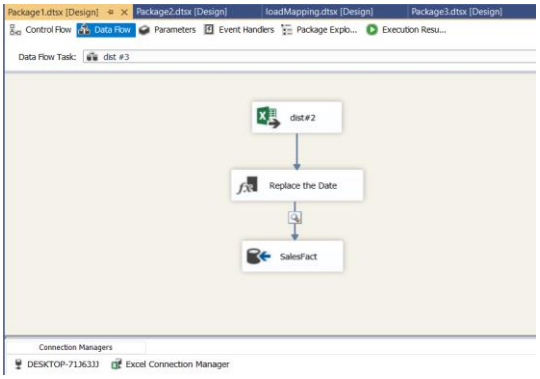
ETL Process: Extracting, Cleaning, and Loading Data from Excel to SQL Server Using SSIS

1. Extracting Data from Excel to SSIS

I used **SSIS (SQL Server Integration Services)** to read

Data from an **Excel** file. This typically involves:

- Adding an **Excel Source** in the **Data Flow Task**.
- Ensuring that column data types are properly mapped.

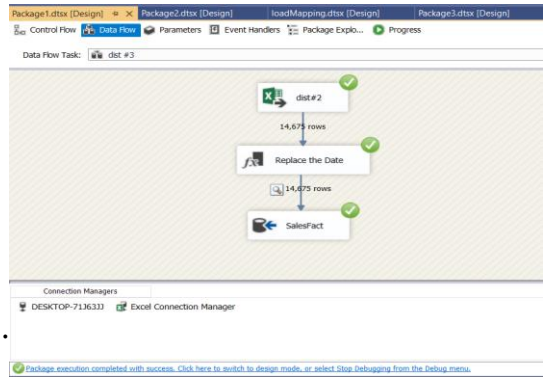


2. Loading Data into SQL Server

Once SSIS extracted the data, I loaded it into

SQL Server using **OLE DB Destination**, which involves:

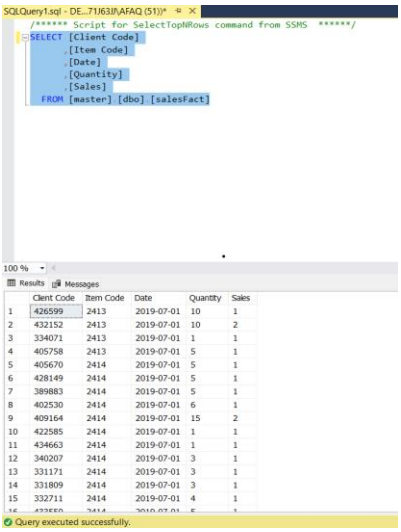
- Defining the **destination table** in the database.
- Mapping columns between Excel and the SQL table.
- Choosing an **insert mode**:
  - **Append** (Add new records without deleting existing ones).
  - **Overwrite** (Replace old data with new data).



3. Cleaning Data Efficiently in SSIS

I used SSIS tools to clean and transform data before loading, which may include:

- **Derived Column**: Creating new columns or modifying existing values.
- **Data Conversion**: Resolving data type mismatches (e.g., converting text to numbers or dates).
- **Conditional Split**: Filtering data based on conditions (e.g., removing blank rows or invalid values).
- **Lookup**: Matching records with reference tables to ensure data consistency.
- **Aggregate**: Performing summarization before loading the data.



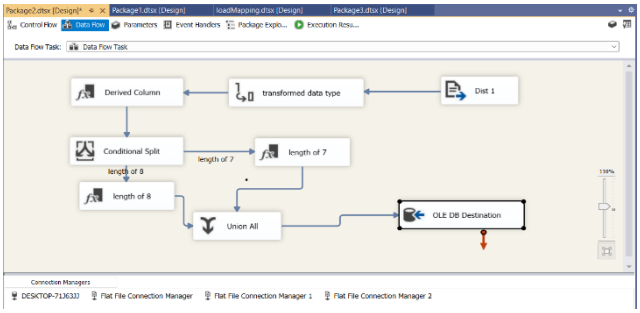
Processing Flat File Data Using SSIS: Extraction, Cleaning, and Loading into SQL Server

After successfully handling Excel data, the next step i completed involved working with a **Flat File**. Here's a breakdown of the process:

1. Extracting Data from a Flat File to SSIS

I used **SSIS** to read data from a **Flat File (CSV, TXT, etc.)** , which requires:

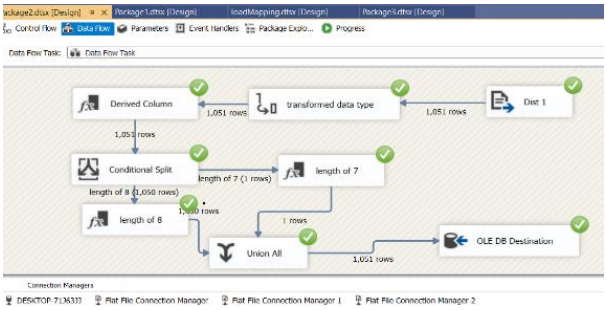
- Adding a **Flat File Source** in the **Data Flow Task**.
- Configuring the **file format** (Delimited, Fixed Width, etc.).
- Setting the correct **column delimiters** .
- Handling encoding issues (e.g., UTF-8, ANSI).



2. Loading Data into SQL Server

After extracting the data, i loaded it into **SQL Server** using **OLE DB Destination**:

- Defined the **destination table** in the SQL database.
- Mapped **columns** between the Flat File and the SQL table.
- Selected an appropriate **loading mode**:
  - **Fast Load** for bulk inserts.
  - **Row-by-Row processing** for complex transformations.

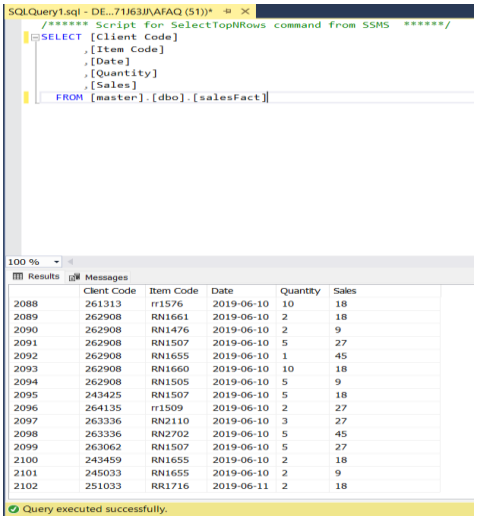


3. Cleaning and Transforming Data in SSIS

Before loading the data into SQL, I used **SSIS transformations**

to clean and enhance the data:

- **Derived Column**: Created new calculated fields or reformatted existing data.
- **Data Conversion**: Fixed data type mismatches (e.g., converting text to numbers or dates).
- **Conditional Split**: Separated records based on conditions (e.g., filtering out invalid data).
- **Lookup Transformation**: Validated data against a reference table in SQL Server.
- **Replace Nulls**: Handled missing values to avoid errors in SQL queries.



```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT [Client Code]
      ,[Item Code]
      ,[Date]
      ,[Quantity]
      ,[Sales]
  FROM [master].[dbo].[salesFact]
```

	Client Code	Item Code	Date	Quantity	Sales
2088	261313	rr1576	2019-06-10	10	18
2089	262908	RN1661	2019-06-10	2	18
2090	262908	RN1476	2019-06-10	2	9
2091	262908	RN1507	2019-06-10	5	27
2092	262908	RN1655	2019-06-10	1	45
2093	262908	RN1660	2019-06-10	10	18
2094	262908	RN1505	2019-06-10	5	9
2095	243425	RN1507	2019-06-10	5	18
2096	264135	rr1509	2019-06-10	2	27
2097	263336	RN2110	2019-06-10	3	27
2098	263336	RN2702	2019-06-10	5	45
2099	263062	RN1507	2019-06-10	5	27
2100	243459	RN1655	2019-06-10	2	18
2101	245033	RN1655	2019-06-10	2	9
2102	251033	RR1716	2019-06-11	2	18

Query executed successfully.