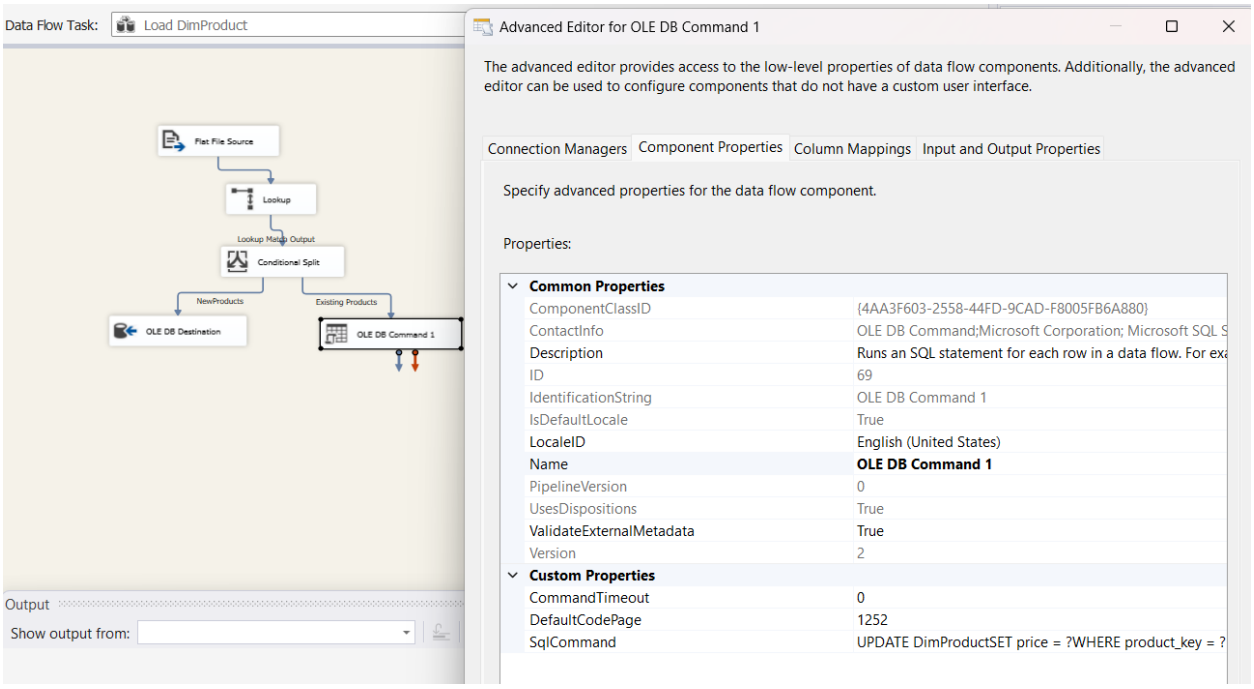


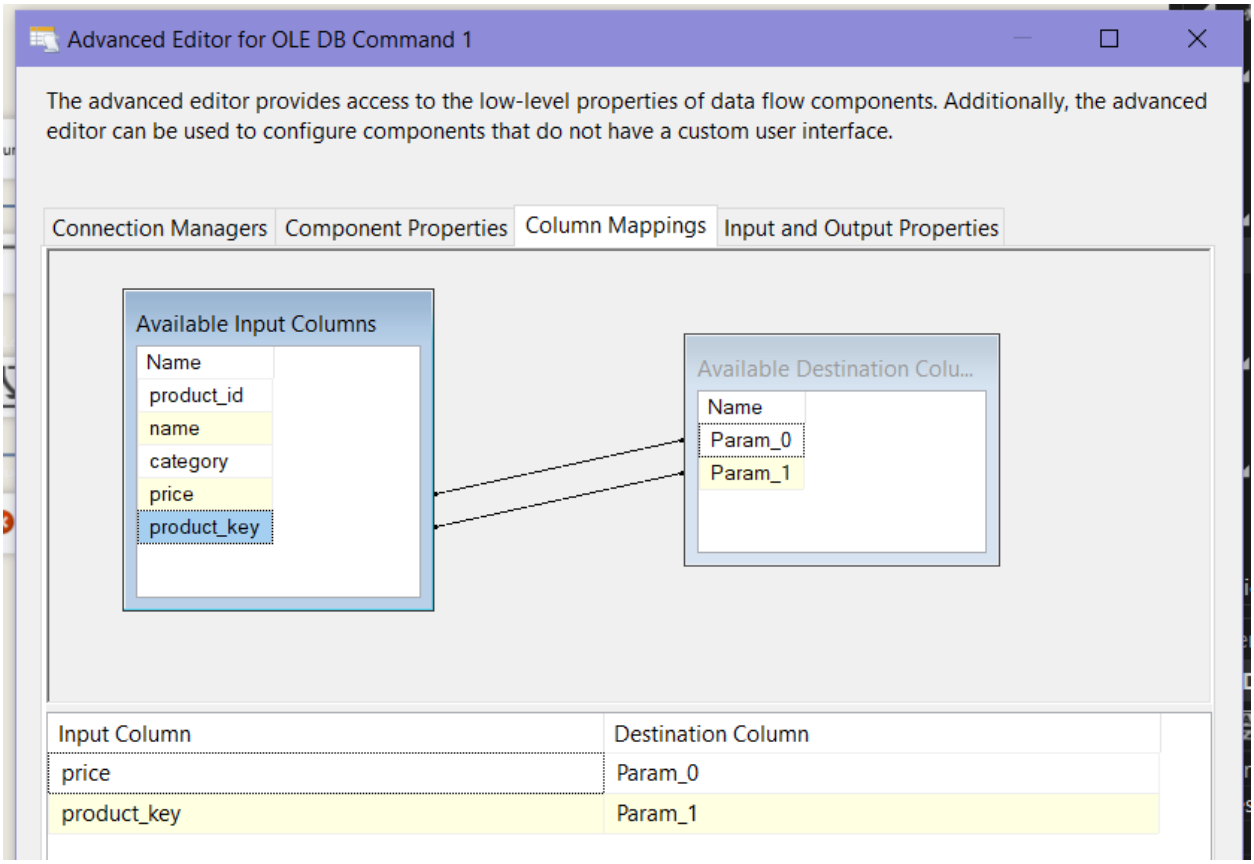
A lookup transformation was added to it as below and relevant columns mapped.



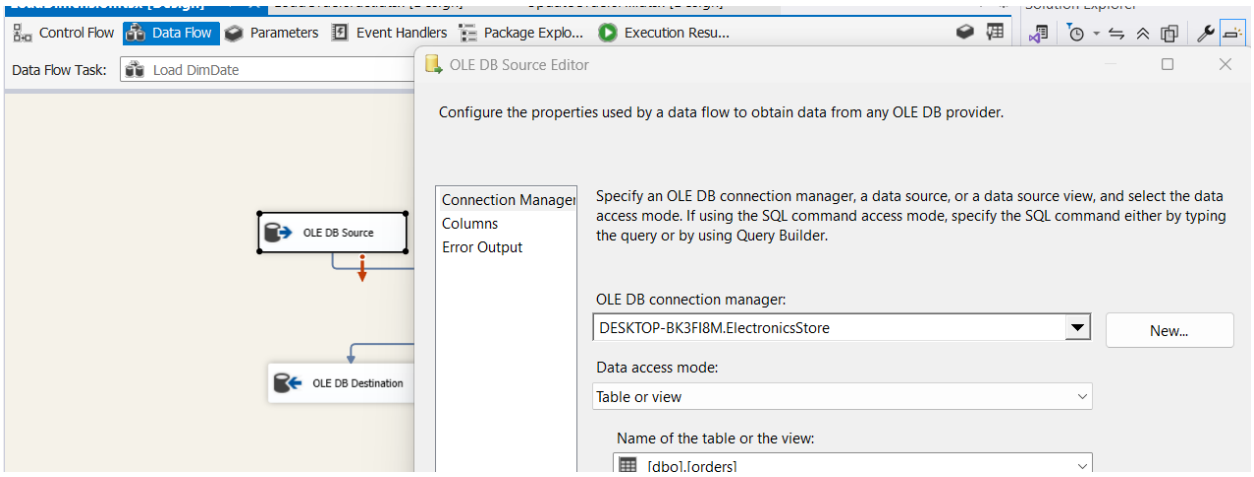


Two OLE DB Connections were added to get two outputs for error handling.

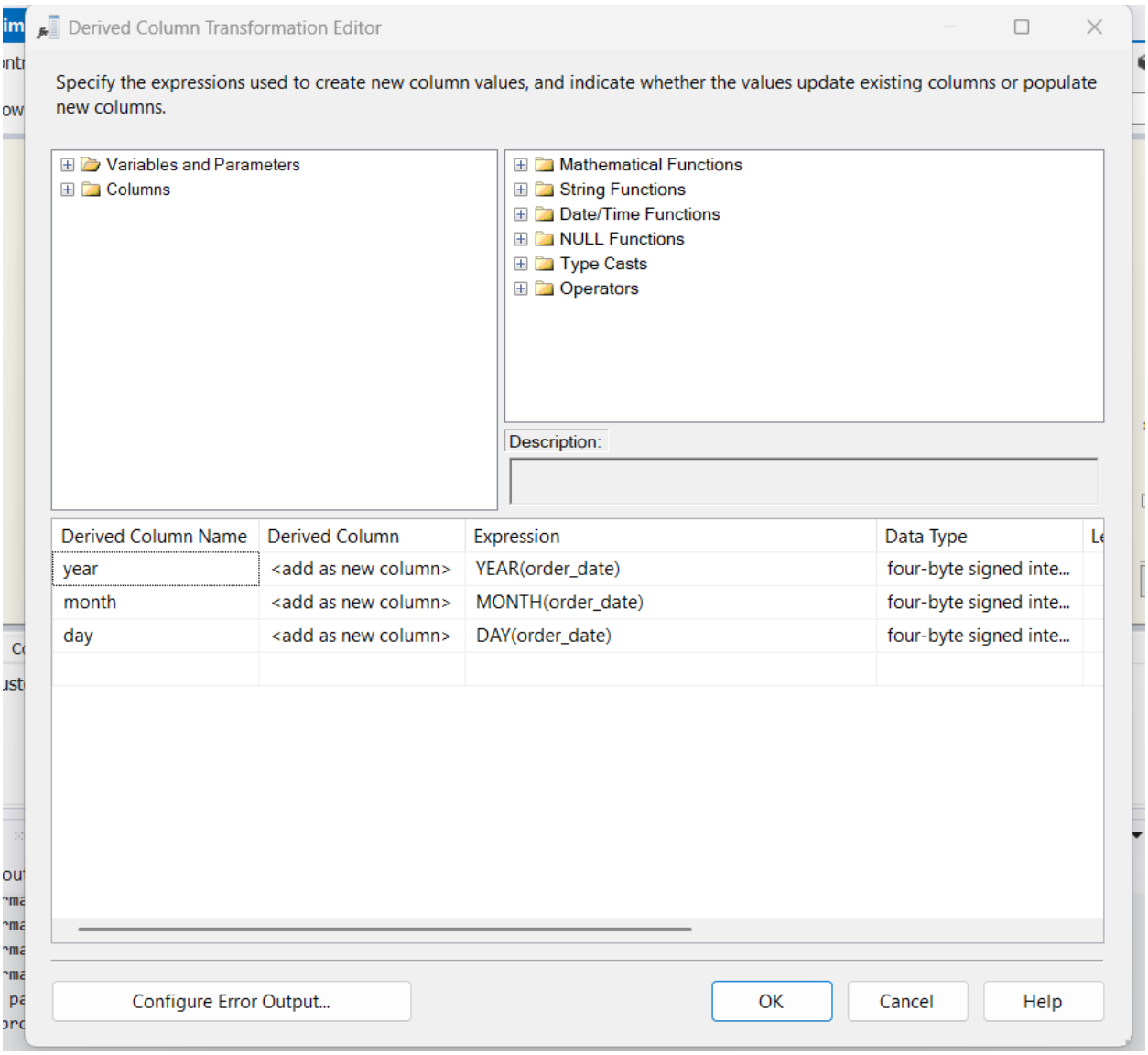




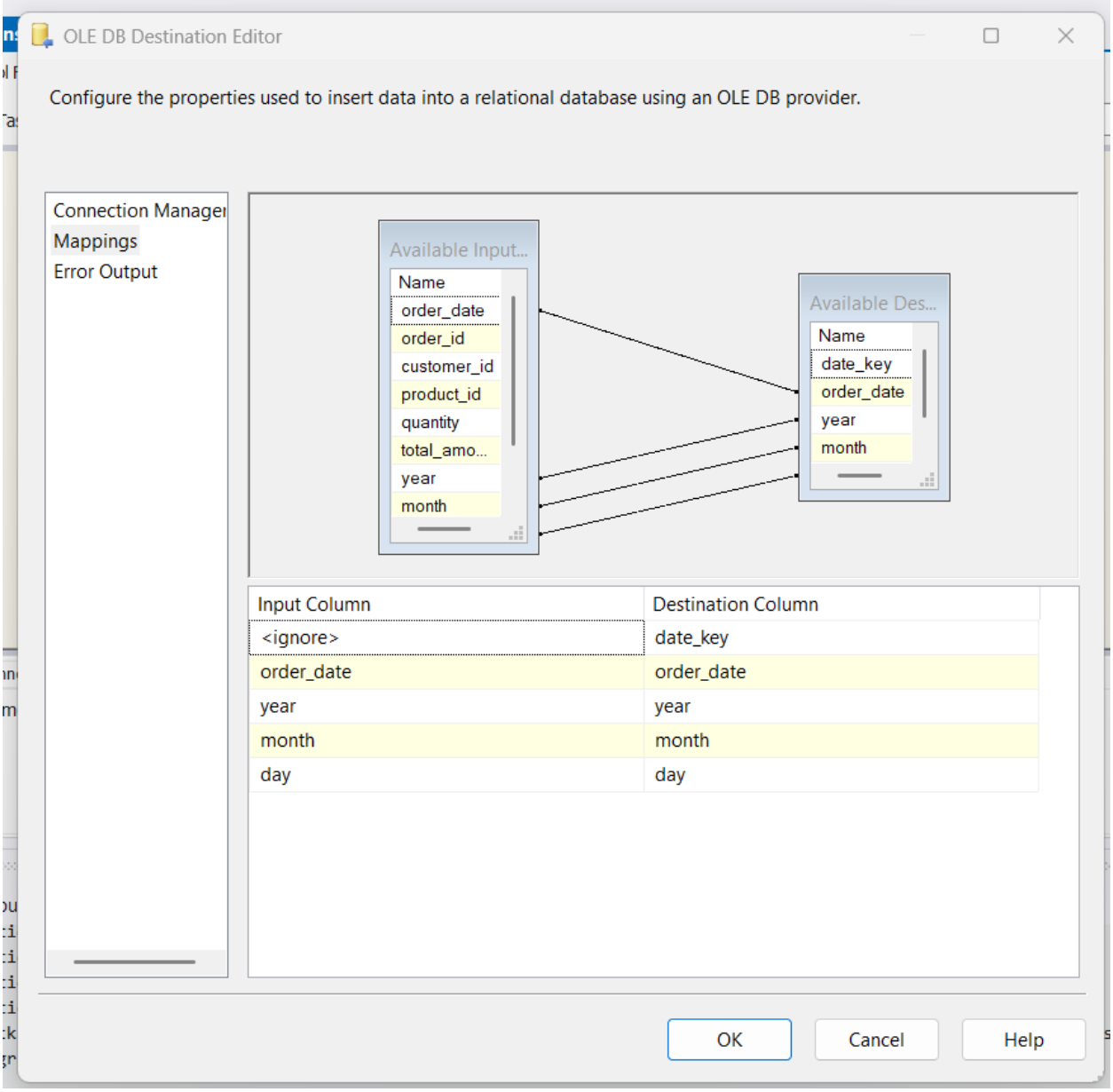
✓ Similarly, the data flow for Load DimDate should be configured as follows.



Derived Column : transformation component that creates new column values or replaces existing columns by applying expressions to input data.



OLE DB Destination: The endpoint that loads the transformed data into an OLE DB-compliant destination database.

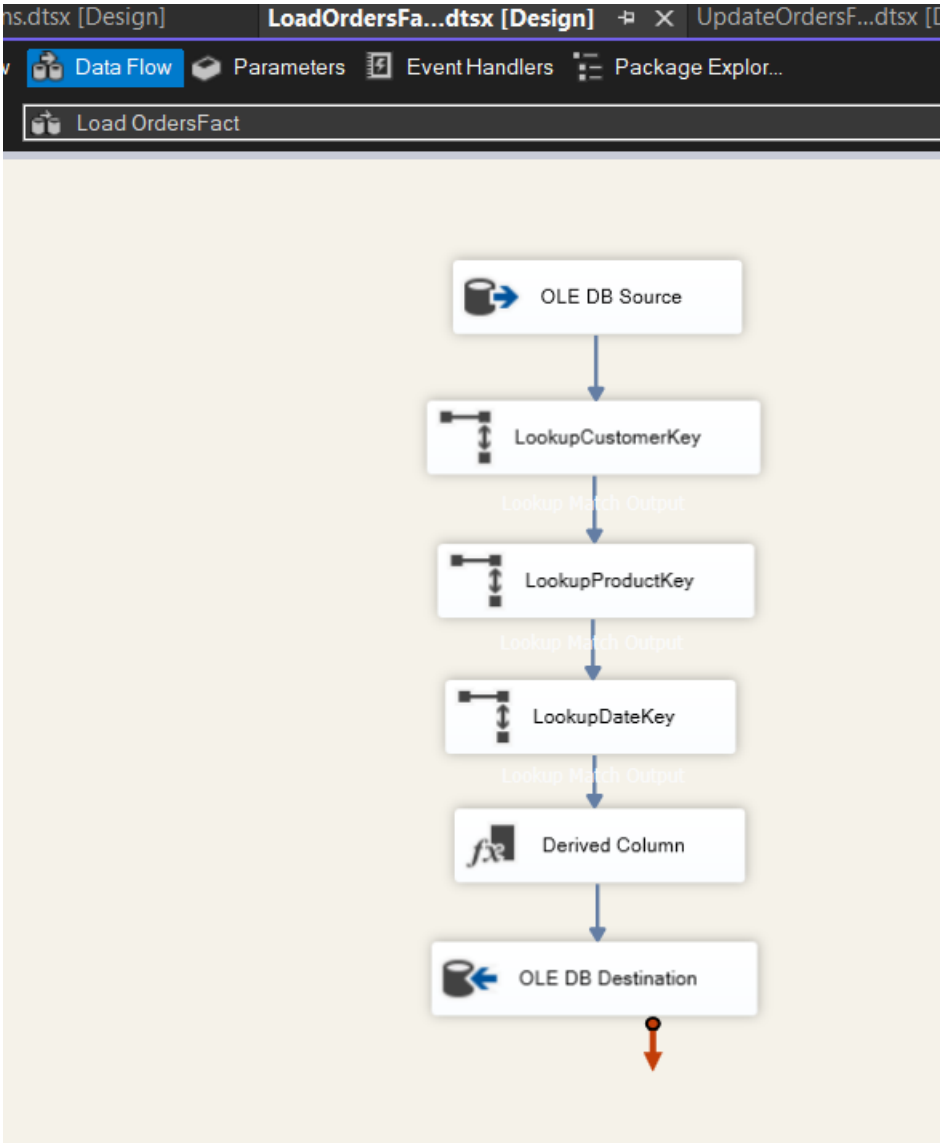


This package is responsible for extracting data from the source and loading it into the dimension tables of the data warehouse (DimCustomer, DimProduct, and DimDate). It implements Type 1 Slowly Changing Dimension (SCD) logic for the DimProduct table to handle product price changes and retrieves date data to update the DimDate table. It also ensures that the dimension tables are populated with clean and transformed data, ready to be used for analysis and to provide context to the fact table.

LoadOrdersFact.dtsx

This package was created to extract order transaction data from the source OLTP database and loads it into the OrdersFact table in the data warehouse. It extracts order details from the orders table and performs lookup operations to retrieve the lookup keys for customers, products, and dates from the dimension tables. It captures the transaction creation timestamp (accm_txn_create_time). It also performs the loading of the order facts, along with the associated dimension keys, into the fact table, establishing the relationships between the transactional data and the dimensions.

The data flow for this package was configured as below.

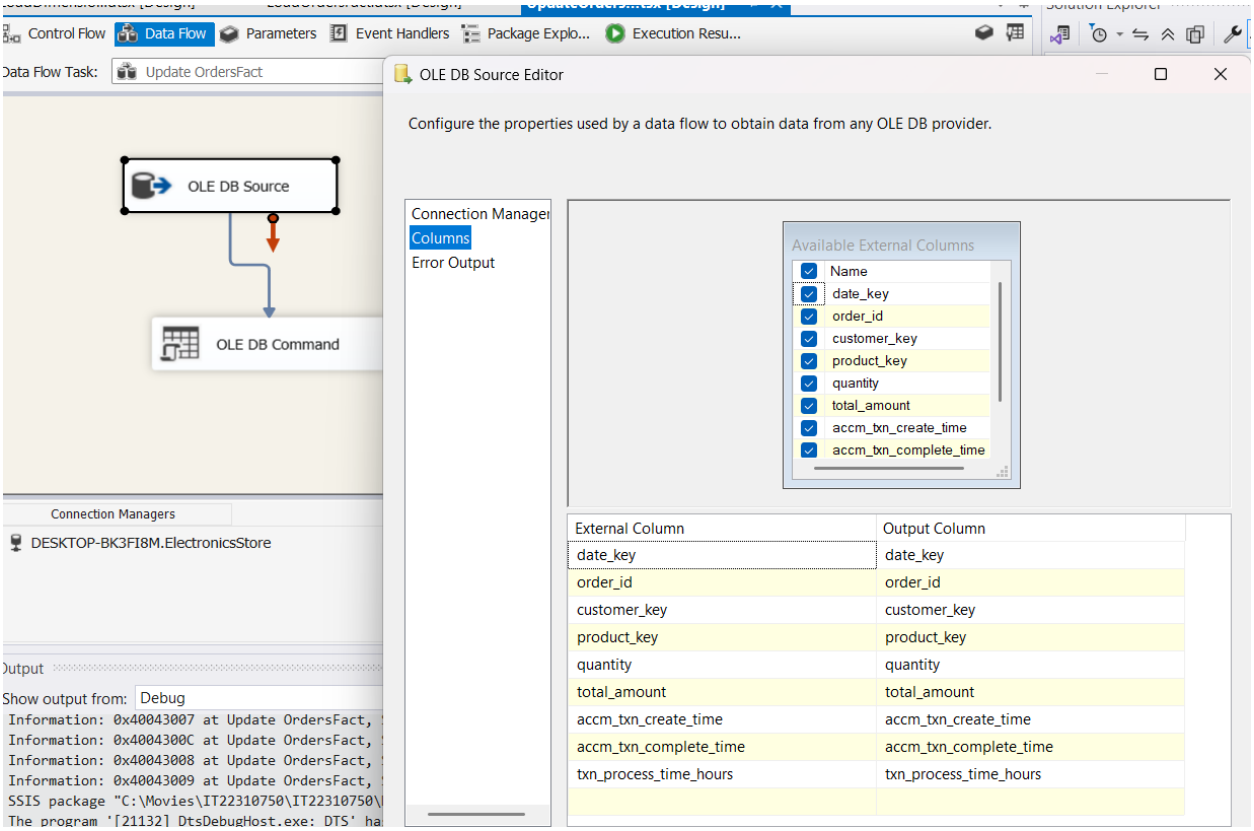


UpdateOrdersFact.dtsx

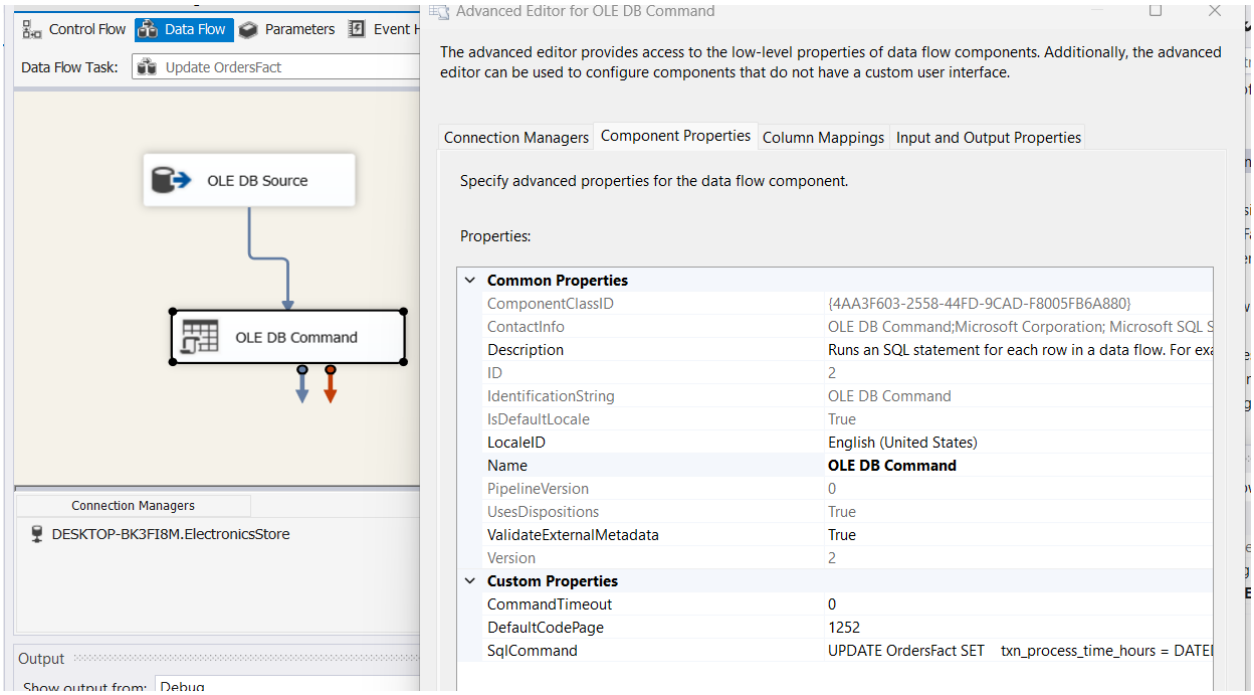
This package was created with the purpose of handling the updating of specific columns in the OrdersFact table related to transaction completion time and processing duration.

It reads data from a separate source (CSV file or SQL table) containing order completion times and updates the accm_txn_complete_time column in the OrdersFact table for the corresponding orders. It also calculates the transaction processing time (txn_process_time_hours) by finding the difference between the creation and completion times. It also addresses the requirement to handle late-arriving data and update the fact table with information that becomes available after the initial data load.

OLE DB Source Configurations:



OLE DB Command Configurations:



Extract

Extract data from customers.csv and load it into the DimCustomer table, products.csv and load it into the DimProduct table and orders SQL Server table and load it into the OrdersFact table while performing lookups to get the customer_key, product_key, and date_key.

Transform

Performing data type conversions where necessary, cleanse data and lookup dimension keys (customer_key, product_key, date_key).

Load

Load the transformed data into the respective Data Warehouse tables (DimCustomer, DimProduct, DimDate, OrdersFact).

SSIS Tasks

- Data Flow Tasks
  - For each source (CSV, SQL):
    - Flat File Source: To read data from CSV files.
    - OLE DB Source: To read data from the SQL Server orders table.
    - Data Conversion: To convert data types (e.g., string to integer, string to date).
    - Lookup Transformation: To retrieve customer_key from DimCustomer, product_key from DimProduct, and date_key from DimDate.
    - OLE DB Destination: To load data into the Data Warehouse tables.
- Control Flow Task
  - Execute SQL Task: To truncate tables before loading (for initial load or refresh).
- Order of Execution
  - Load DimCustomer
  - Load DimProduct
  - Load DimDate
  - Load OrdersFact

(Dimensions must be loaded before the fact table due to foreign key constraints).

Eg. The Lookup Transformation in SSIS is very important. When loading the OrdersFact table, you'd use a Lookup to find the customer_key in the DimCustomer table based on the customer_id from the orders table.

Step 6: ETL Development - Accumulating Fact Tables

The following code was used to update the OrdersFact table

```sql
ALTER TABLE OrdersFact
ADD     accm_txn_create_time DATETIME,
accm_txn_complete_time DATETIME,
txn_process_time_hours DECIMAL(10,2);
```

A new sql was also created with the following structure

```sql
-- Create table to track completion times
CREATE TABLE FactTransactionCompletion (
    txn_id INT PRIMARY KEY,
    acrm_txn_complete_time DATETIME NOT NULL,
    FOREIGN KEY (txn_id) REFERENCES OrdersFact(order_id)
);
```

SQL for Update (within SSIS OLE DB Command)

```sql
-- Update existing records with creation time (assuming current time)
UPDATE OrdersFact
SET accm_txn_create_time = GETDATE(),
    txn_process_time_hours = NULL;
```