

Data Warehousing & Business Intelligent

3rd Year, 1st Semester

# Assignment 1

Submitted to

Sri Lanka Institute of Information Technology

Bachelor of Science Special Honors Degree in Data Science

IT20038700

S.A.C.H.Senadeera

Weekend Batch

**Step 01: Data Set Selection**

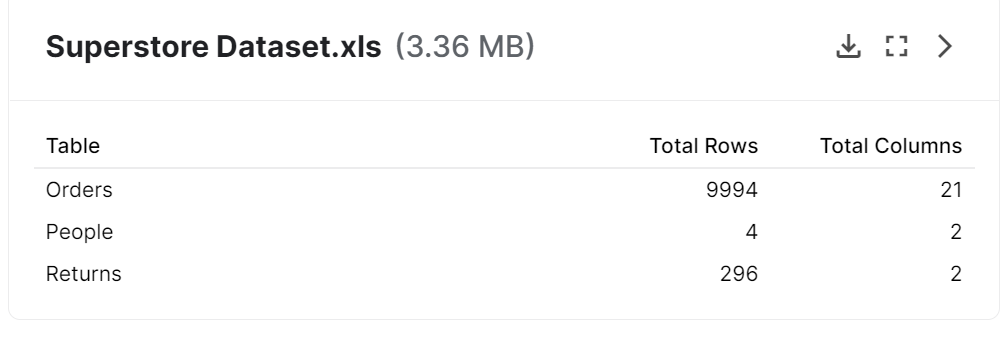
Data Set : Superstore Sales Data Set

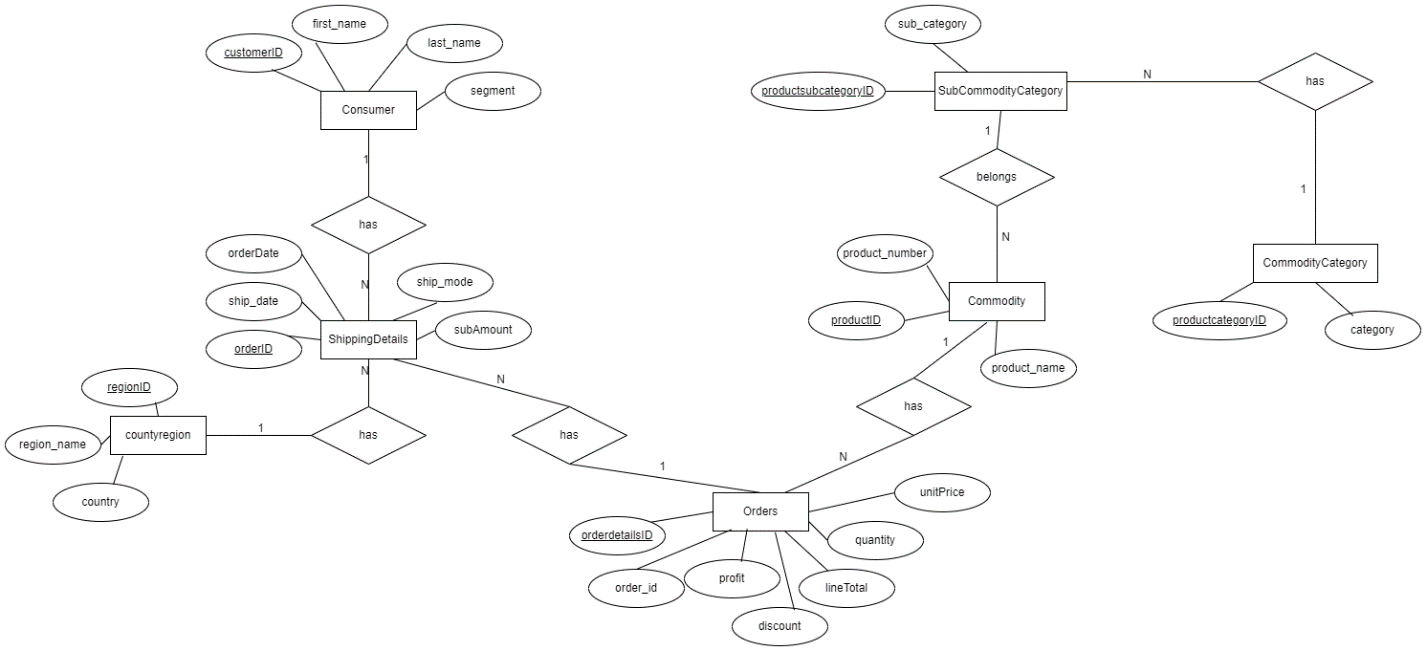
Source : Kaggle

Link to the Source : <https://www.kaggle.com/datasets/chiragrathi/superstore-sales-dataset>

The Data Set consists of a large CSV file along with two small CSV files. I have partitioned the main large CSV file into small sub CSV files. The sub CSV files consists of new IDs.

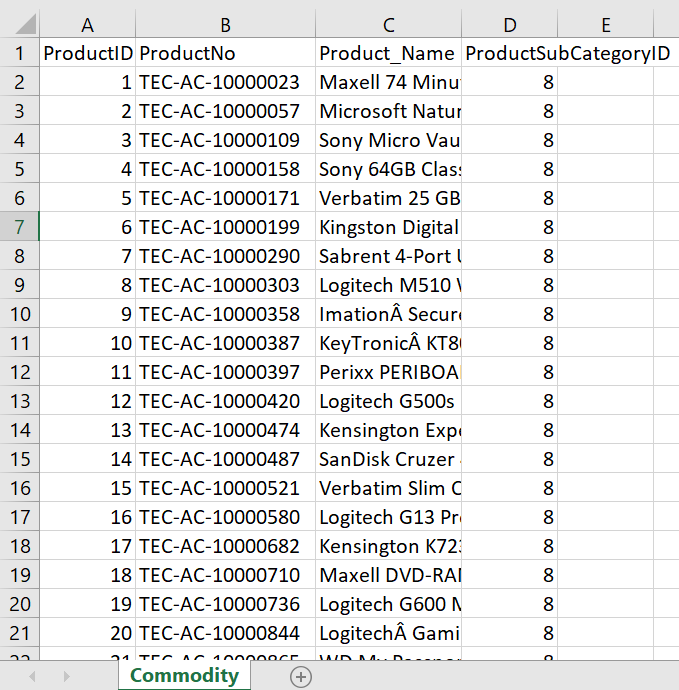
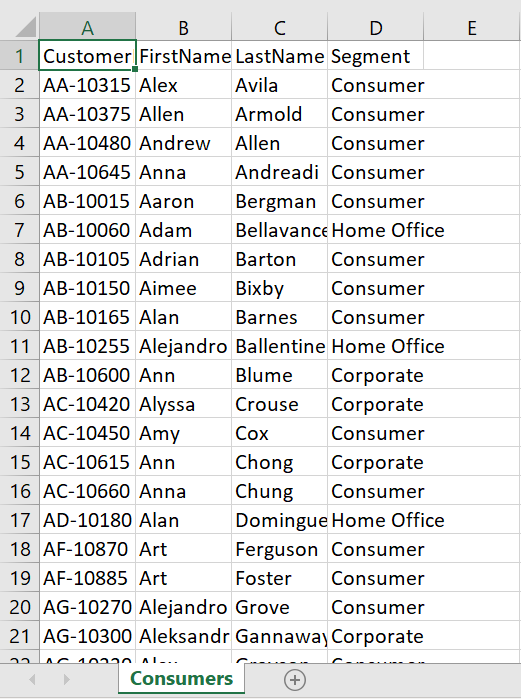
The Data Set was initiated with sufficient data, according to the assignment principles. The approval of choosing this data set was taken by the respected lecture.

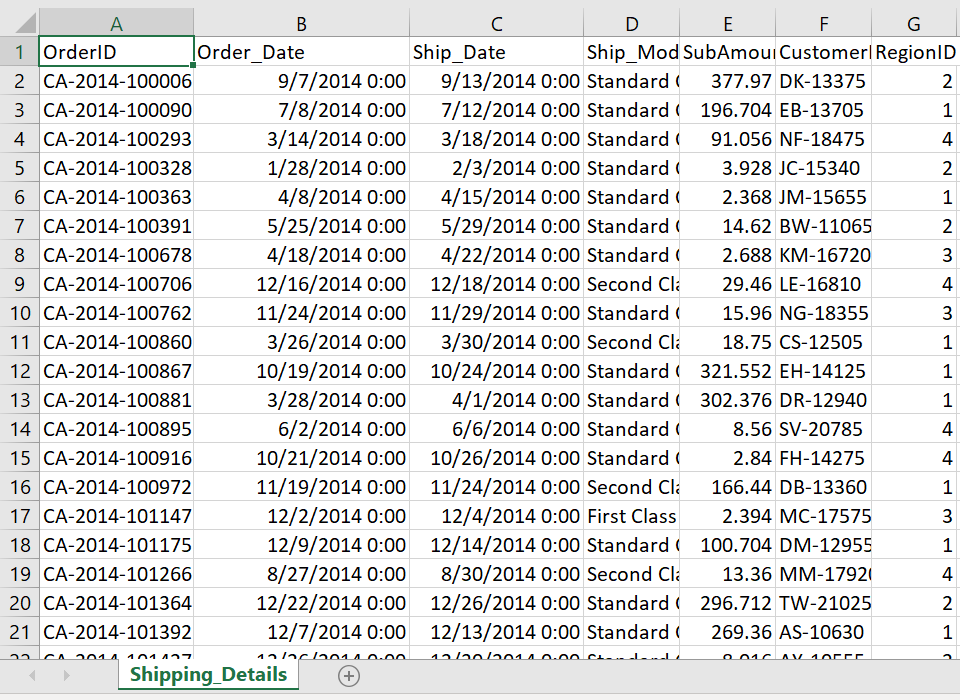
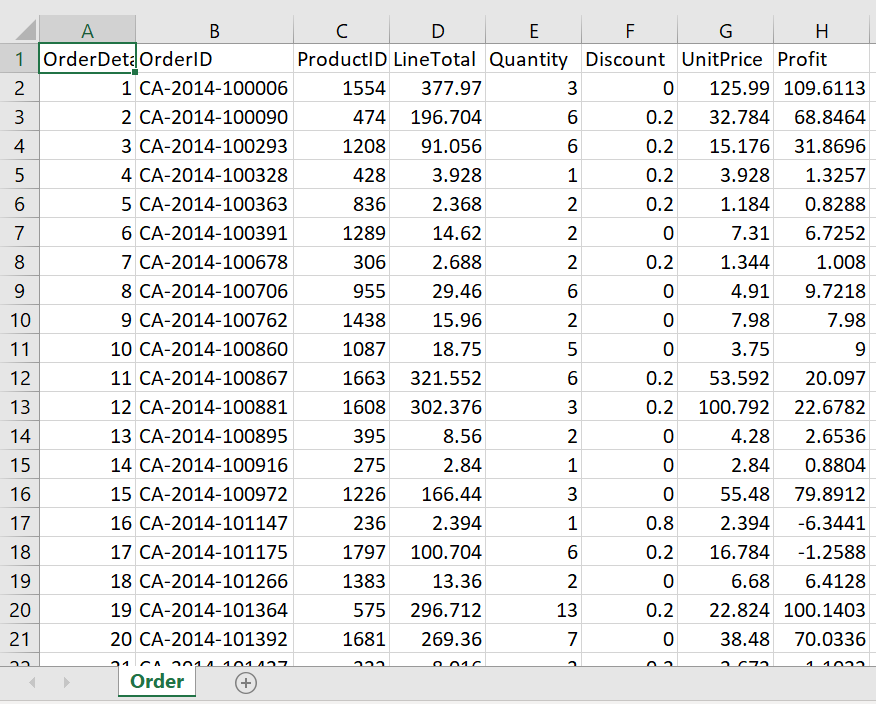


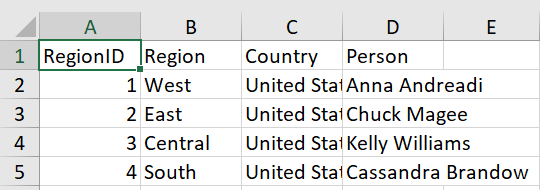
ER Diagram

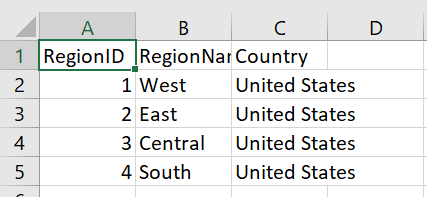
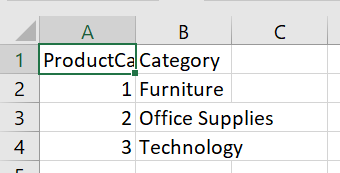
**Step 02: Preparation of Data Sources**

The implement data warehouse used data set view, it is Store Sale transaction data set, it partitioned into separate Source files to implement data warehouse, such as Consumers.csv, Commodity.csv, SubCommodityCategory.csv, CommodityCategory.csv, CountryRegion.csv, Orders.csv, ShippingDetails.csv, SalesConsumerRegion.csv and ConsumerLocations.txt Text file.

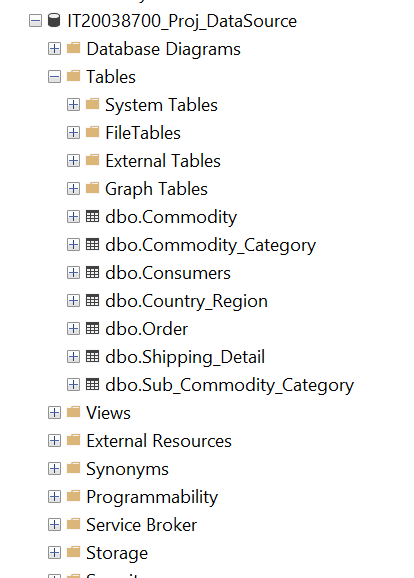






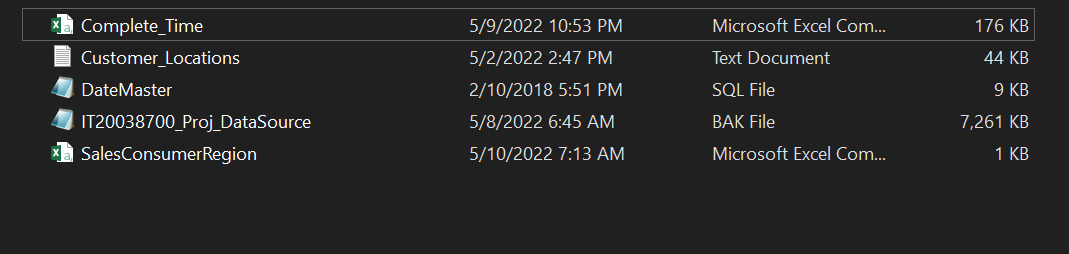


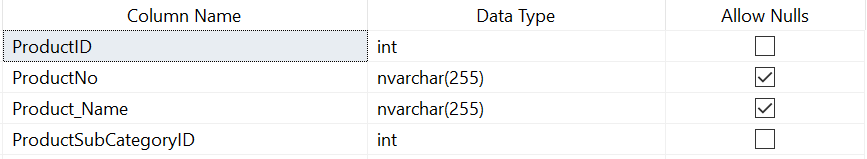
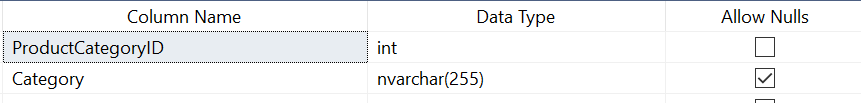
I have Loaded Commodity.csv, CommodityCategory.csv, Consumers.csv, CountryRegion.csv, Orders.csv, ShippingDetails.csv and SubCommodityCategory.csv files to the DB Called IT20038700\_Proj\_DataSource.

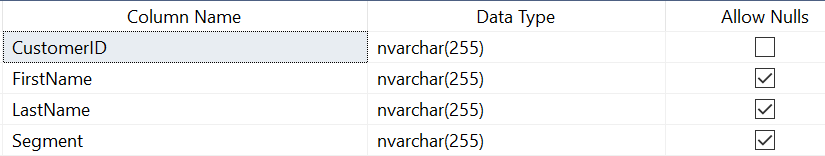
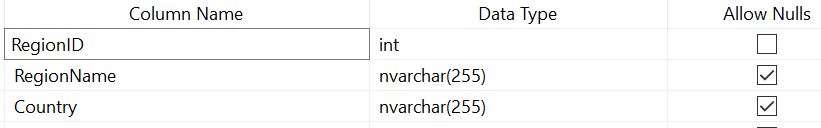


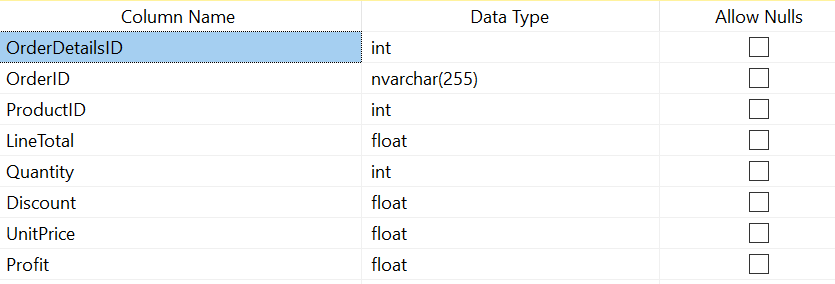
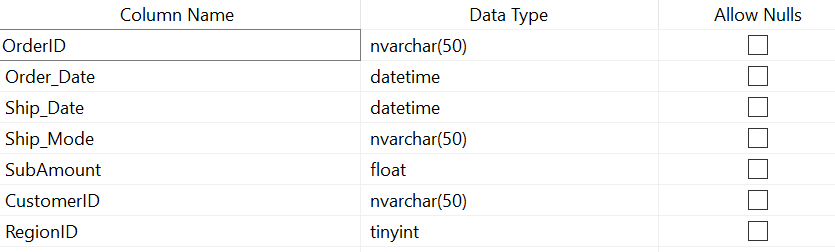
I have converted ConsumerLocation.csv to Text file so that I can use Flat Source in SSIS tool, to extract ConsumerLocation.txt data.

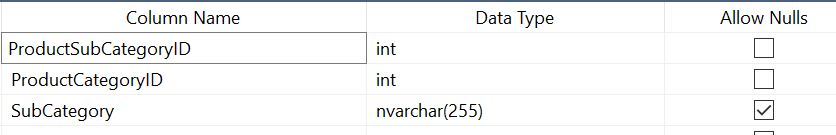
I have used separate data file called SalesConsumerRegion.csv file. It can use Flat Source in SSIS tool, to extract data.

Final Set of Sources

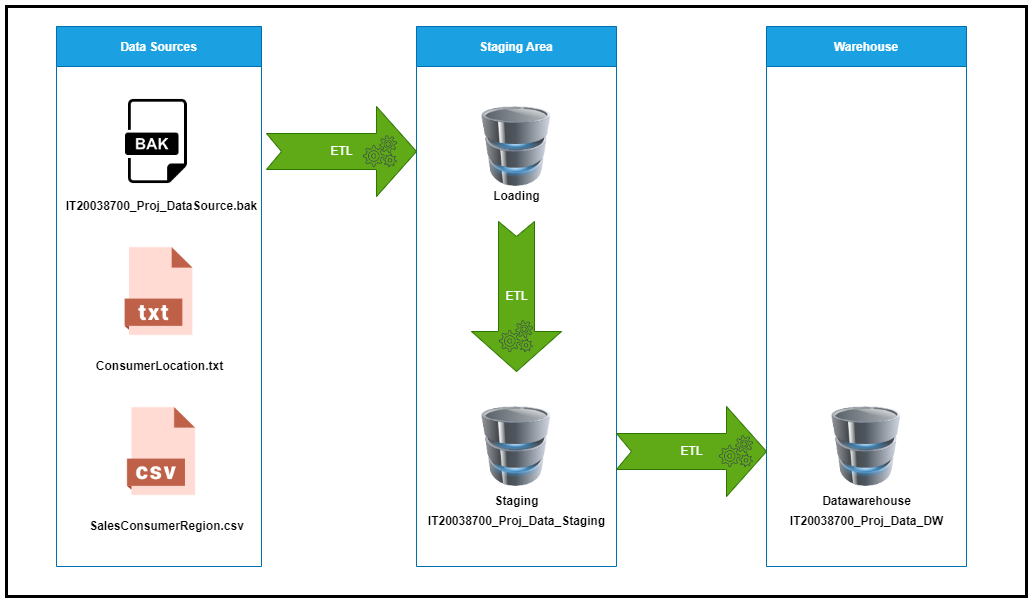
Describe Sources

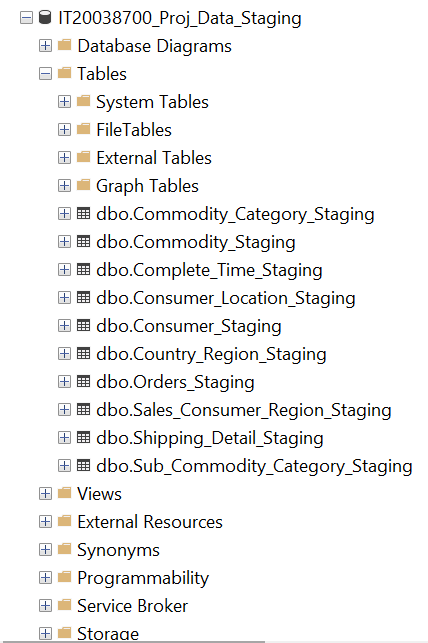






**Step 03: Solution Architecture**

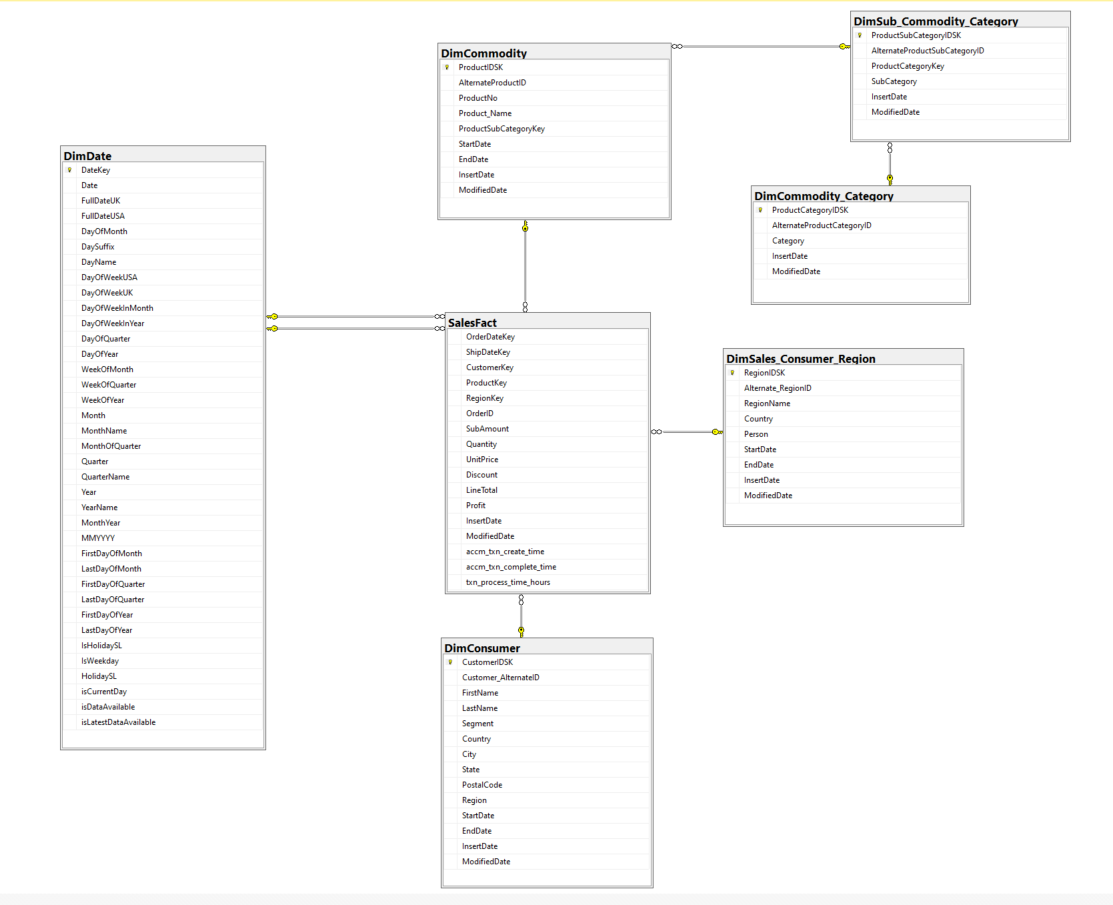




* As can be seen in the above image, different resource types have been used to extract data to staging. Staging layer has been used to have all the tables in a single location as in the below image.
* The tables at the staging are then profiled and after performing a rich set of ETL tasks, data is loaded to the data warehouse where from that several reporting tools and analysing tools can use data for reporting mining and analysing.

**Step 04: Data Warehouse Design & Development**

The datawarehouse is designed as a snow flake schema with one fact table and six dimension table.



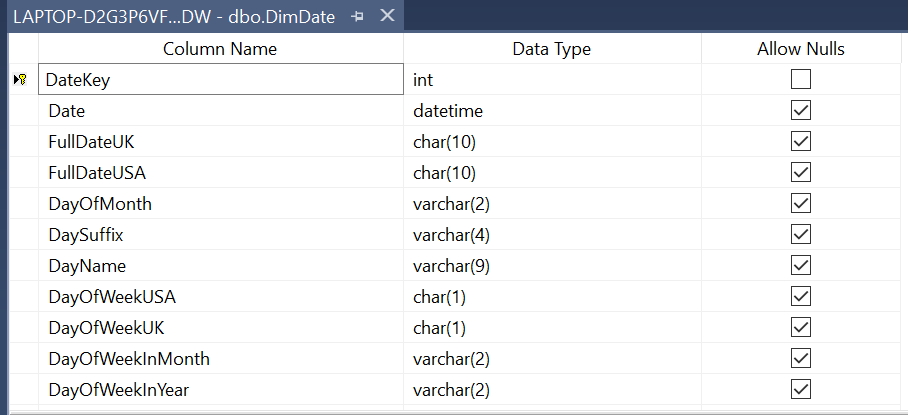
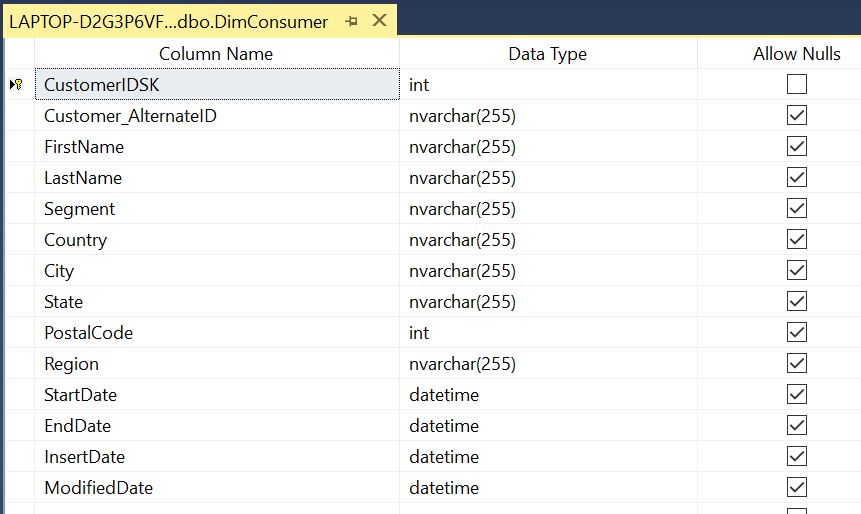
**Assumption**

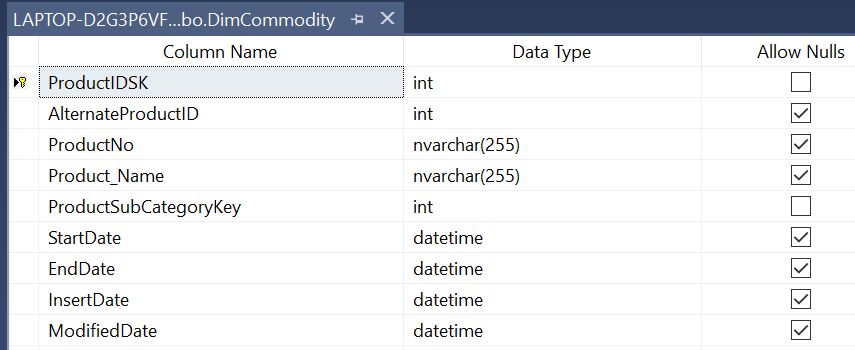
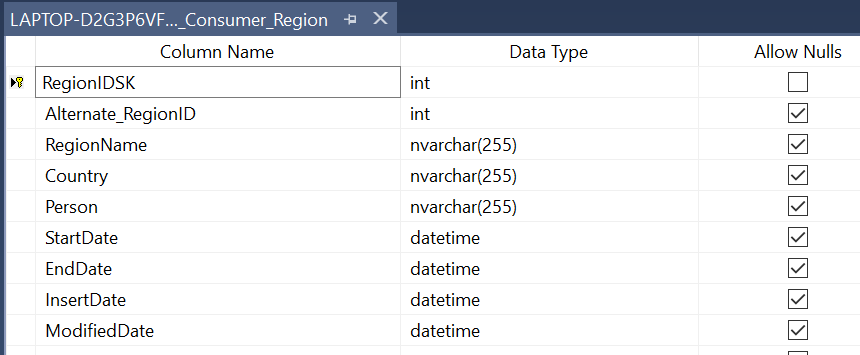
* I have taken DimConsumer as slowly changing dimension, Consumer addresses can change time to time, and we need to keep track of their historical address.
* In the data set product name is too large So assume that if Store owner wants to change the product name and we need to keep track of their historical name so that I have taken DimCommodity as slowly changing dimension.
* The reason for taking Dim Region Sales Representative as slowly changing dimension, is  
  Sales Representative’s state can be change time to time, and we should have to keep track of their historical Representative record. Additionally, if the Region Sales Representative  
  changes, we should replace the old Representative with the new Representative.

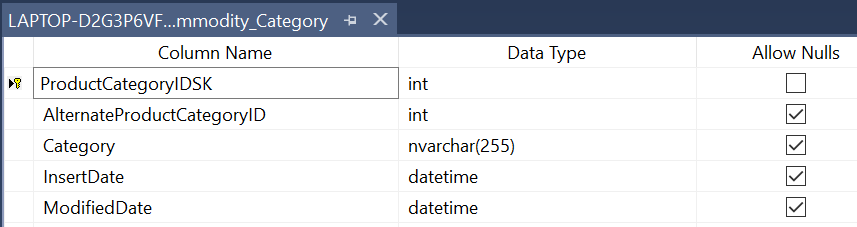
**Calculations**

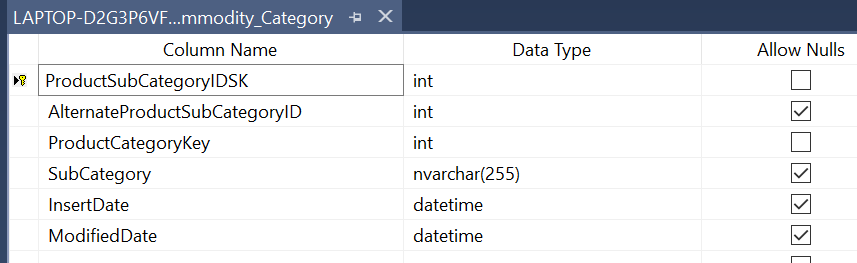
* SubAmount in SalesFact table is calculated by ([UnitPrice\*Quantity]-Discount)

Before creating the sales fact table & Other dimensions, start by creating the Date Dimension.  
For that I use the code in the file ‘DateMaster.sql’ file.

**Dimension Tables**

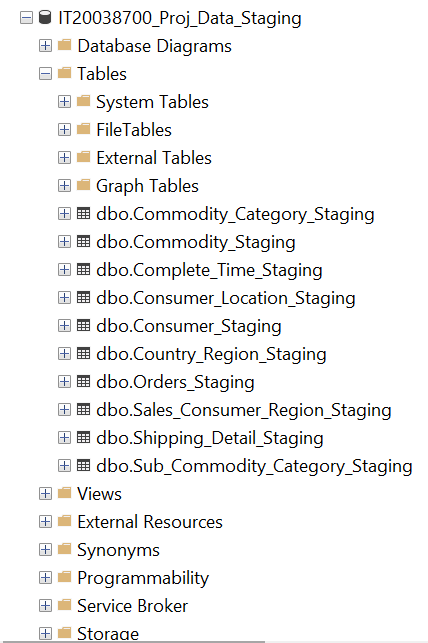
****

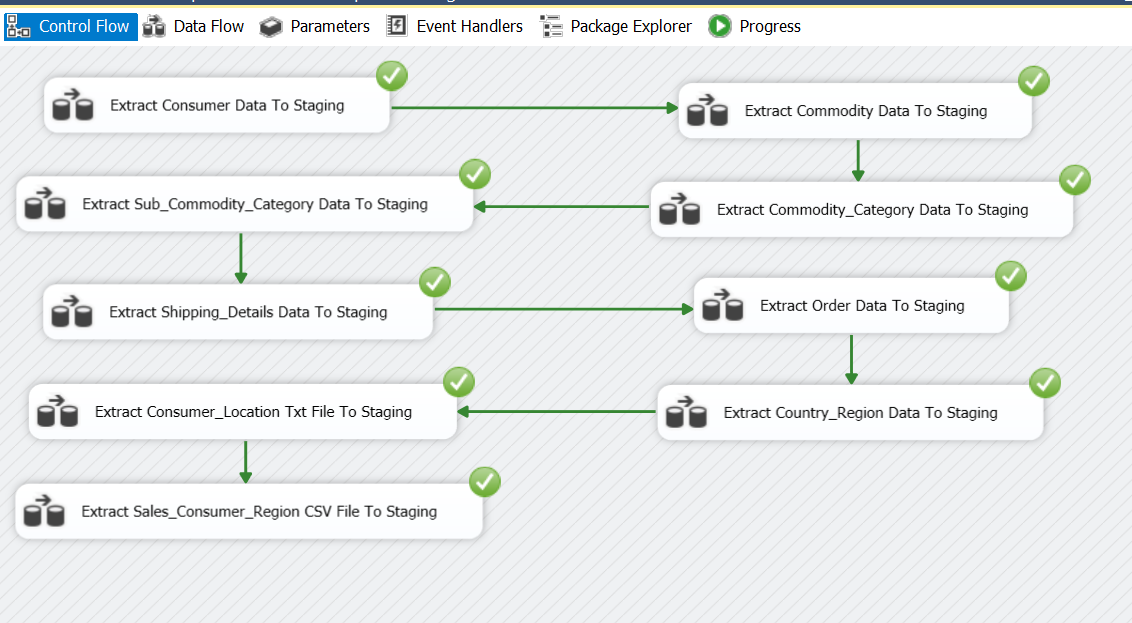
****



**Step 05: ETL Development**

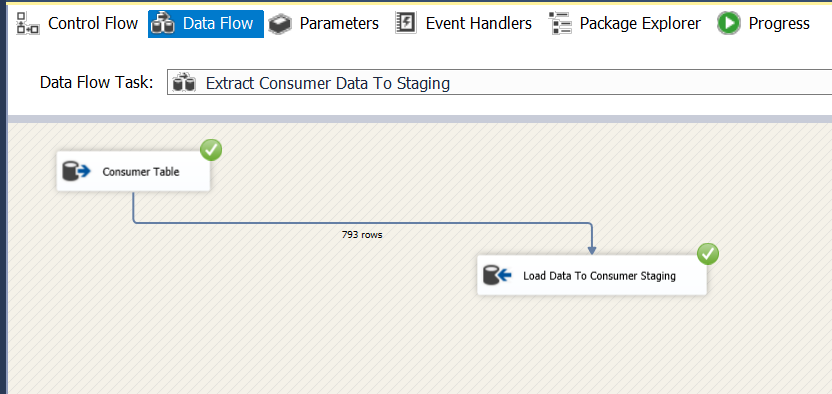
First using the SQL Server Integration Services Software, I have extracted all the data from the tables  
which were in the IT20038700\_Proj\_DataSource, Extract ConsumerLocation.txt and SalesConsumerRegion.csv to separate staging DB called IT20038700\_Proj\_Data\_Staging as shown in the  
below .



Overall Control Flow

## Data Extraction

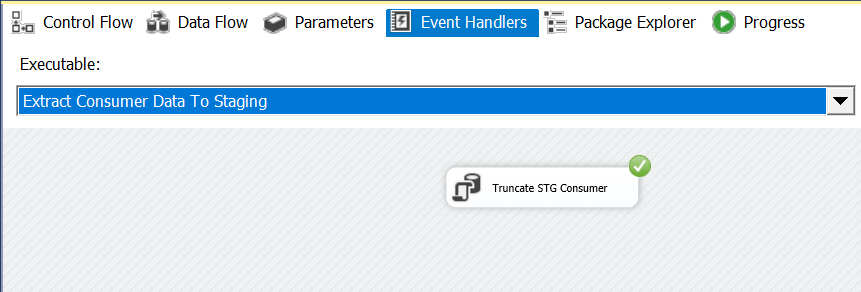
1. Consumer Data from Source to Staging

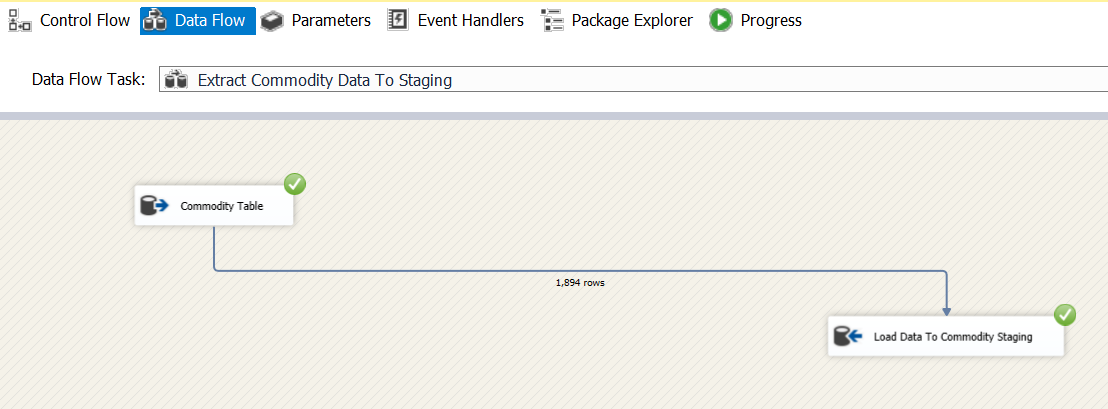


Consumer Data in Consumer Database Table has been extracted and loaded to Consumer\_Staging table

Event Handler

Before executing ‘extract Consumer Data to staging’ existing data in the staging layer has been truncated.

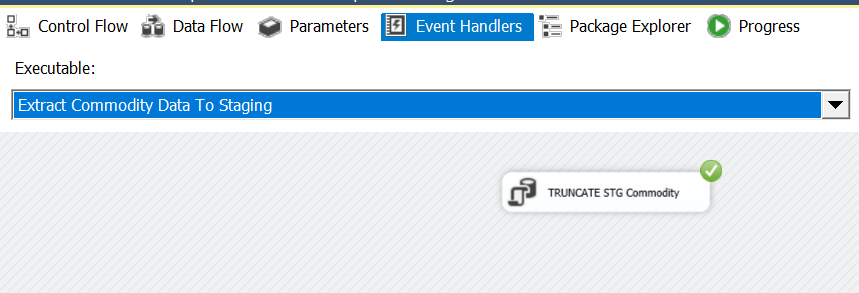


1. Commodity Data from Source to Staging

Commodity Data in Commodity Database Table has been extracted and loaded to Commodity\_Staging table

Event Handler

Before executing ‘extract Commodity Data to staging’ existing data in the staging layer has been truncated.

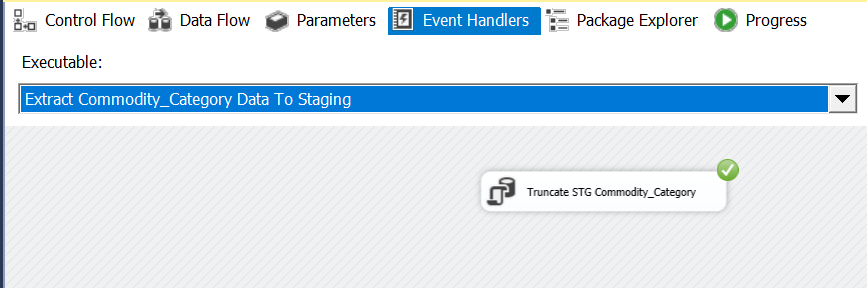


1. Commodity\_Category Data from Source to Staging

Commodity Category Data in Commodity\_Category Database Table has been extracted and loaded to Commodity\_Category\_Staging table

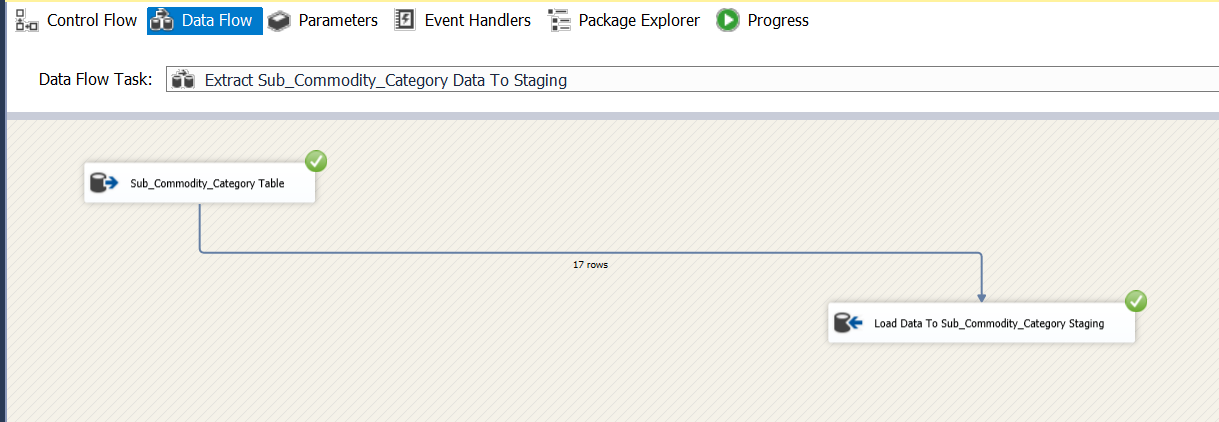
Event Handler

Before executing ‘extract Commodity Category Data to staging’ existing data in the staging layer has been truncated.



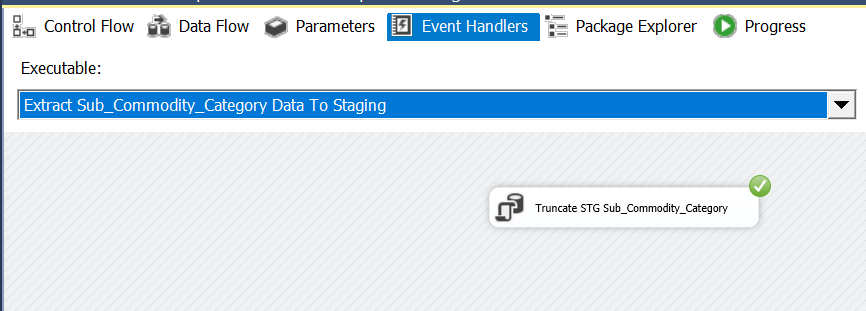
1. Sub\_Commodity\_Category Data from Source to Staging

Sub Commodity Category Data in Sub\_Commodity\_Category Database Table has been extracted and loaded to Sub\_Commodity\_Category\_Staging table



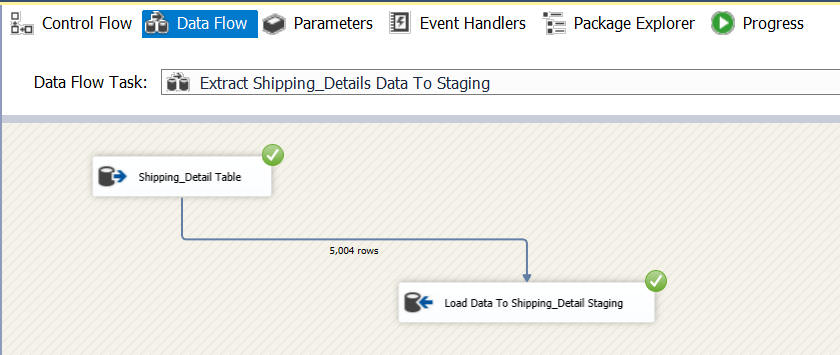
Event Handler

Before executing ‘extract Sub Commodity Category Data to staging’ existing data in the staging layer has been truncated.



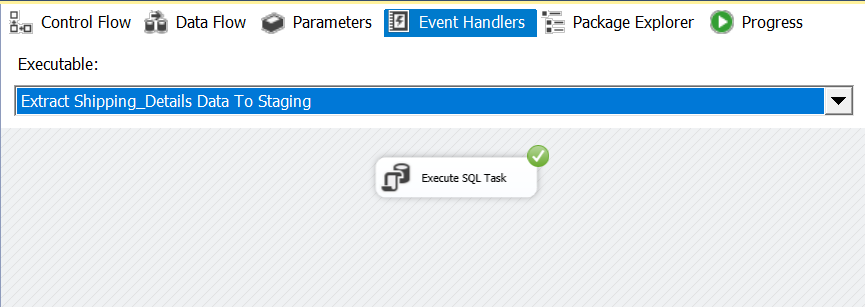
1. Shipping\_Details Data from Source to Staging

Shipping Detail Data in Shipping\_Details Database Table has been extracted and loaded to Shipping\_Details\_Staging table

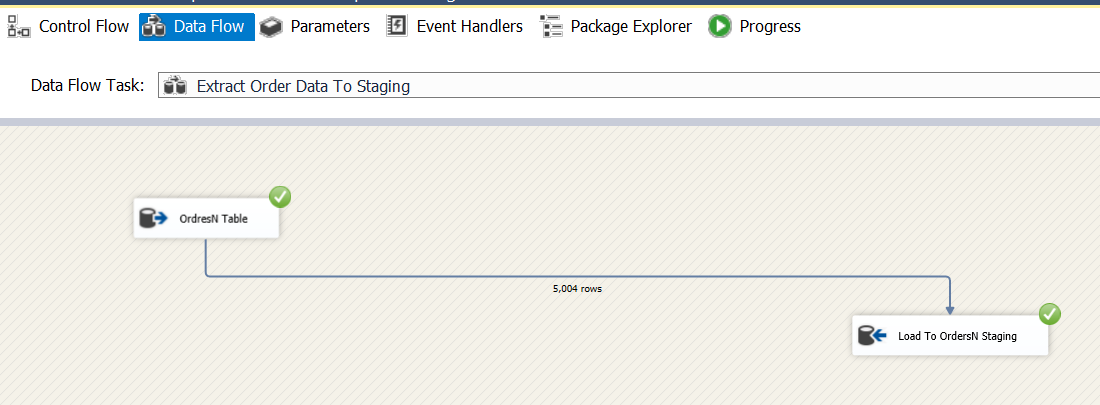


Event Handler

Before executing ‘extract Sub Commodity Category Data to staging’ existing data in the staging layer has been truncated.



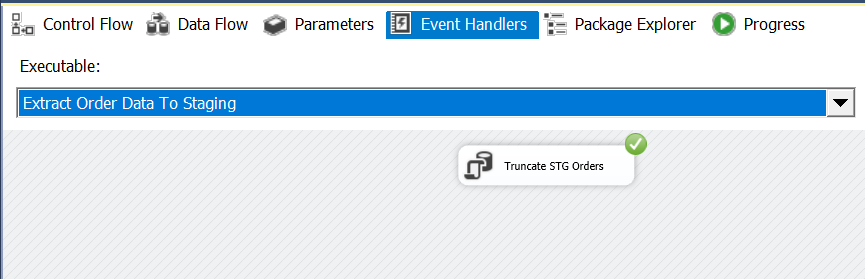
1. Order Data from Source to Staging

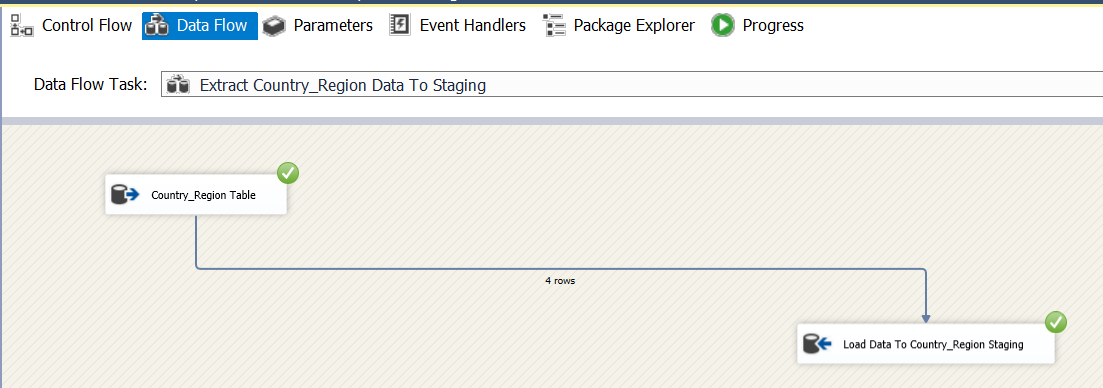


Order Data in Orders Database Table has been extracted and loaded to Order\_Staging table

Event Handler

Before executing ‘extract Order Data to staging’ existing data in the staging layer has been truncated.

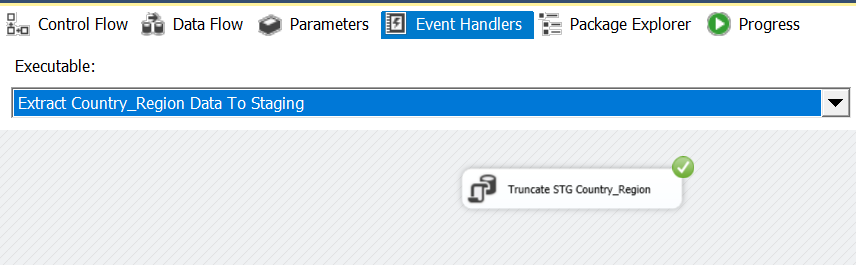


1. Country\_Region Data from Source to Staging

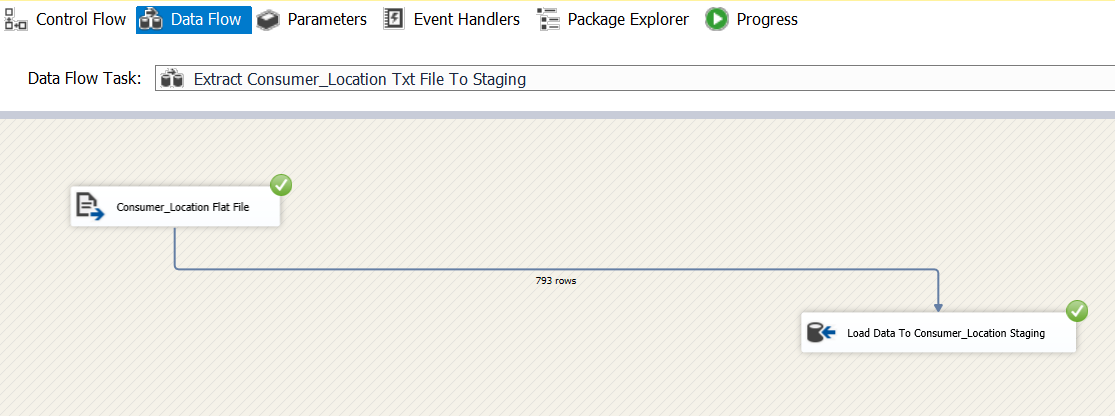
Country Region Data in Country\_Region Database Table has been extracted and loaded to Country\_Region \_Staging table

Event Handler

Before executing ‘extract Country Region Data to staging’ existing data in the staging layer has been truncated.



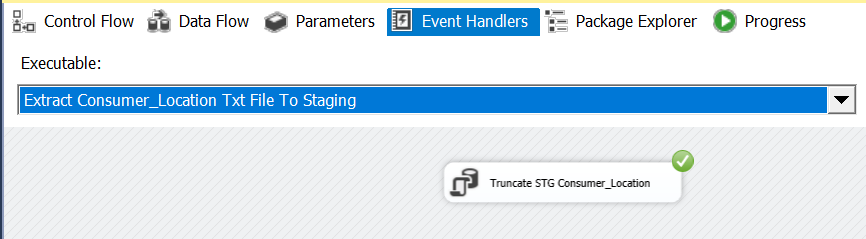
1. Consumer\_Location Data Text File to Staging

Used Flat file Source SSIS tool, to extract ConsumerLocation.txt data .

Customer Locations Data in ConsumerLocation.txt has been extracted and loaded to ConsumerLocation \_Staging table

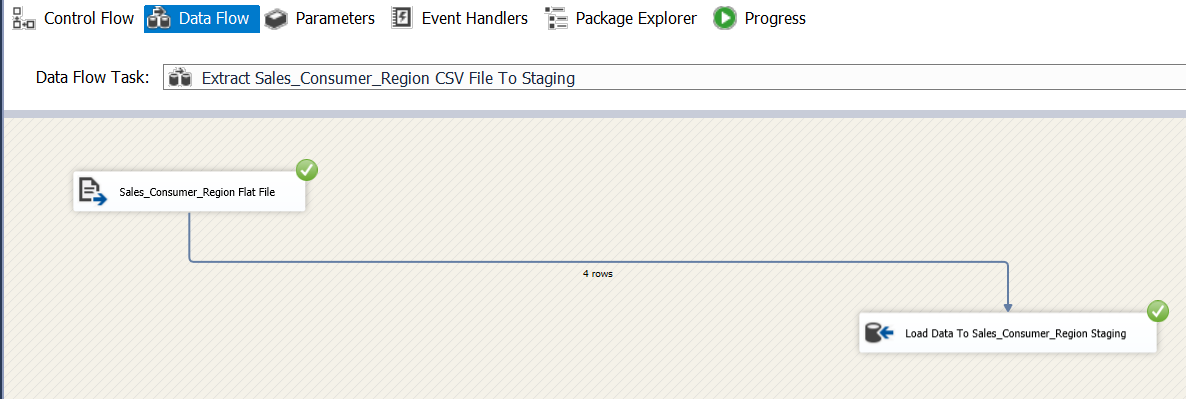
Event Handler

Before executing ‘extract Consumer Locations Data to staging’ existing data in the staging layer has been truncated.



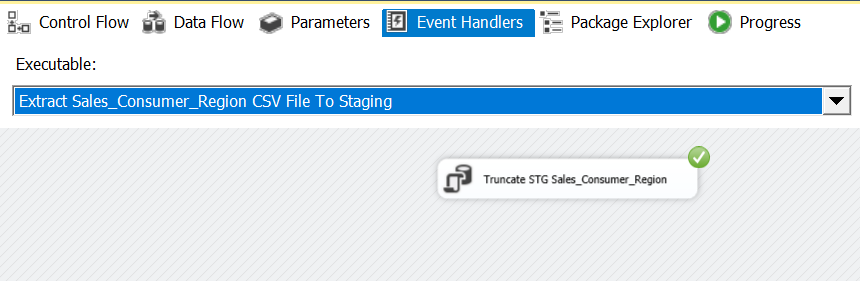
1. Sales\_Consumer\_Region Data CSV File to Staging

Sales Consumer Regions Data in Sales\_Consumer\_Regions.csv has been extracted and loaded to Sales\_Consumer\_Regions \_Staging table



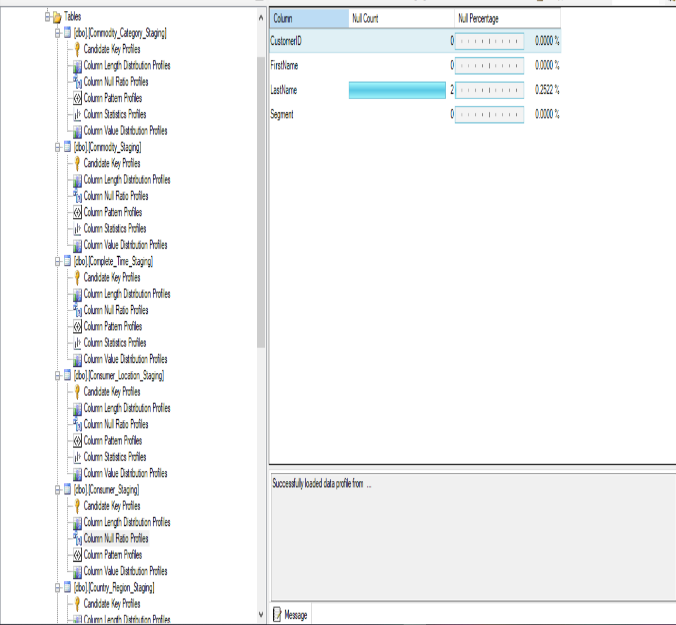
Event Handler

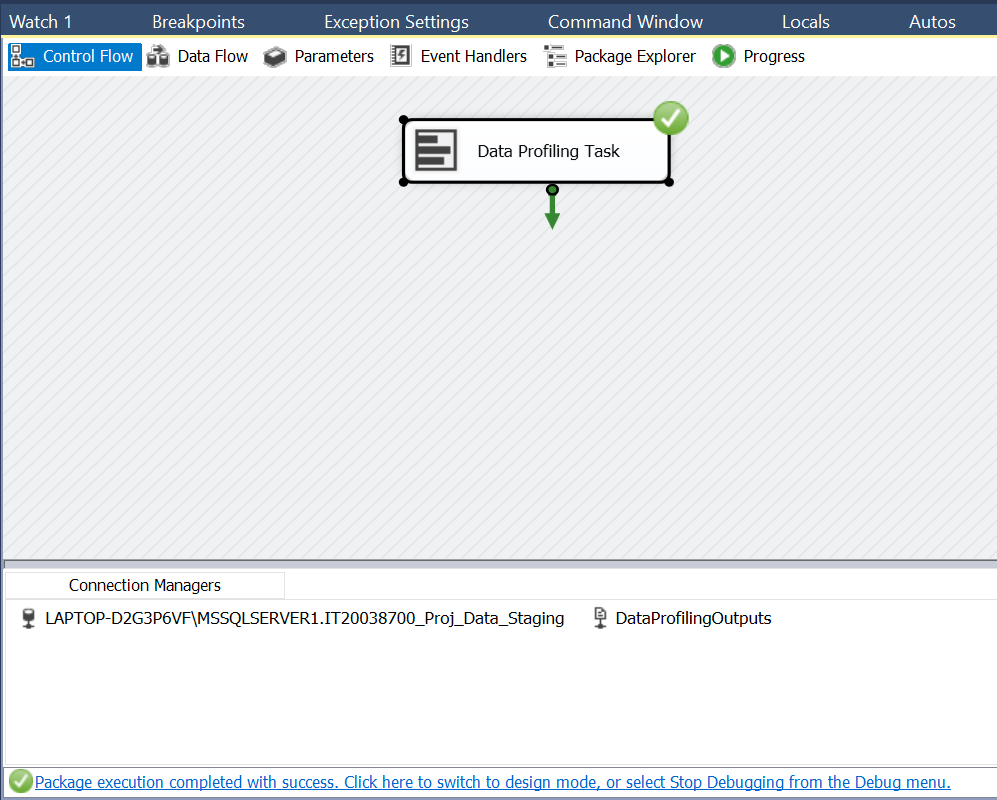
Before executing ‘extract Sales Consumer Regions Data to staging’ existing data in the staging layer has been truncated.



Data Profiling

Before Loading staging tables to the data warehouse data has to be enriched to obtain the most suitable data for analyzing. Data profiling has been done in order to identify what need to be corrected in ETL process in order to meet this requirement.

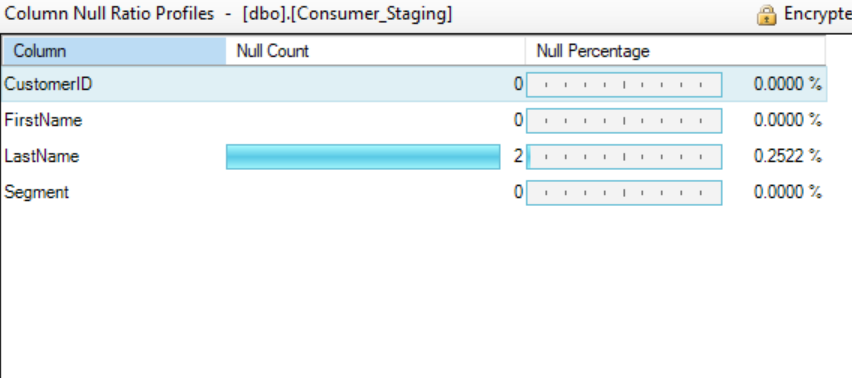


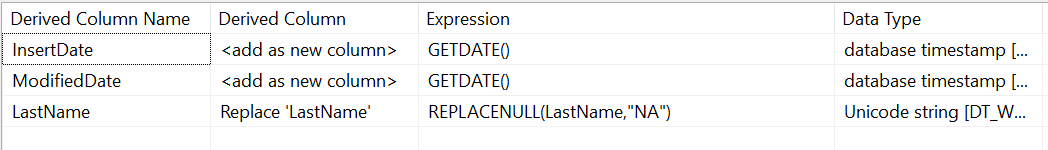


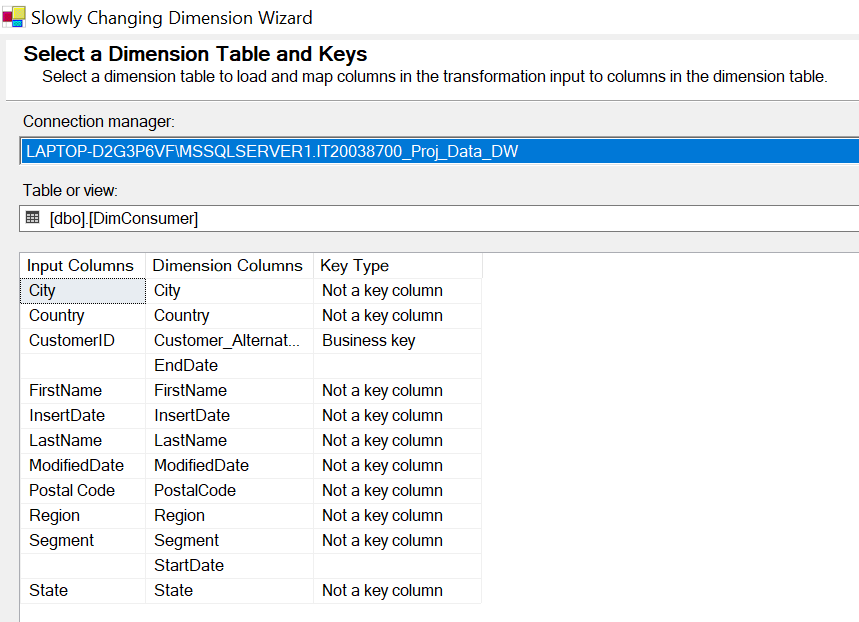
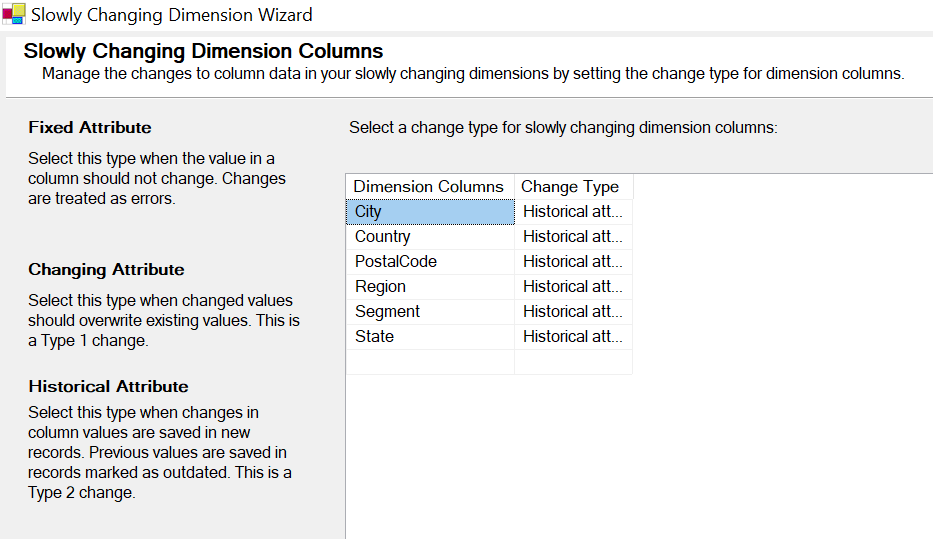
Data Transforming and Loading

1. **Transform and Load Consumer Details (Slowly Changing Dimension Table)**

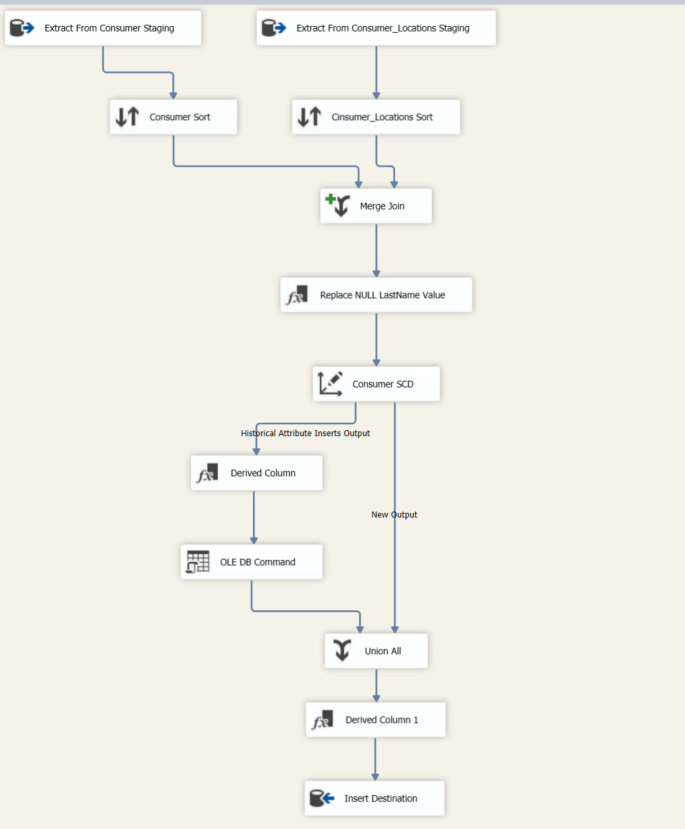
* Consumer table in the staging has been merged into Customer\_Location table since there are only few rows in Customer\_Location table. Merge has been performed by sorting both the tables using the common filed ‘CustomerID’.
* I Dragged and dropped Derived Column item and connect the Merge Join item to  
  Derived Column because I have found in the data profiling part Consumer’s last name  
  Contain two null value.



* Then I Replace the Replace null by using the Derived Column
* DimConsumer dimension has been identified as a slowly changing dimension. Hence necessary steps have been followed to make DimConsumer a slowly changing dimension.

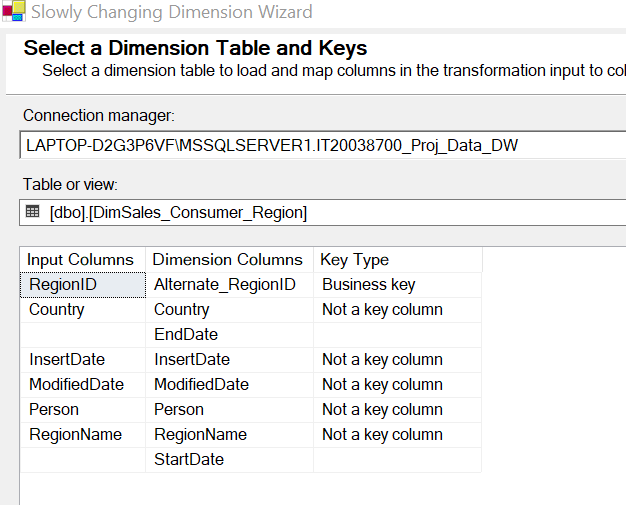
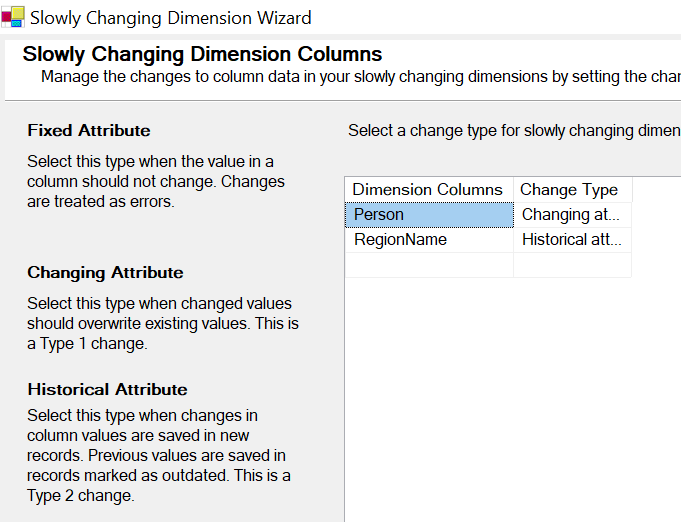


* Following all the steps ,finally Consumer data has been Loaded to DimConsumer table in data warehouse.

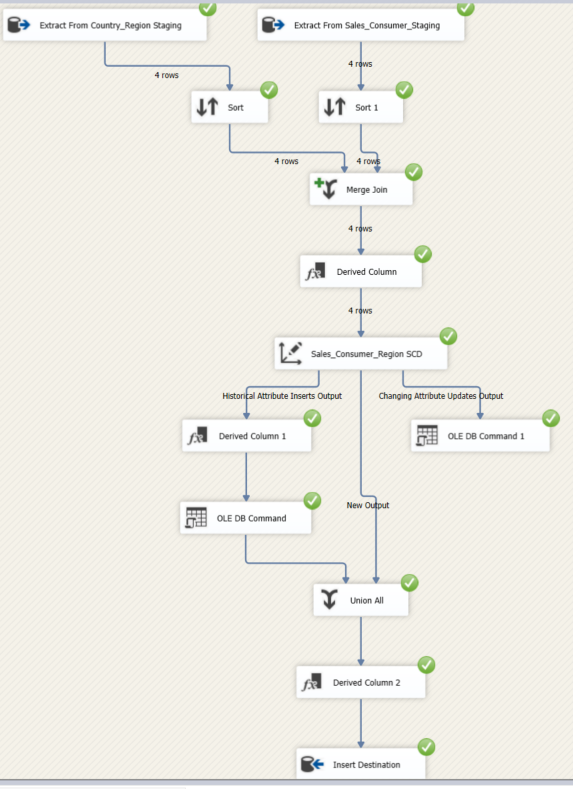


1. **Transform and Load Sales\_Consumer\_Region Details (Slowly Changing Dimension Table)**

* The reason for taking DimSalesConsumerRegion as slowly changing dimension, is  
  SalesConsumerRegion’s state can be change time to time, and we should have to keep track of their historical Representative record. Additionally, if the SalesConsumerRegion  
  changes, we should replace the old Region with the new Region.

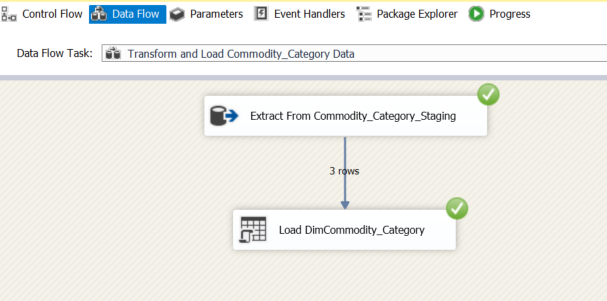


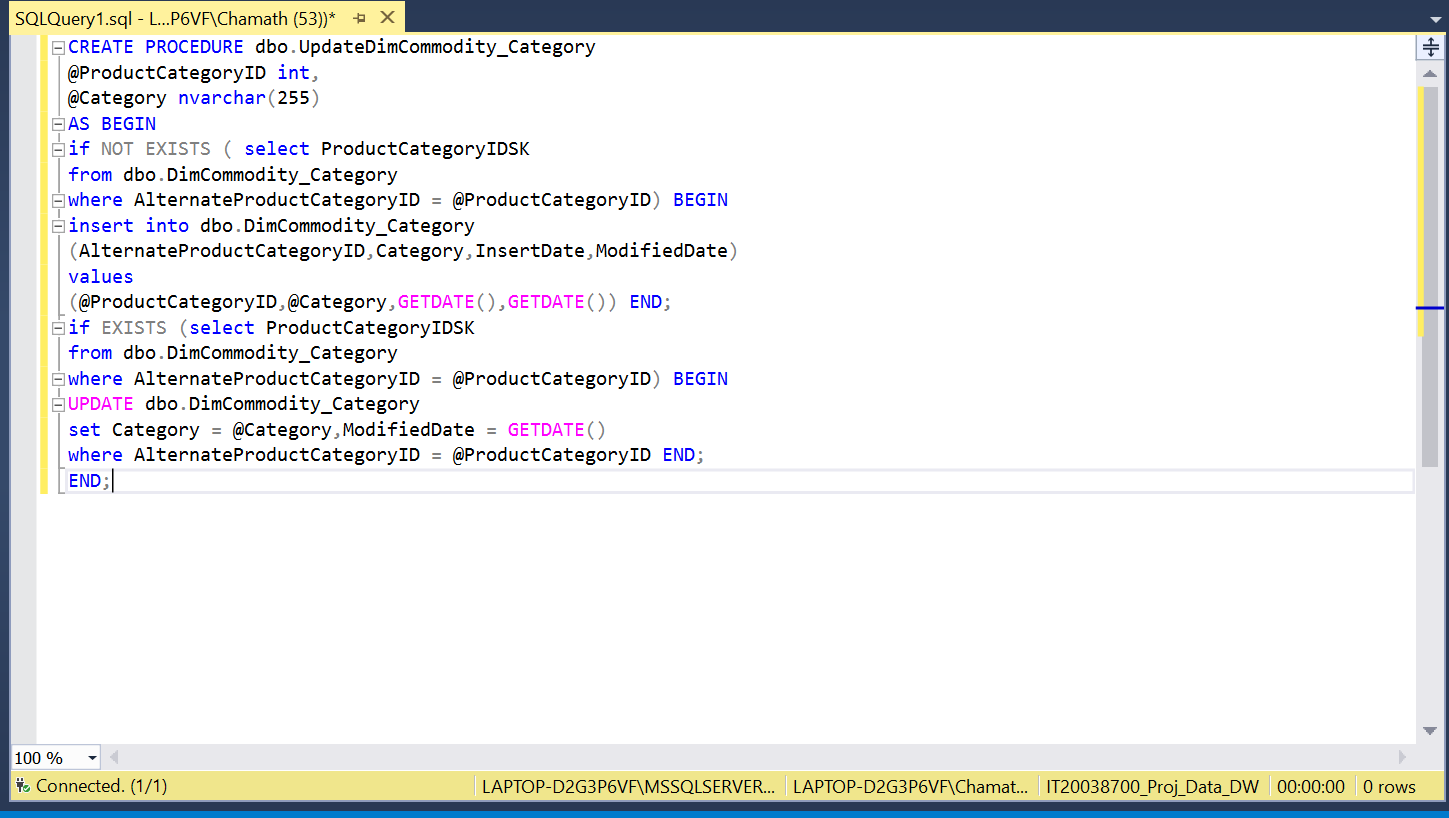
* Following all the steps ,finally SalesConsumer data has been Loaded to DimSales\_Consumer\_Region table in data warehouse.



1. **Transform and Load Commodity\_Category Details**

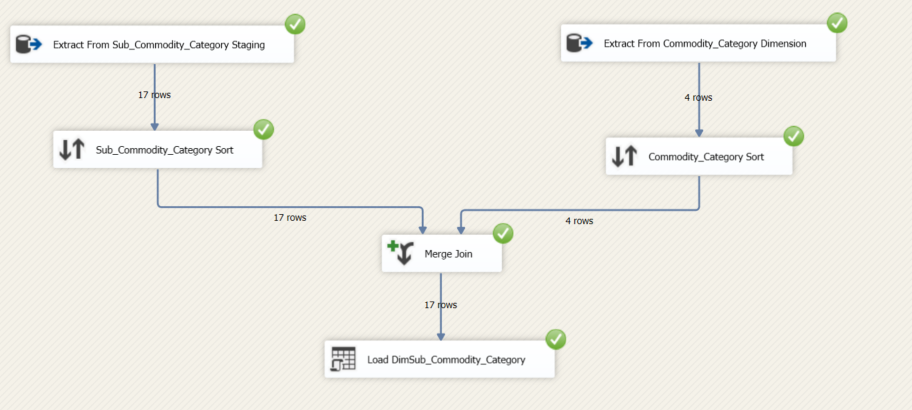
Commodity Category data has been loaded to Dim Commodity\_Category



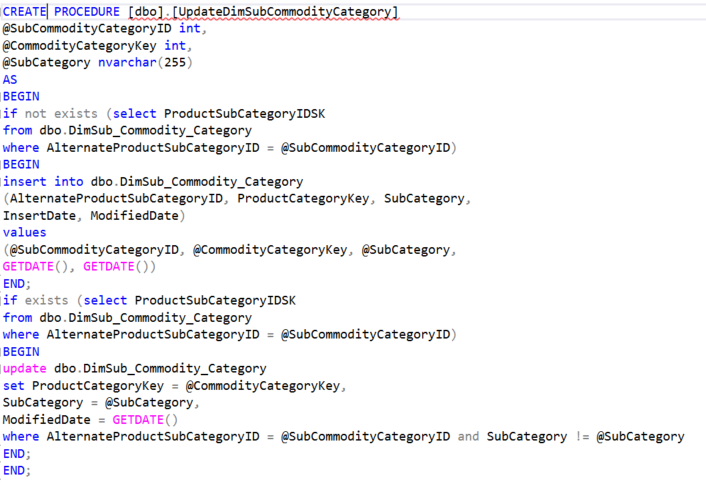
* Commodity Category data has been loaded to DimCommodity\_Category table in Datawarehouse. The following procedure is used in order to load data.

1. **Transform and Load Sub\_Commodity\_Category Details**

* Use two OLE DB SOURCE to Commodity Sub category Staging and  
  DimCommodity\_Catagory in **IT20038700\_Proj\_DW** database after use each source to Sort, SSIS tool and sort by ProductCategoryID in Commodity Subcategory Staging table and DimCommodity\_Catagory Dimension table by AlternateProductCategoryID

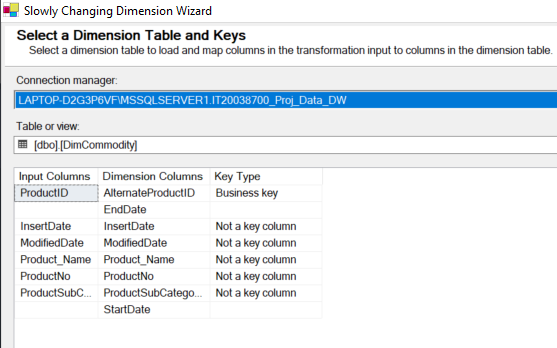
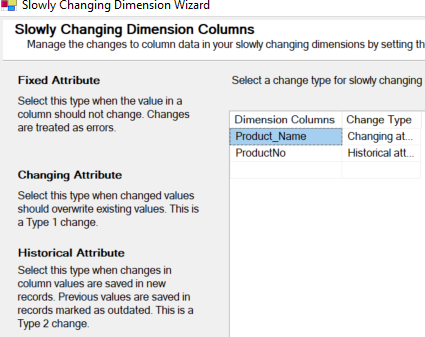


* Sub Commodity Category data has been loaded to DimSub\_Commodity\_Category table in Datawarehouse. The following procedure is used in order to load data.

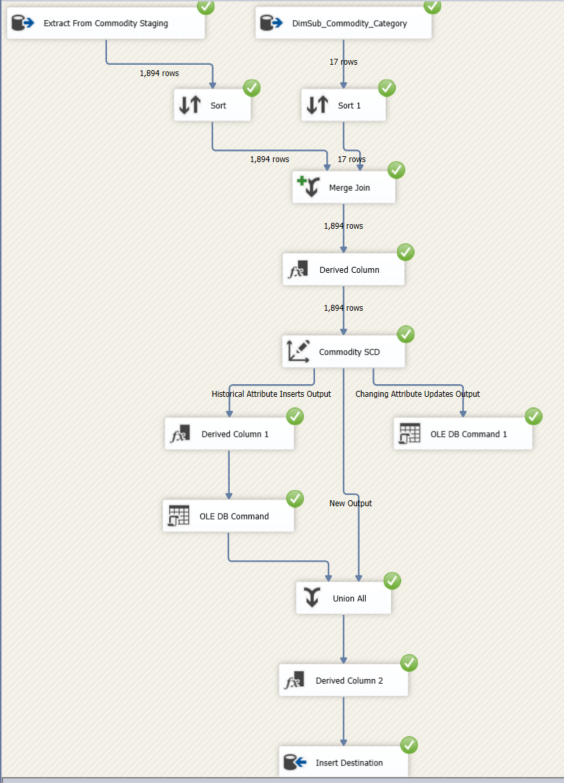


1. **Transform and Load Commodity Details (Slowly Changing Dimension Table)**

* Use two OLE DB SOURCE to Commodity\_staging and DimSub\_Commodity\_Catagory in  
  in **IT20038700\_Proj\_DW** database after use each source to Sort ,SSIS tool and sort by  
  ProductSubCategoryID Commodity\_staging table and DimSub\_Commodity\_Catagory table  
  AlternateProductSubCategoryID.

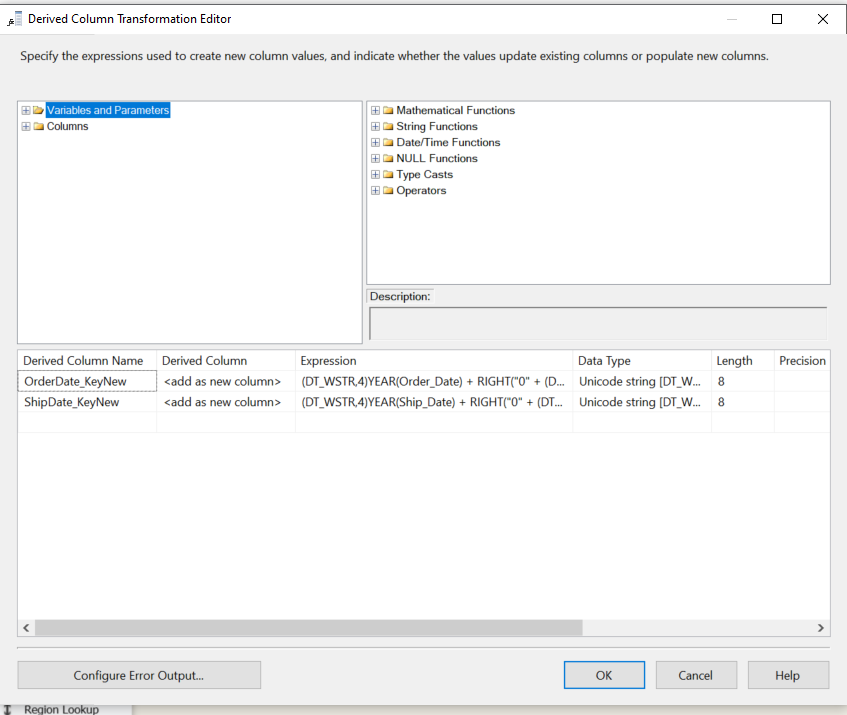


* Following all the steps ,finally Commodity data has been Loaded to DimCommodity table in data warehouse.

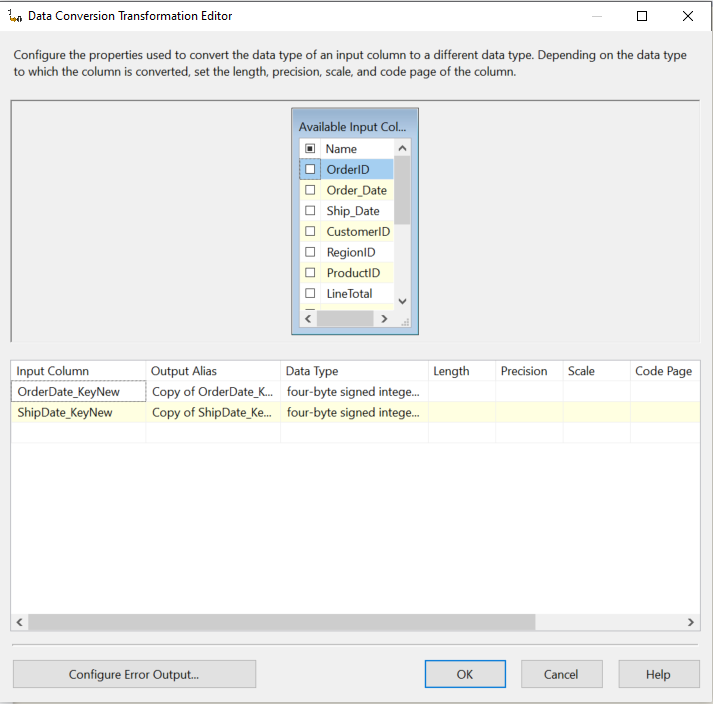


1. **Transform and Load SalesFact**

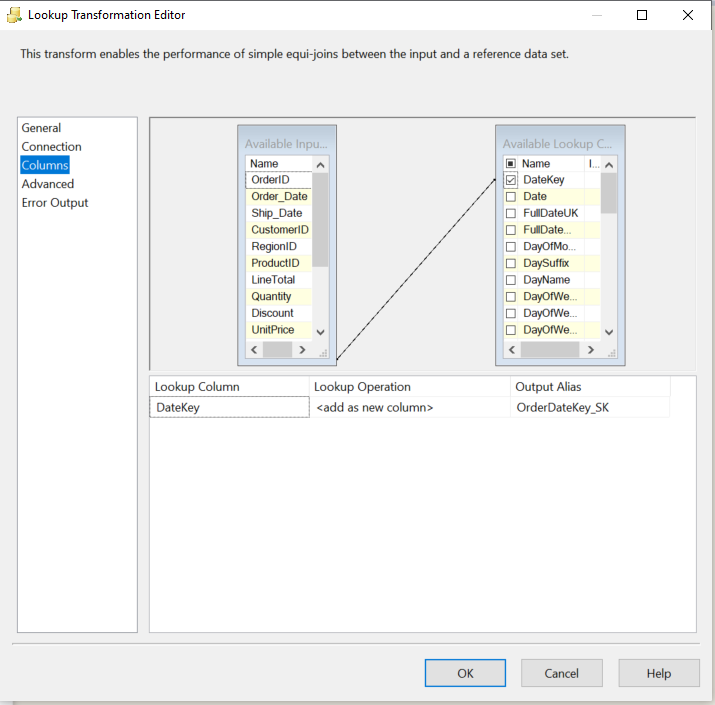
* Shipping\_Details\_Staging table and Order\_Staging table is merged in order to make the fact table. Shipping\_Details\_Staging is loaded and merged to obtain OrderID. All required surrogate keys has been loaded to data warehouse after a lookup through alternate keys in dimension tables.
* I Added a Derived Column and join the Merge Join item to Derived Column.
* After that I Added a Data Conversion item & link the Derived Column item to Data  
  Transformation item.



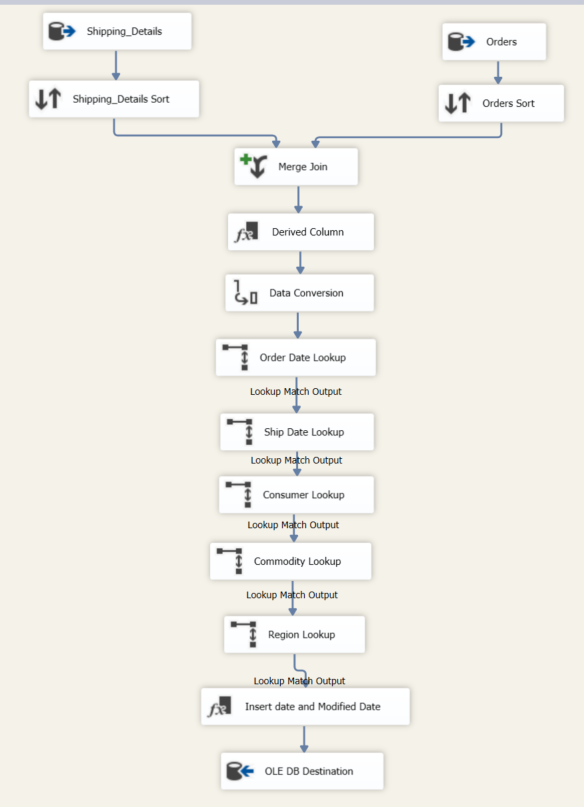
* Then I Converted dates into numeric format by using Data Conversion



* Then I Added a Lookup item and connect the Data Conversion item to it



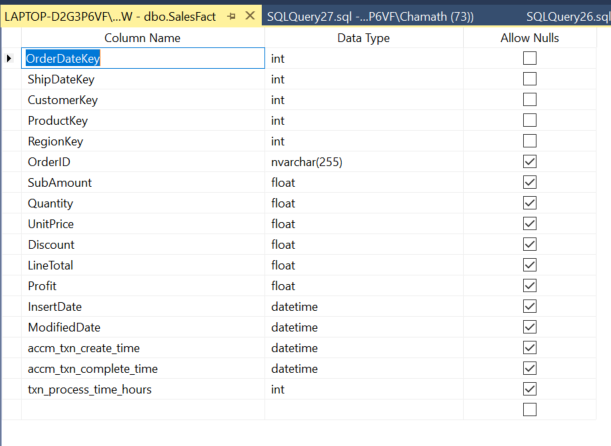
* This section of your data flow should look similar to below



**Step 06: ETL Development-Accumulating Fact Table**

Step 1:-

* I have extended SalesFact Table with 3 additional columns.



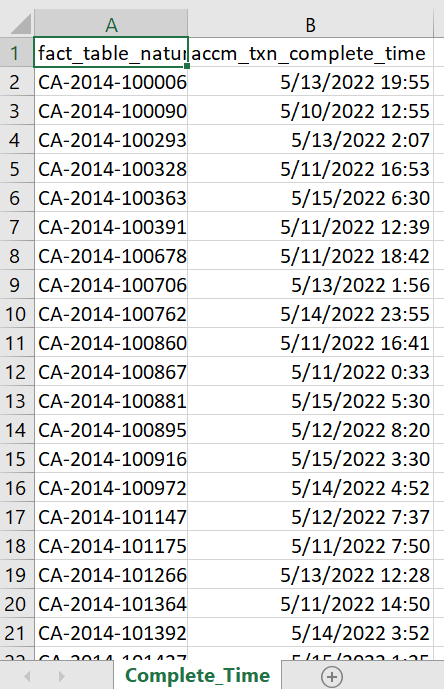
Step 2:-

* I had Set accm\_txn\_create\_time to be equal to the current system date when load data to Salesfact table.



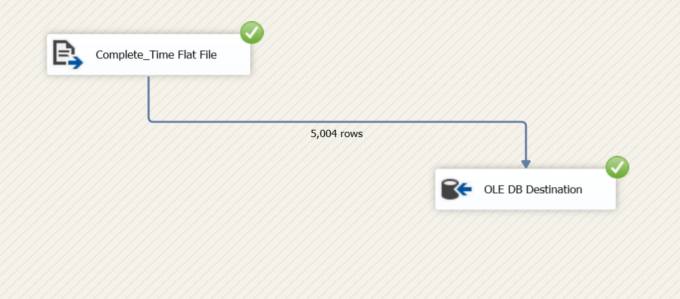
Step 3:-

* I had Prepared a separate data set called Complete\_Time.csv.

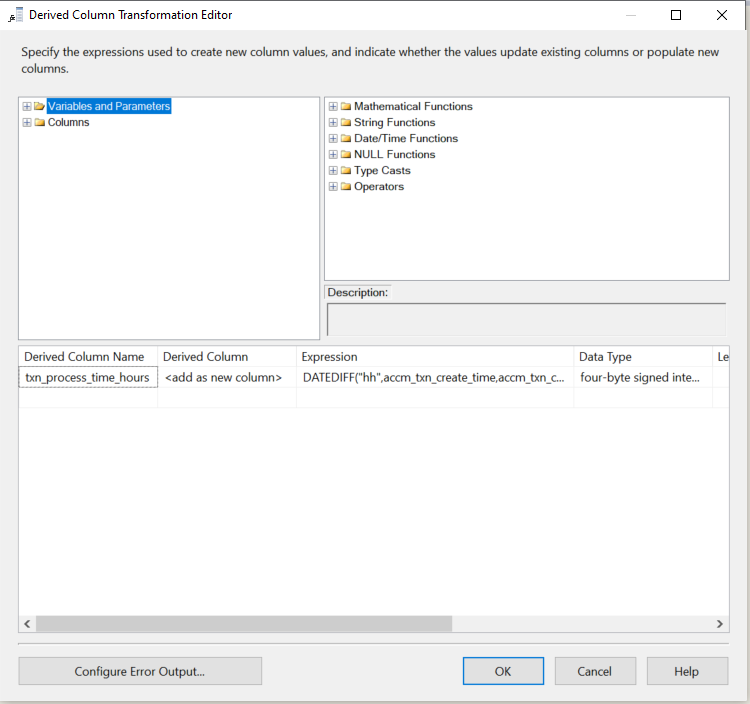


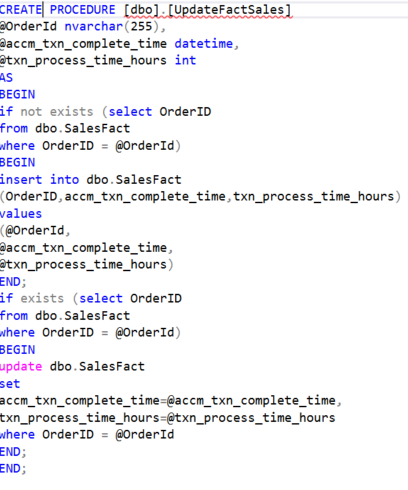
Step 4:-

* Designed Separate ETL package and load Complete\_Time.csv to Staging.

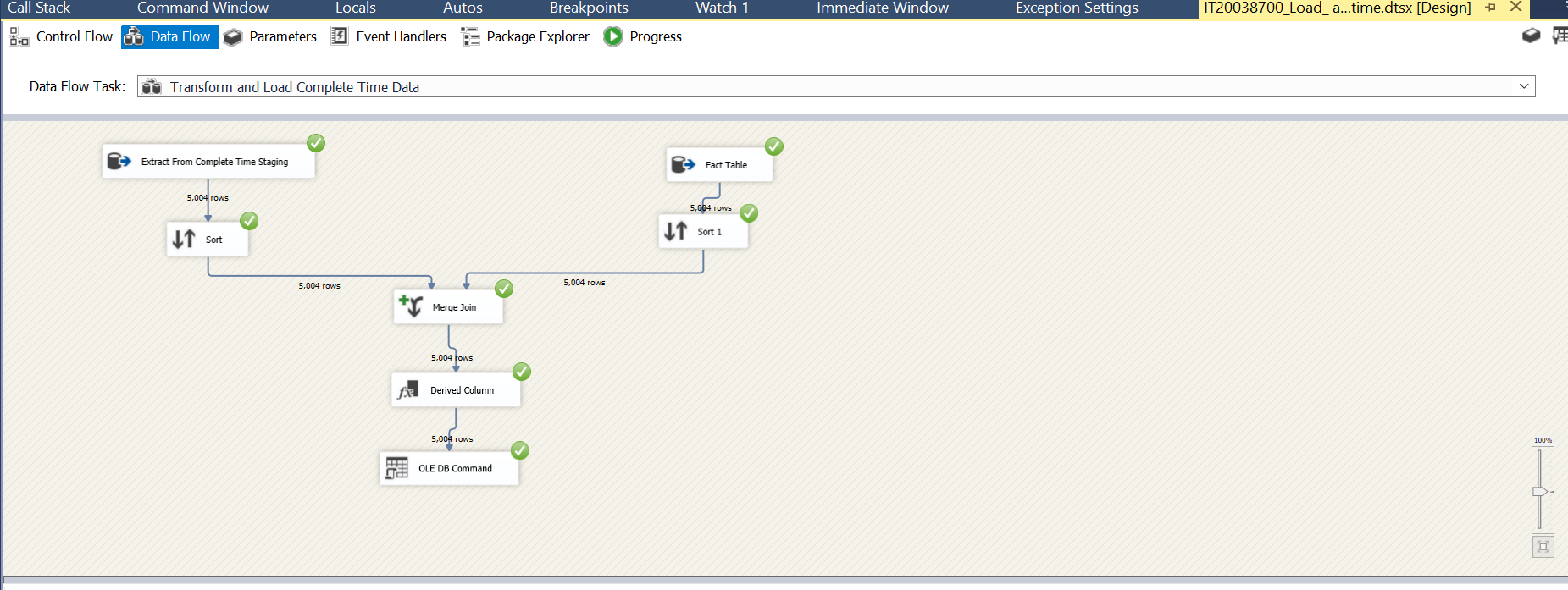


* Used Derived Column to Calculate Processing hours.

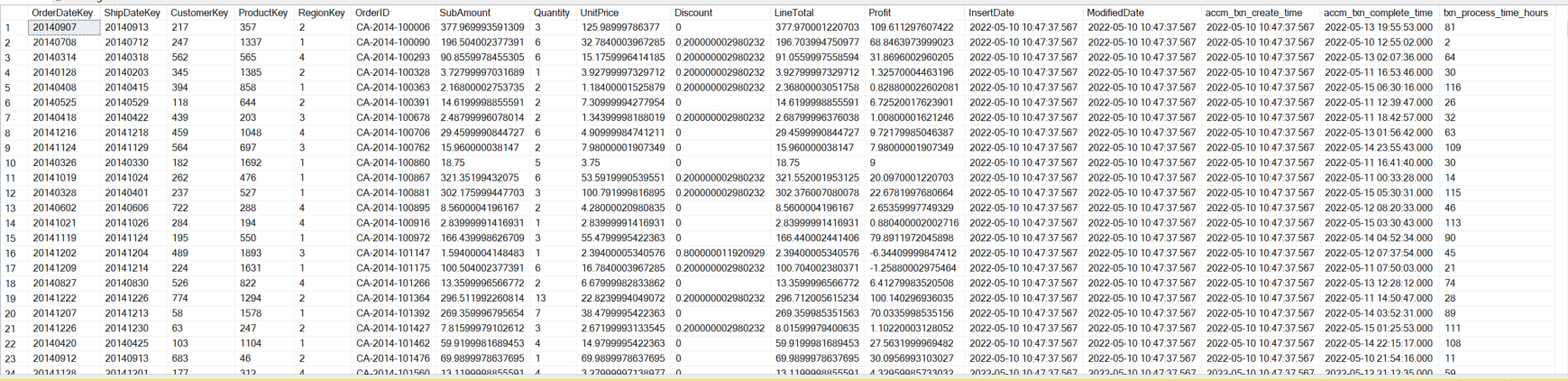




* Update the corresponding accm\_txn\_complete\_time in your DW fact table.



Step 5:-

* Updated SalesFact Table.

References

<https://www.guru99.com/utlimate-guide-etl-datawarehouse-testing.html>

<https://www.tutorialgateway.org/ssis-slowly-changing-dimension-type-2/>