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

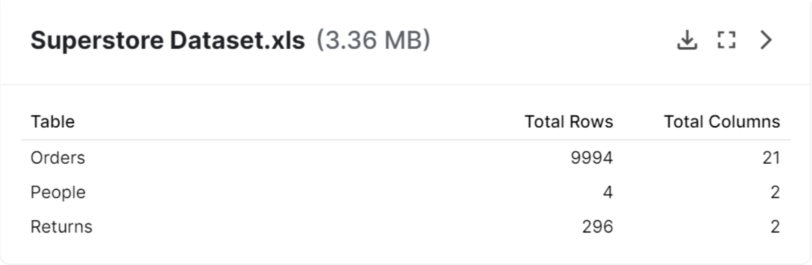
Data Set : Superstore Sales Data Set

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.

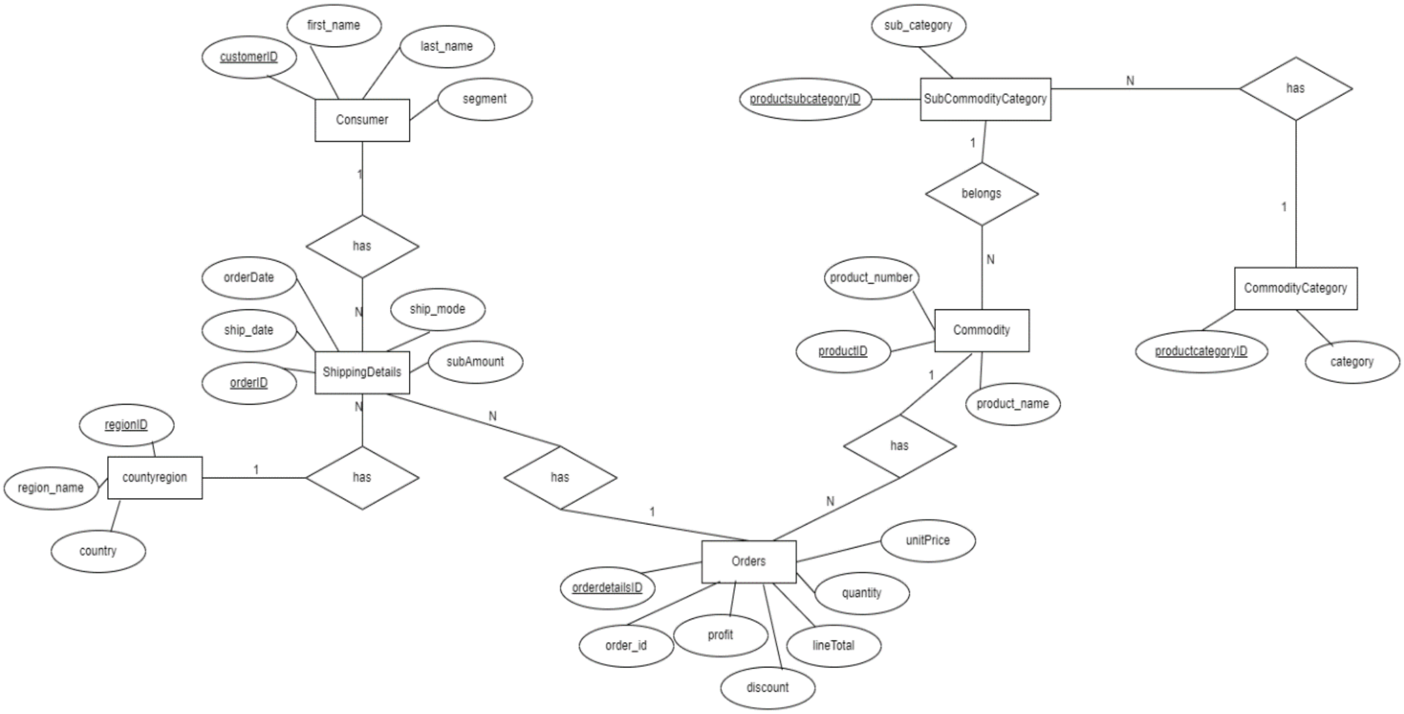
The Data Set was initiated with sufficient data, according to the assignment principles. The approval of choosing this data set was taken by the respected lecture.



Superstore Dataset.xls (3.36 MB)

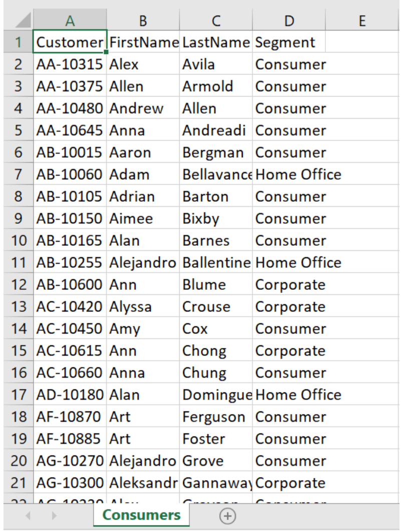
Table	Total Rows	Total Columns
Orders	9994	21
People	4	2
Returns	296	2

ER Diagram

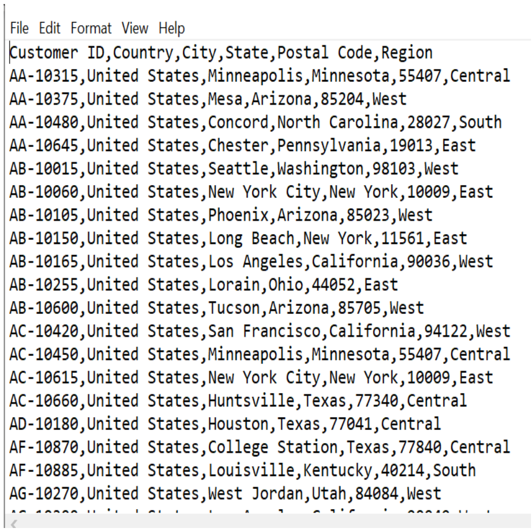


Step 02: Preparation of Data Sources

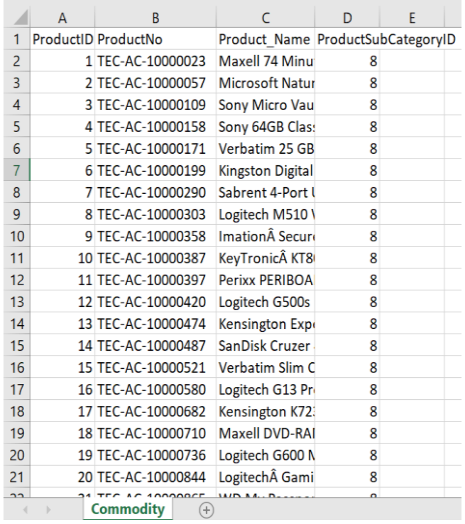
The implement data warehouse used data set view, it is Store Sale transaction data set, it partitioned into separate Source files to implement data warehouse, such as Consumers.csv, Commodity.csv, SubCommodityCategory.csv, CommodityCategory.csv, CountryRegion.csv, Orders.csv, ShippingDetails.csv, SalesConsumerRegion.csv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



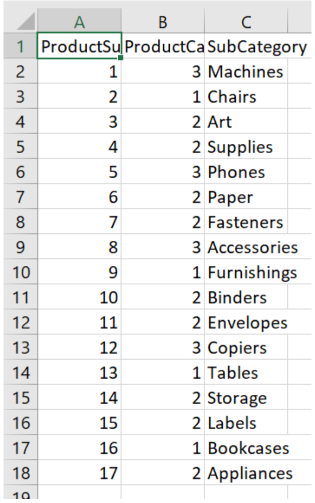
	A	B	C	D	E
1	Customer	FirstName	LastName	Segment	
2	AA-10315	Alex	Avila	Consumer	
3	AA-10375	Allen	Armold	Consumer	
4	AA-10480	Andrew	Allen	Consumer	
5	AA-10645	Anna	Andreadi	Consumer	
6	AB-10015	Aaron	Bergman	Consumer	
7	AB-10060	Adam	Bellavance	Home Office	
8	AB-10105	Adrian	Barton	Consumer	
9	AB-10150	Aimee	Bixby	Consumer	
10	AB-10165	Alan	Barnes	Consumer	
11	AB-10255	Alejandro	Ballentine	Home Office	
12	AB-10600	Ann	Blume	Corporate	
13	AC-10420	Alyssa	Crouse	Corporate	
14	AC-10450	Amy	Cox	Consumer	
15	AC-10615	Ann	Chong	Corporate	
16	AC-10660	Anna	Chung	Consumer	
17	AD-10180	Alan	Domingue	Home Office	
18	AF-10870	Art	Ferguson	Consumer	
19	AF-10885	Art	Foster	Consumer	
20	AG-10270	Alejandro	Grove	Consumer	
21	AG-10300	Aleksandr	Gannaway	Corporate	



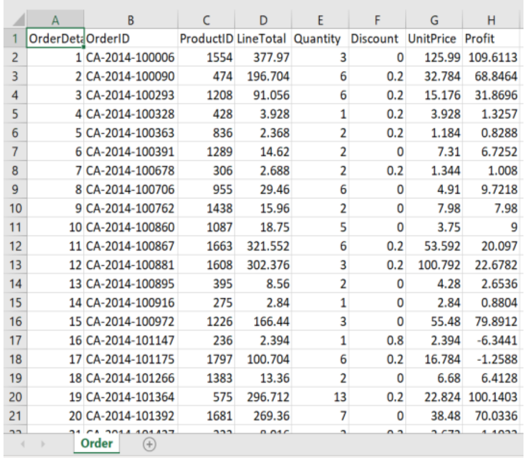
```
Customer ID,Country,City,State,Postal Code,Region
AA-10315,United States,Minneapolis,Minnesota,55407,Central
AA-10375,United States,Mesa,Arizona,85204,West
AA-10480,United States,Concord,North Carolina,28027,South
AA-10645,United States,Chester,Pennsylvania,19013,East
AB-10015,United States,Seattle,Washington,98103,West
AB-10060,United States,New York City,New York,10009,East
AB-10105,United States,Phoenix,Arizona,85023,West
AB-10150,United States,Long Beach,New York,11561,East
AB-10165,United States,Los Angeles,California,90036,West
AB-10255,United States,Lorain,Ohio,44052,East
AB-10600,United States,Tucson,Arizona,85705,West
AC-10420,United States,San Francisco,California,94122,West
AC-10450,United States,Minneapolis,Minnesota,55407,Central
AC-10615,United States,New York City,New York,10009,East
AC-10660,United States,Huntsville,Texas,77340,Central
AD-10180,United States,Houston,Texas,77041,Central
AF-10870,United States,College Station,Texas,77840,Central
AF-10885,United States,Louisville,Kentucky,40214,South
AG-10270,United States,West Jordan,Utah,84084,West
```



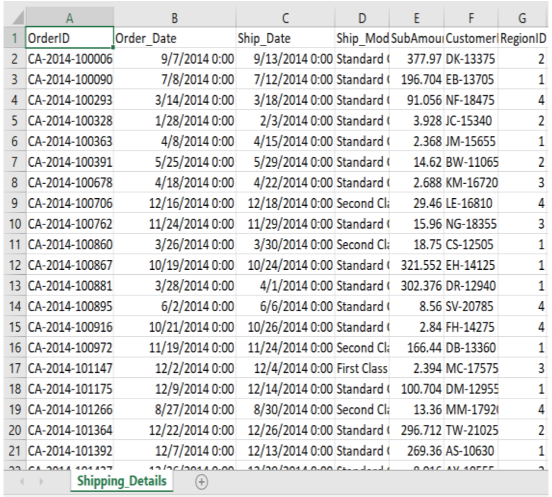
	A	B	C	D	E
1	ProductID	ProductNo	Product_Name	ProductSubCategoryID	
2	1	TEC-AC-10000023	Maxell 74 Minu	8	
3	2	TEC-AC-10000057	Microsoft Natur	8	
4	3	TEC-AC-10000109	Sony Micro Vau	8	
5	4	TEC-AC-10000158	Sony 64GB Clas	8	
6	5	TEC-AC-10000171	Verbatim 25 GB	8	
7	6	TEC-AC-10000199	Kingston Digital	8	
8	7	TEC-AC-10000290	Sabrent 4-Port	8	
9	8	TEC-AC-10000303	Logitech M510	8	
10	9	TEC-AC-10000358	ImationÂ Secur	8	
11	10	TEC-AC-10000387	KeyTronicÂ KT8	8	
12	11	TEC-AC-10000397	Perixx PERIBOA	8	
13	12	TEC-AC-10000420	Logitech G500s	8	
14	13	TEC-AC-10000474	Kensington Exp	8	
15	14	TEC-AC-10000487	SanDisk Cruzer	8	
16	15	TEC-AC-10000521	Verbatim Slim C	8	
17	16	TEC-AC-10000580	Logitech G13 Pr	8	
18	17	TEC-AC-10000682	Kensington K72	8	
19	18	TEC-AC-10000710	Maxell DVD-RAI	8	
20	19	TEC-AC-10000736	Logitech G600 N	8	
21	20	TEC-AC-10000844	LogitechÂ Gami	8	



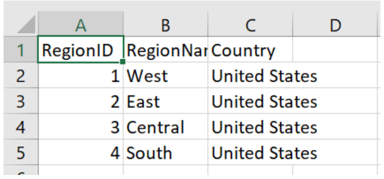
	A	B	C
1	ProductSu	ProductCa	SubCategory
2	1	3	Machines
3	2	1	Chairs
4	3	2	Art
5	4	2	Supplies
6	5	3	Phones
7	6	2	Paper
8	7	2	Fasteners
9	8	3	Accessories
10	9	1	Furnishings
11	10	2	Binders
12	11	2	Envelopes
13	12	3	Copiers
14	13	1	Tables
15	14	2	Storage
16	15	2	Labels
17	16	1	Bookcases
18	17	2	Appliances



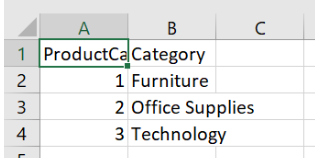
	A	B	C	D	E	F	G	H
1	OrderDeta	OrderID	ProductID	LineTotal	Quantity	Discount	UnitPrice	Profit
2	1	CA-2014-100006	1554	377.97	3	0	125.99	109.6113
3	2	CA-2014-100090	474	196.704	6	0.2	32.784	68.8464
4	3	CA-2014-100293	1208	91.056	6	0.2	15.176	31.8696
5	4	CA-2014-100328	428	3.928	1	0.2	3.928	1.3257
6	5	CA-2014-100363	836	2.368	2	0.2	1.184	0.8288
7	6	CA-2014-100391	1289	14.62	2	0	7.31	6.7252
8	7	CA-2014-100678	306	2.688	2	0.2	1.344	1.008
9	8	CA-2014-100706	955	29.46	6	0	4.91	9.7218
10	9	CA-2014-100762	1438	15.96	2	0	7.98	7.98
11	10	CA-2014-100860	1087	18.75	5	0	3.75	9
12	11	CA-2014-100867	1663	321.552	6	0.2	53.592	20.097
13	12	CA-2014-100881	1608	302.376	3	0.2	100.792	22.6782
14	13	CA-2014-100895	395	8.56	2	0	4.28	2.6536
15	14	CA-2014-100916	275	2.84	1	0	2.84	0.8804
16	15	CA-2014-100972	1226	166.44	3	0	55.48	79.8912
17	16	CA-2014-101147	236	2.394	1	0.8	2.394	-6.3441
18	17	CA-2014-101175	1797	100.704	6	0.2	16.784	-1.2588
19	18	CA-2014-101266	1383	13.36	2	0	6.68	6.4128
20	19	CA-2014-101364	575	296.712	13	0.2	22.824	100.1403
21	20	CA-2014-101392	1681	269.36	7	0	38.48	70.0336



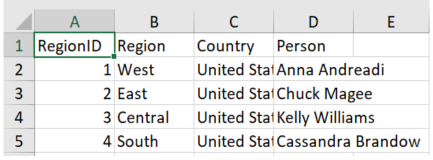
	A	B	C	D	E	F	G
1	OrderID	Order_Date	Ship_Date	Ship_Mod	SubAmou	Customer	RegionID
2	CA-2014-100006	9/7/2014 0:00	9/13/2014 0:00	Standard (	377.97	DK-13375	2
3	CA-2014-100090	7/8/2014 0:00	7/12/2014 0:00	Standard (	196.704	EB-13705	1
4	CA-2014-100293	3/14/2014 0:00	3/18/2014 0:00	Standard (	91.056	NF-18475	4
5	CA-2014-100328	1/28/2014 0:00	2/3/2014 0:00	Standard (	3.928	JC-15340	2
6	CA-2014-100363	4/8/2014 0:00	4/15/2014 0:00	Standard (	2.368	JM-15655	1
7	CA-2014-100391	5/25/2014 0:00	5/29/2014 0:00	Standard (	14.62	BW-11065	2
8	CA-2014-100678	4/18/2014 0:00	4/22/2014 0:00	Standard (	2.688	KM-16720	3
9	CA-2014-100706	12/16/2014 0:00	12/18/2014 0:00	Second Cl	29.46	LE-16810	4
10	CA-2014-100762	11/24/2014 0:00	11/29/2014 0:00	Standard (	15.96	NG-18355	3
11	CA-2014-100860	3/26/2014 0:00	3/30/2014 0:00	Second Cl	18.75	CS-12505	1
12	CA-2014-100867	10/19/2014 0:00	10/24/2014 0:00	Standard (	321.552	EH-14125	1
13	CA-2014-100881	3/28/2014 0:00	4/1/2014 0:00	Standard (	302.376	DR-12940	1
14	CA-2014-100895	6/2/2014 0:00	6/6/2014 0:00	Standard (	8.56	SV-20785	4
15	CA-2014-100916	10/21/2014 0:00	10/26/2014 0:00	Standard (	2.84	FH-14275	4
16	CA-2014-100972	11/19/2014 0:00	11/24/2014 0:00	Second Cl	166.44	DB-13360	1
17	CA-2014-101147	12/2/2014 0:00	12/4/2014 0:00	First Class	2.394	MC-17575	3
18	CA-2014-101175	12/9/2014 0:00	12/14/2014 0:00	Standard (	100.704	DM-1295	1
19	CA-2014-101266	8/27/2014 0:00	8/30/2014 0:00	Second Cl	13.36	MM-1792	4
20	CA-2014-101364	12/22/2014 0:00	12/26/2014 0:00	Standard (	296.712	TW-21025	2
21	CA-2014-101392	12/7/2014 0:00	12/13/2014 0:00	Standard (	269.36	AS-10630	1



	A	B	C	D
1	RegionID	RegionNar	Country	
2	1	West	United States	
3	2	East	United States	
4	3	Central	United States	
5	4	South	United States	

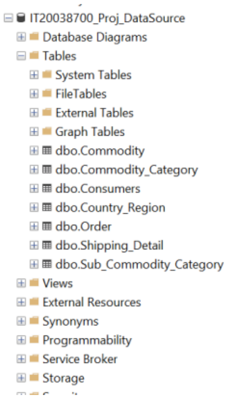


	A	B	C
1	ProductCa	Category	
2	1	Furniture	
3	2	Office Supplies	
4	3	Technology	



	A	B	C	D	E
1	RegionID	Region	Country	Person	
2	1	West	United Stat	Anna Andreadi	
3	2	East	United Stat	Chuck Magee	
4	3	Central	United Stat	Kelly Williams	
5	4	South	United Stat	Cassandra Brandow	

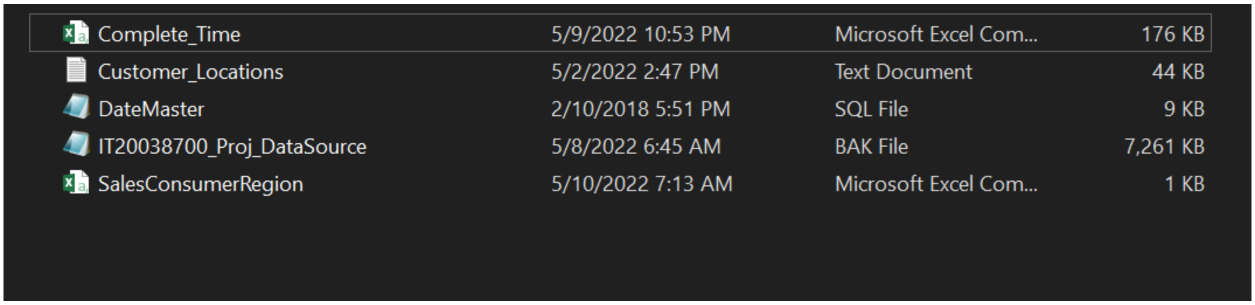
I have Loaded Commodity.csv, CommodityCategory.csv, Consumers.csv, CountryRegion.csv, Orders.csv, ShippingDetails.csv and SubCommodityCategory.csv files to the DB Called IT20038700_Proj_DataSource.



```
⊟ IT20038700_Proj_DataSource
  ⊞ Database Diagrams
  ⊟ Tables
    ⊞ System Tables
    ⊞ FileTables
    ⊞ External Tables
    ⊞ Graph Tables
    ⊞ dbo.Commodity
    ⊞ dbo.Commodity_Category
    ⊞ dbo.Consumers
    ⊞ dbo.Country_Region
    ⊞ dbo.Order
    ⊞ dbo.Shipping_Detail
    ⊞ dbo.Sub_Commodity_Category
  ⊞ Views
  ⊞ External Resources
  ⊞ Synonyms
  ⊞ Programmability
  ⊞ Service Broker
  ⊞ Storage
```

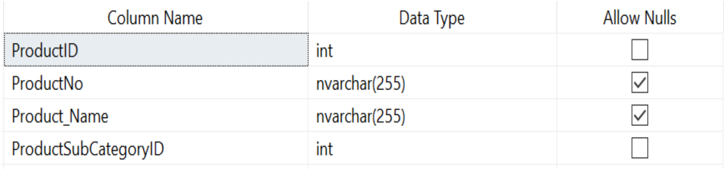
I have converted ConsumerLocation.csv to Text file so that I can use Flat Source in SSIS tool, to extract ConsumerLocation.txt data.
I have used separate data file called SalesConsumerRegion.csv file. It can use Flat Source in SSIS tool, to extract data.

Final Set of Sources

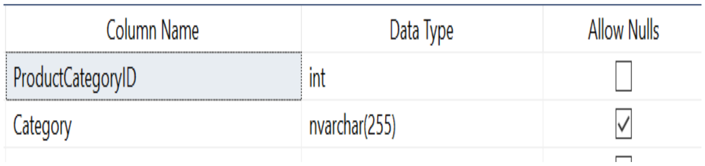


Complete_Time	5/9/2022 10:53 PM	Microsoft Excel Com...	176 KB
Customer_Locations	5/2/2022 2:47 PM	Text Document	44 KB
DateMaster	2/10/2018 5:51 PM	SQL File	9 KB
IT20038700_Proj_DataSource	5/8/2022 6:45 AM	BAK File	7,261 KB
SalesConsumerRegion	5/10/2022 7:13 AM	Microsoft Excel Com...	1 KB

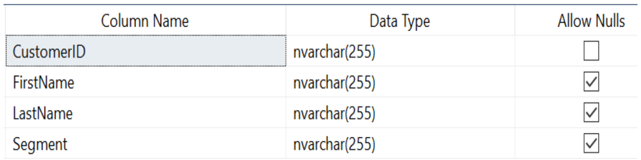
Describe Sources



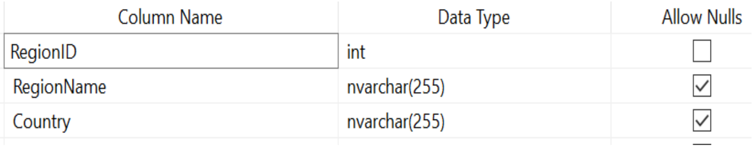
Column Name	Data Type	Allow Nulls
ProductID	int	☐
ProductNo	nvarchar(255)	☑
Product_Name	nvarchar(255)	☑
ProductSubCategoryID	int	☐



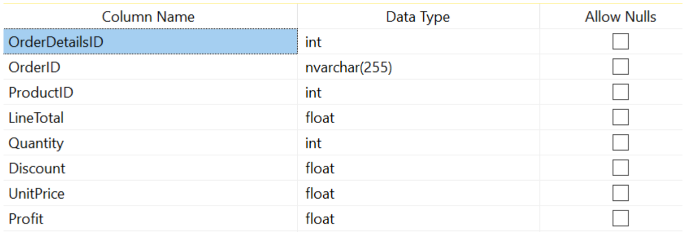
Column Name	Data Type	Allow Nulls
ProductCategoryID	int	☐
Category	nvarchar(255)	☑



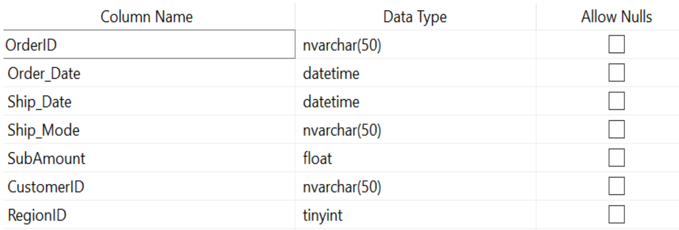
Column Name	Data Type	Allow Nulls
CustomerID	nvarchar(255)	☐
FirstName	nvarchar(255)	☑
LastName	nvarchar(255)	☑
Segment	nvarchar(255)	☑



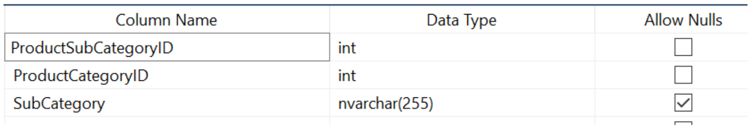
Column Name	Data Type	Allow Nulls
RegionID	int	☐
RegionName	nvarchar(255)	☑
Country	nvarchar(255)	☑



Column Name	Data Type	Allow Nulls
OrderDetailsID	int	☐
OrderID	nvarchar(255)	☐
ProductID	int	☐
LineTotal	float	☐
Quantity	int	☐
Discount	float	☐
UnitPrice	float	☐
Profit	float	☐

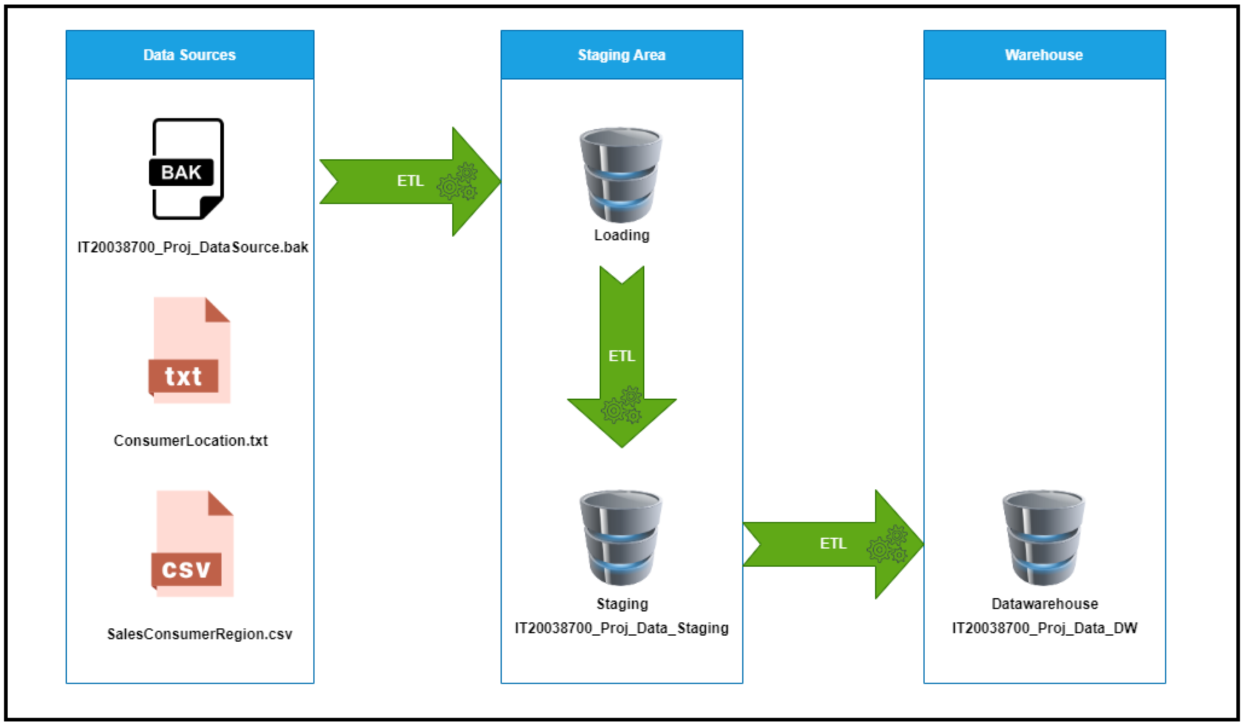


Column Name	Data Type	Allow Nulls
OrderID	nvarchar(50)	☐
Order_Date	datetime	☐
Ship_Date	datetime	☐
Ship_Mode	nvarchar(50)	☐
SubAmount	float	☐
CustomerID	nvarchar(50)	☐
RegionID	tinyint	☐



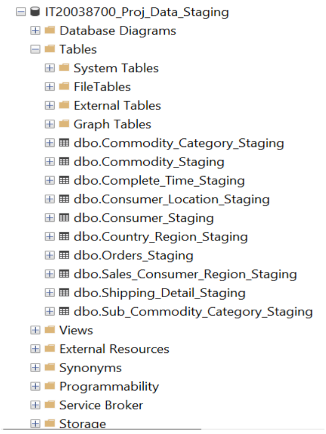
Column Name	Data Type	Allow Nulls
ProductSubCategoryID	int	☐
ProductCategoryID	int	☐
SubCategory	nvarchar(255)	☑

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.

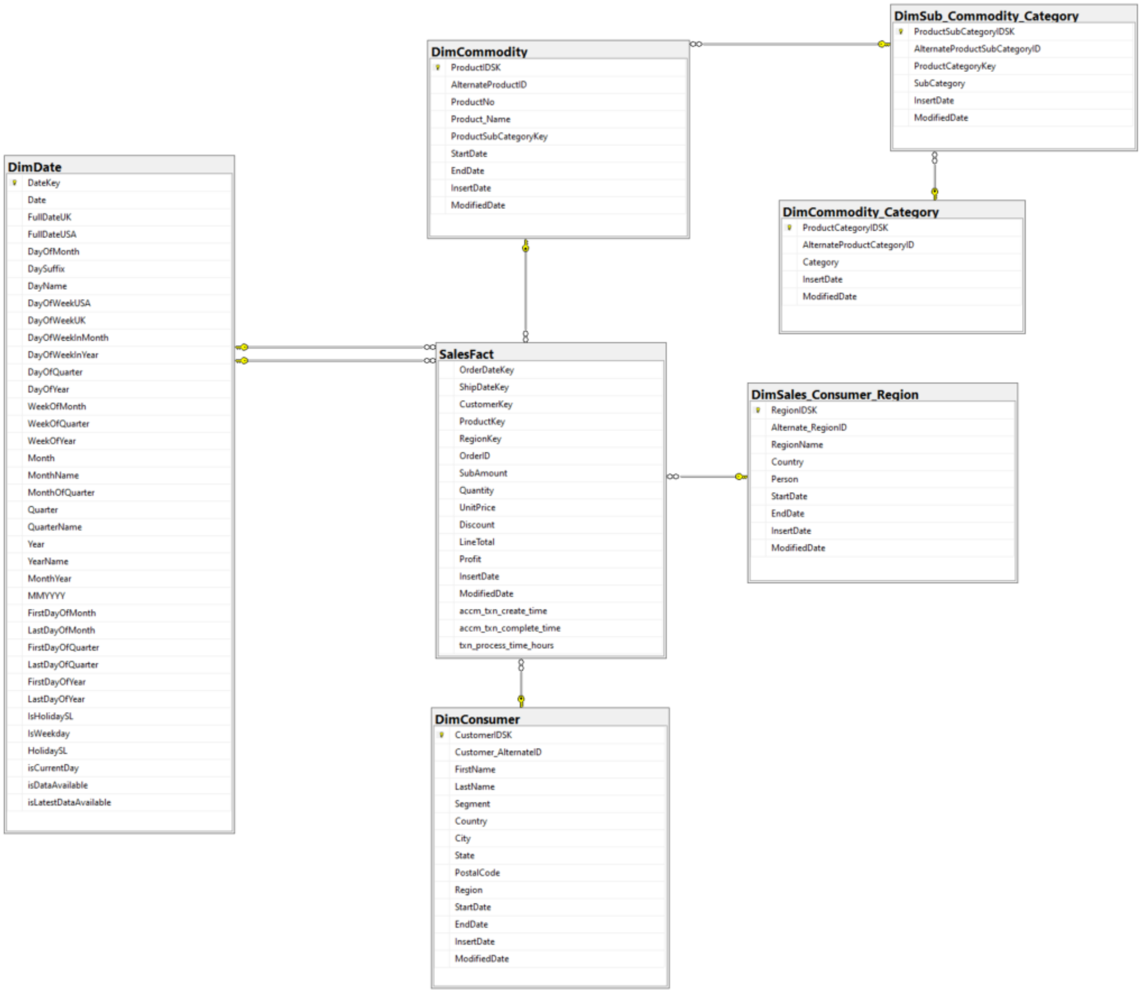
- The tables at the staging are then profiled and after performing a rich set of ETL tasks, data is loaded to the data warehouse where from that several reporting tools and analysing tools can use data for reporting mining and analysing.



- IT20038700_Proj_Data_Staging
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
    - dbo.Commodity_Category_Staging
    - dbo.Commodity_Staging
    - dbo.Complete_Time_Staging
    - dbo.Consumer_Location_Staging
    - dbo.Consumer_Staging
    - dbo.Country_Region_Staging
    - dbo.Orders_Staging
    - dbo.Sales_Consumer_Region_Staging
    - dbo.Shipping_Detail_Staging
    - dbo.Sub_Commodity_Category_Staging
  - Views
  - External Resources
  - Synonyms
  - Programmability
  - Service Broker
  - Storage

Step 04: Data Warehouse Design & Development

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



Assumption

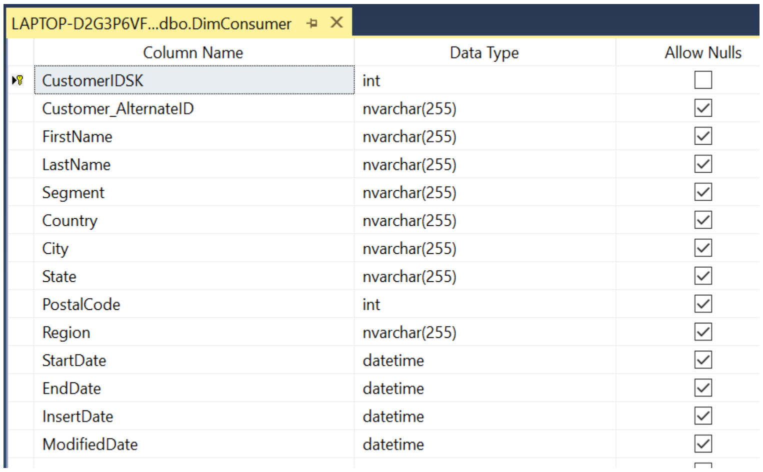
- I have taken DimConsumer as slowly changing dimension, Consumer addresses can change time to time, and we need to keep track of their historical address.
- In the data set product name is too large So assume that if Store owner wants to change the product name and we need to keep track of their historical 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)

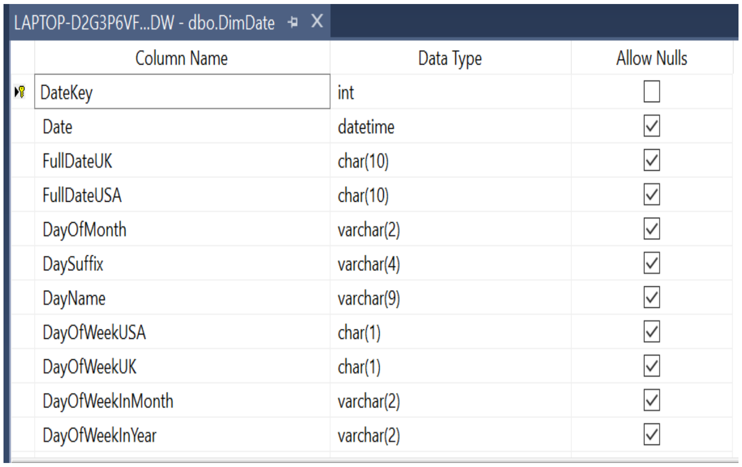
Before creating the sales fact table & Other dimensions, start by creating the Date Dimension. For that I use the code in the file ‘DateMaster.sql’ file.

Dimension Tables



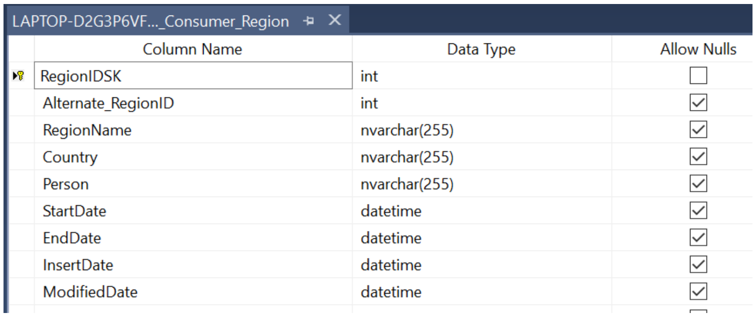
LAPTOP-D2G3P6VF...dbo.DimConsumer

Column Name	Data Type	Allow Nulls
CustomerIDSK	int	☐
Customer_AlternateID	nvarchar(255)	☑
FirstName	nvarchar(255)	☑
LastName	nvarchar(255)	☑
Segment	nvarchar(255)	☑
Country	nvarchar(255)	☑
City	nvarchar(255)	☑
State	nvarchar(255)	☑
PostalCode	int	☑
Region	nvarchar(255)	☑
StartDate	datetime	☑
EndDate	datetime	☑
InsertDate	datetime	☑
ModifiedDate	datetime	☑



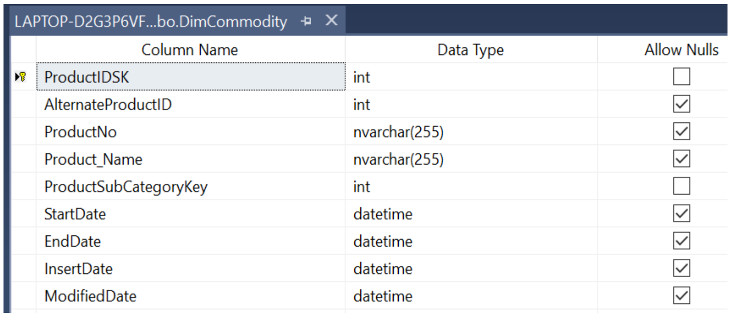
LAPTOP-D2G3P6VF...DW - dbo.DimDate

Column Name	Data Type	Allow Nulls
DateKey	int	☐
Date	datetime	☑
FullDateUK	char(10)	☑
FullDateUSA	char(10)	☑
DayOfMonth	varchar(2)	☑
DaySuffix	varchar(4)	☑
DayName	varchar(9)	☑
DayOfWeekUSA	char(1)	☑
DayOfWeekUK	char(1)	☑
DayOfWeekInMonth	varchar(2)	☑
DayOfWeekInYear	varchar(2)	☑



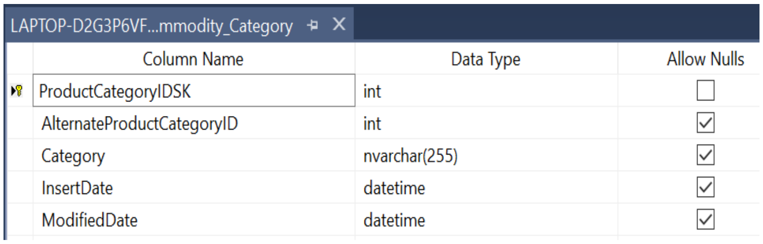
LAPTOP-D2G3P6VF..._Consumer_Region

Column Name	Data Type	Allow Nulls
RegionIDSK	int	☐
Alternate_RegionID	int	☑
RegionName	nvarchar(255)	☑
Country	nvarchar(255)	☑
Person	nvarchar(255)	☑
StartDate	datetime	☑
EndDate	datetime	☑
InsertDate	datetime	☑
ModifiedDate	datetime	☑



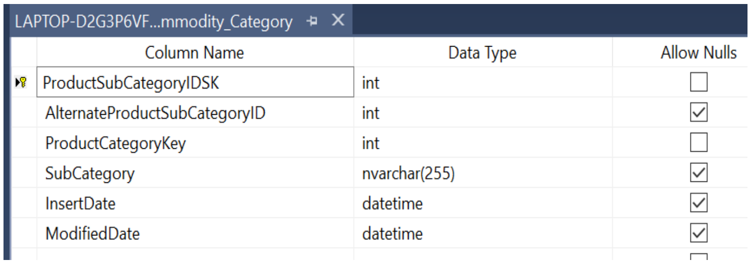
LAPTOP-D2G3P6VF...bo.DimCommodity

Column Name	Data Type	Allow Nulls
ProductIDSK	int	☐
AlternateProductID	int	☑
ProductNo	nvarchar(255)	☑
Product_Name	nvarchar(255)	☑
ProductSubCategoryKey	int	☐
StartDate	datetime	☑
EndDate	datetime	☑
InsertDate	datetime	☑
ModifiedDate	datetime	☑



LAPTOP-D2G3P6VF...mmodity_Category

Column Name	Data Type	Allow Nulls
ProductCategoryIDSK	int	☐
AlternateProductCategoryID	int	☑
Category	nvarchar(255)	☑
InsertDate	datetime	☑
ModifiedDate	datetime	☑

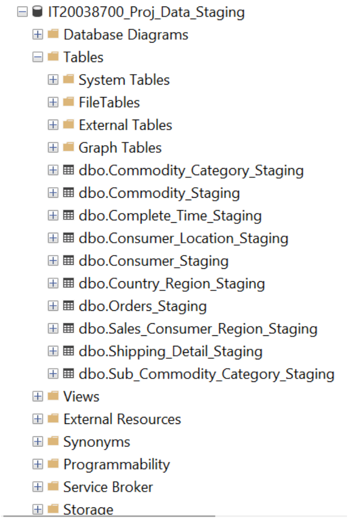


LAPTOP-D2G3P6VF...mmodity_Category

Column Name	Data Type	Allow Nulls
ProductSubCategoryIDSK	int	☐
AlternateProductSubCategoryID	int	☑
ProductCategoryKey	int	☐
SubCategory	nvarchar(255)	☑
InsertDate	datetime	☑
ModifiedDate	datetime	☑

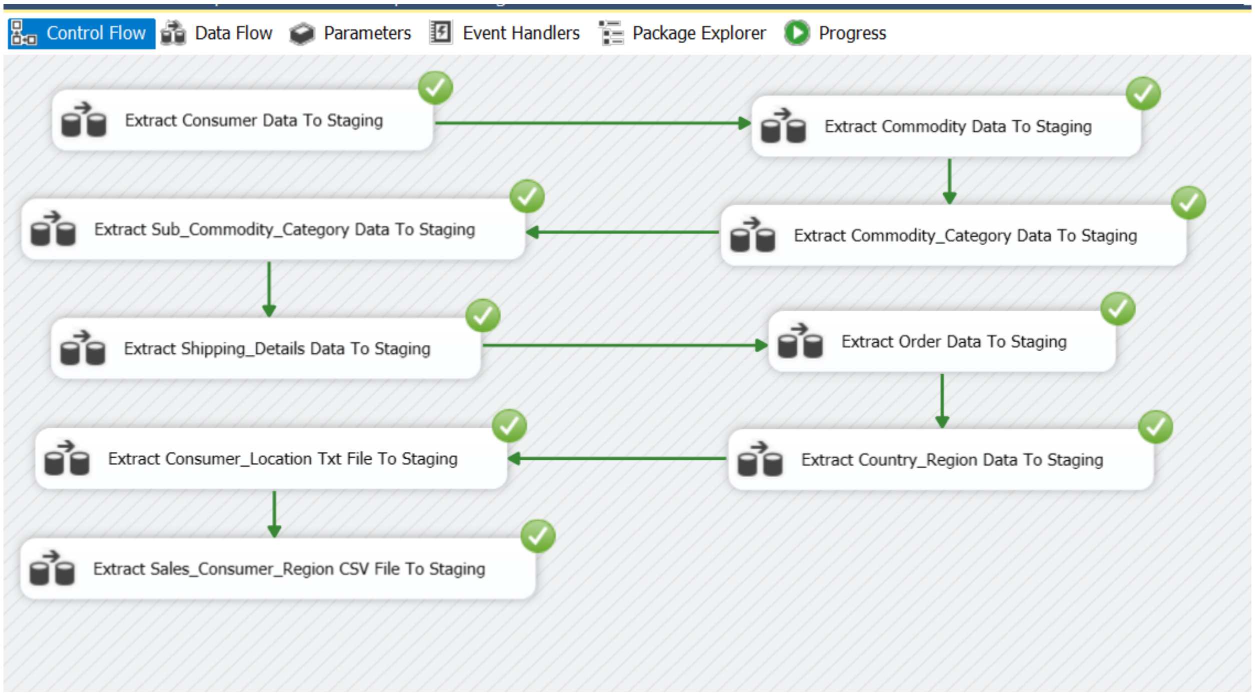
Step 05: ETL Development

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 .



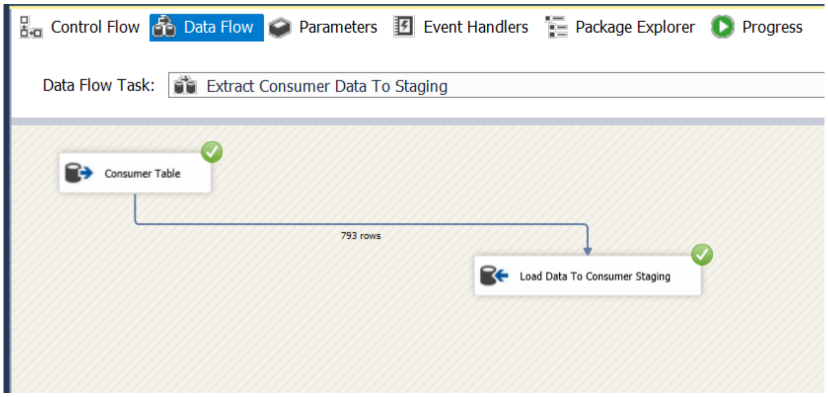
IT20038700_Proj_Data_Staging
- Database Diagrams
- Tables
  - System Tables
  - FileTables
  - External Tables
  - Graph Tables
  - dbo.Commodity_Category_Staging
  - dbo.Commodity_Staging
  - dbo.Complete_Time_Staging
  - dbo.Consumer_Location_Staging
  - dbo.Consumer_Staging
  - dbo.Country_Region_Staging
  - dbo.Orders_Staging
  - dbo.Sales_Consumer_Region_Staging
  - dbo.Shipping_Detail_Staging
  - dbo.Sub_Commodity_Category_Staging
- Views
- External Resources
- Synonyms
- Programmability
- Service Broker
- Storage

Overall Control Flow


Control Flow
Data Flow
Parameters
Event Handlers
Package Explorer
Progress
Extract Consumer Data To Staging
Extract Commodity Data To Staging
Extract Sub_Commodity_Category Data To Staging
Extract Commodity_Category Data To Staging
Extract Shipping_Details Data To Staging
Extract Order Data To Staging
Extract Consumer_Location Txt File To Staging
Extract Country_Region Data To Staging
Extract Sales_Consumer_Region CSV File To Staging

Data Extraction

1. Consumer Data from Source to Staging

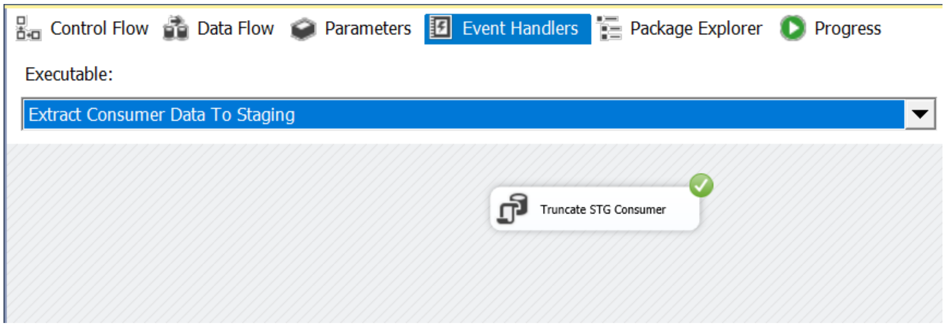


Data Flow Task: Extract Consumer Data To Staging

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

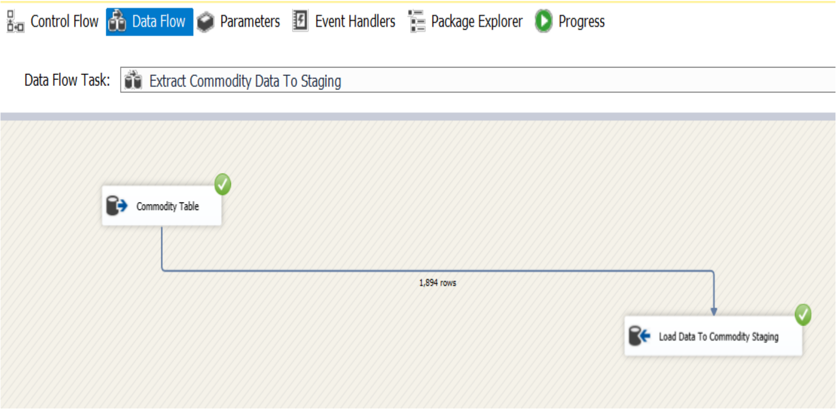
Event Handler

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



Executable: Extract Consumer Data To Staging

2. Commodity Data from Source to Staging

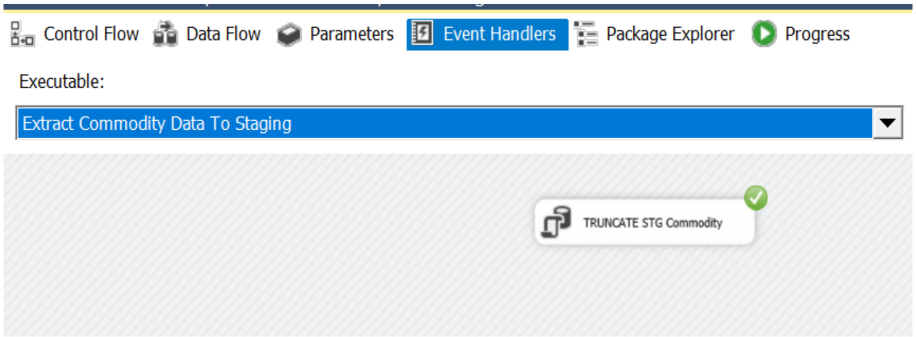


Data Flow Task: Extract Commodity Data To Staging

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

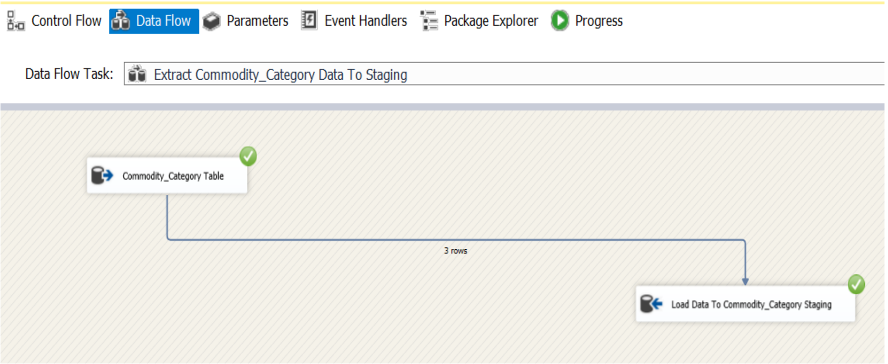
Event Handler

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



Executable: Extract Commodity Data To Staging

3. Commodity_Category Data from Source to Staging

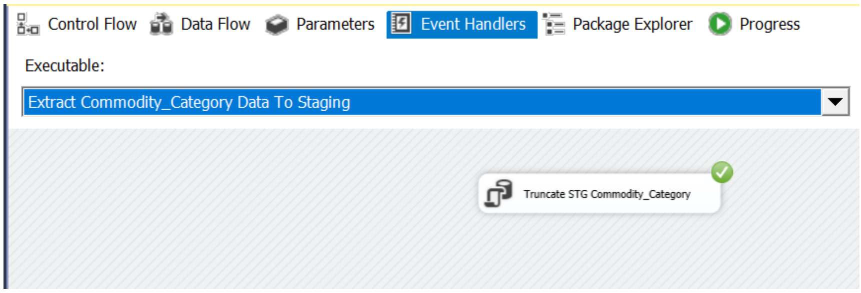


Data Flow Task: Extract Commodity_Category Data To Staging

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

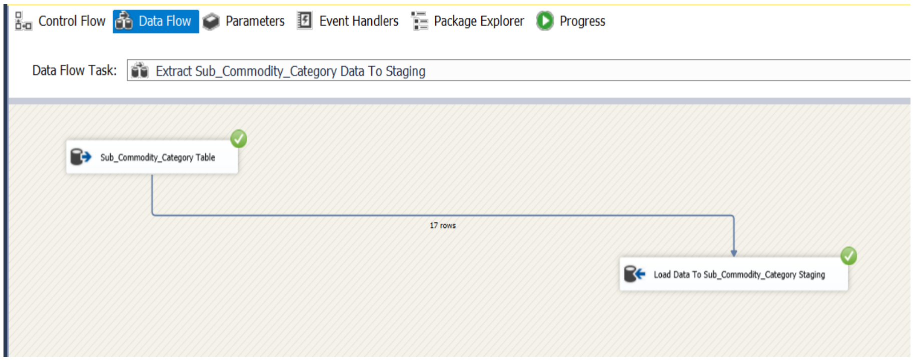
Event Handler

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



Executable: Extract Commodity_Category Data To Staging

4. Sub_Commodity_Category Data from Source to Staging

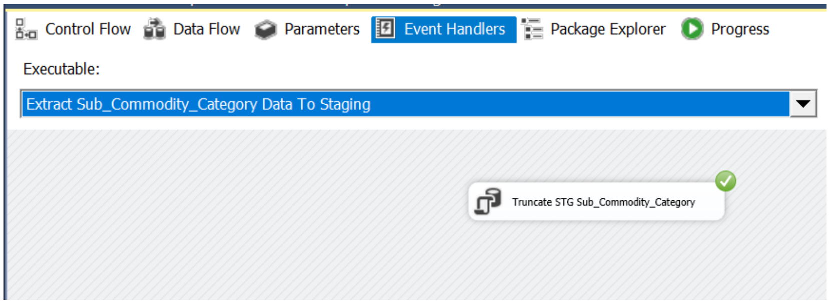


Data Flow Task: Extract Sub_Commodity_Category Data To Staging

Sub Commodity Category Data in Sub_Commodity_Category Database Table has been extracted and loaded to Sub_Commodity_Category_Staging table

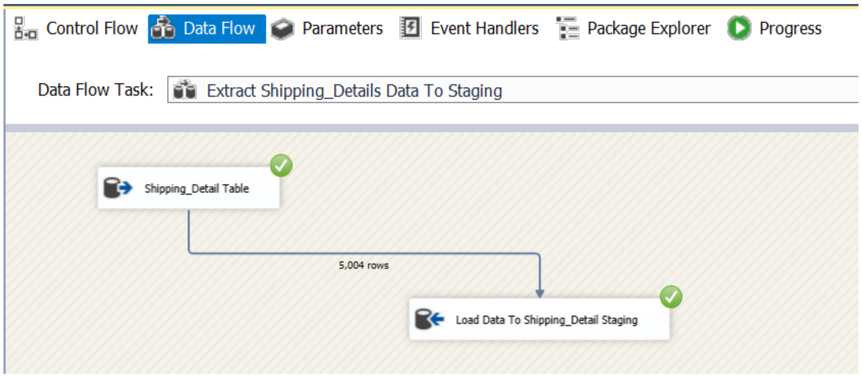
Event Handler

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



Executable: Extract Sub_Commodity_Category Data To Staging

5. Shipping_Details Data from Source to Staging

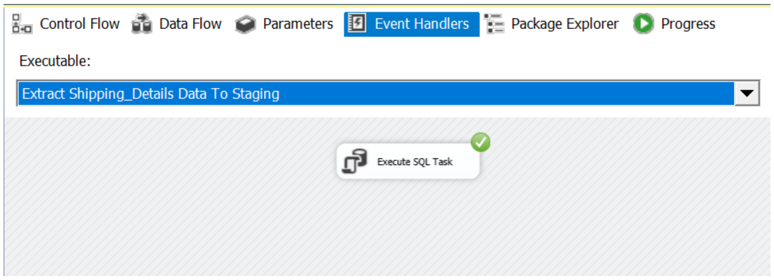


Data Flow Task: Extract Shipping_Details Data To Staging

Shipping Detail Data in Shipping_Details Database Table has been extracted and loaded to Shipping_Details_Staging table

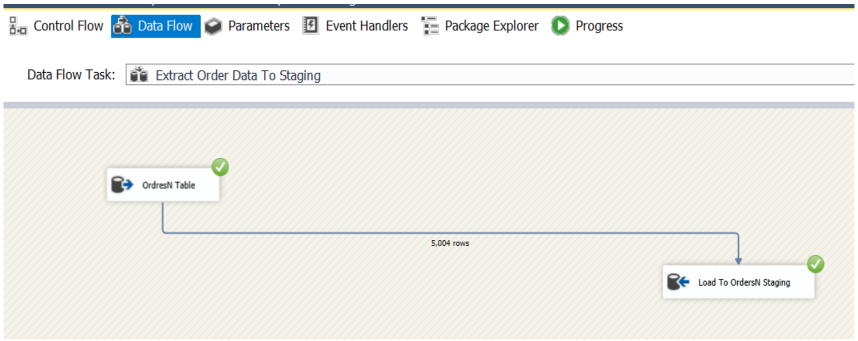
Event Handler

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



Executable: Extract Shipping_Details Data To Staging

6. Order Data from Source to Staging

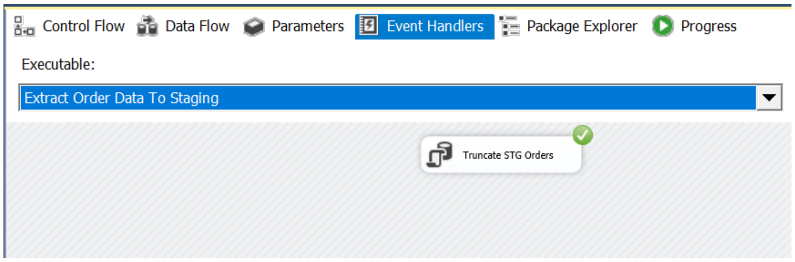


Data Flow Task: Extract Order Data To Staging

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

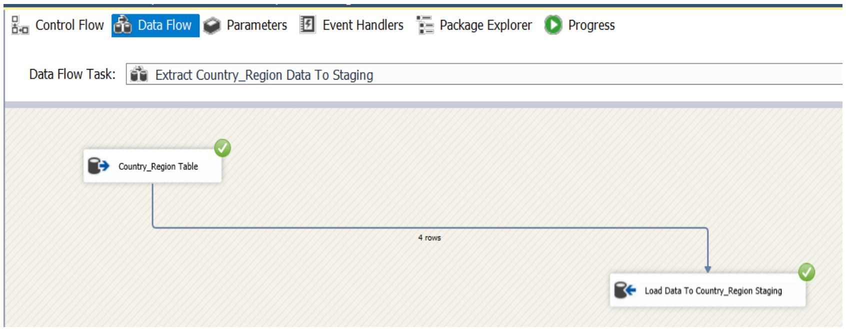
Event Handler

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



Executable: Extract Order Data To Staging

7. Country_Region Data from Source to Staging

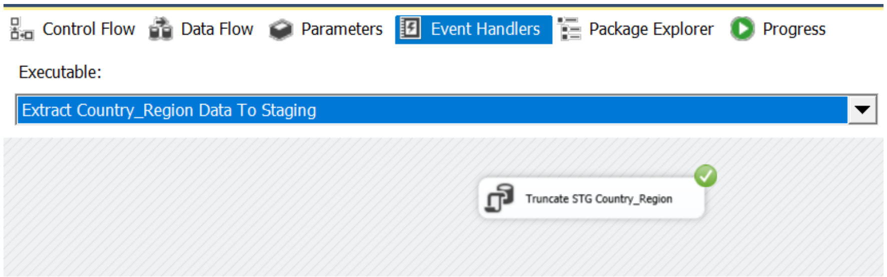


Data Flow Task: Extract Country_Region Data To Staging

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

Event Handler

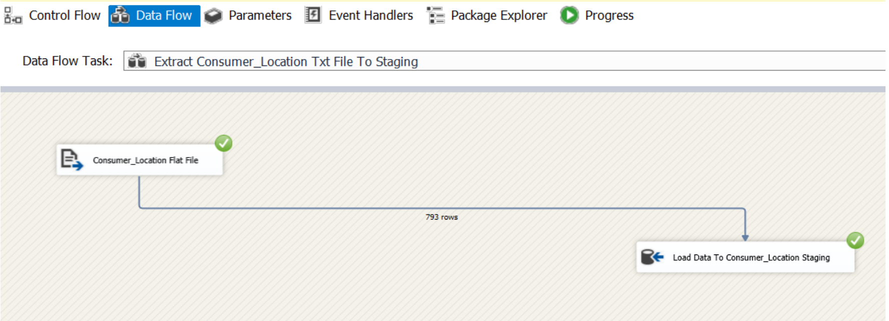
Before executing ‘extract Country Region Data to staging’ existing data in the staging layer has been truncated.



Executable: Extract Country_Region Data To Staging

8. Consumer_Location Data Text File to Staging

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



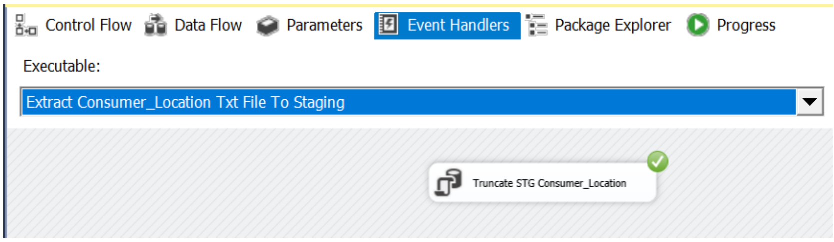
Data Flow Task: Extract Consumer_Location Txt File To Staging

Consumer_Location Flat File
793 rows
Load Data To Consumer_Location Staging

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

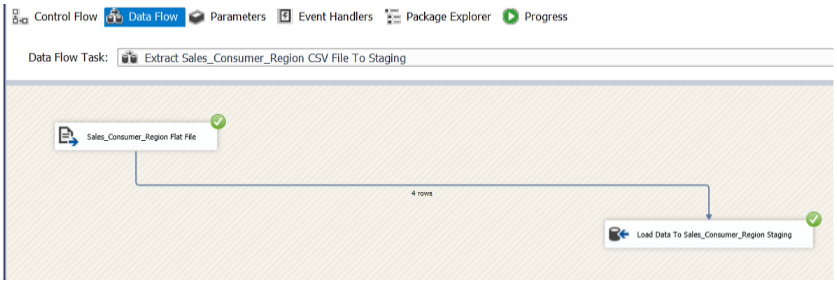
Event Handler

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



Executable: Extract Consumer_Location Txt File To Staging

9. Sales_Consumer_Region Data CSV File to Staging



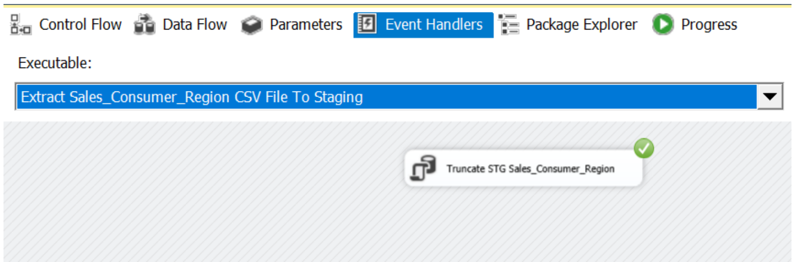
Data Flow Task: Extract Sales_Consumer_Region CSV File To Staging

Sales_Consumer_Region Flat File
4 rows
Load Data To Sales_Consumer_Region Staging

Sales Consumer Regions Data in Sales_Consumer_Regions .csv has been extracted and loaded to Sales_Consumer_Regions _Staging table

Event Handler

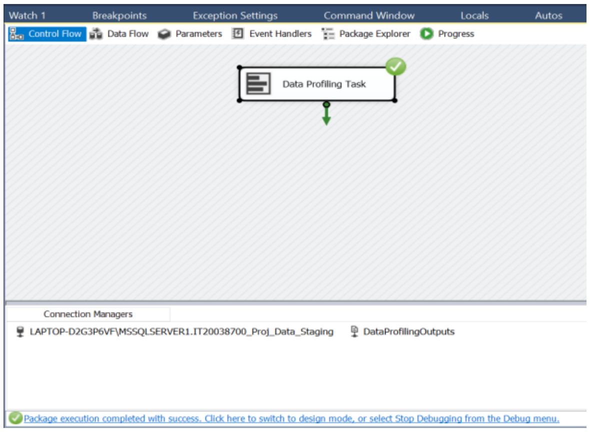
Before executing ‘extract Sales Consumer Regions Data to staging’ existing data in the staging layer has been truncated.



Executable: Extract Sales_Consumer_Region CSV File To Staging

Data Profiling

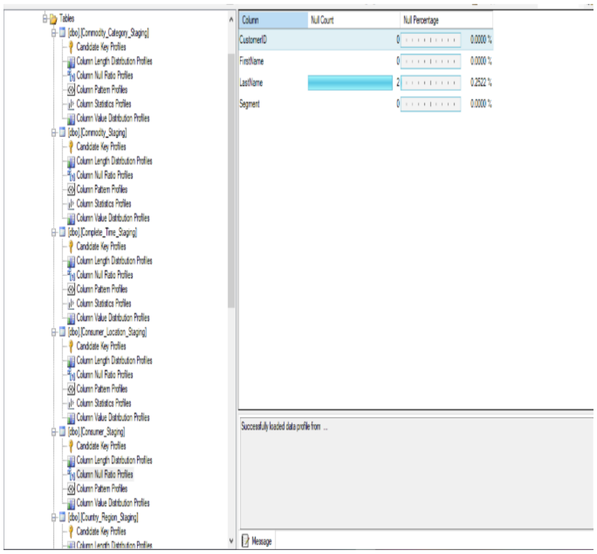
Before Loading staging tables to the data warehouse data has to be enriched to obtain the most suitable data for analyzing. Data profiling has been done in order to identify what need to be corrected in ETL process in order to meet this requirement.



LAPTOP-D2G3P6VF\MSSQLSERVER1.IT20038700_Proj_Data_Staging

DataProfilingOutputs

Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.


Tables
[dbo].[Commodity_Category_Staging]
Candidate Key Profiles
Column Length Distribution Profiles
Column Null Ratio Profiles
Column Pattern Profiles
Column Statistics Profiles
Column Value Distribution Profiles
[dbo].[Commodity_Staging]
Candidate Key Profiles
Column Length Distribution Profiles
Column Null Ratio Profiles
Column Pattern Profiles
Column Statistics Profiles
Column Value Distribution Profiles
[dbo].[Complete_Time_Staging]
Candidate Key Profiles
Column Length Distribution Profiles
Column Null Ratio Profiles
Column Pattern Profiles
Column Statistics Profiles
Column Value Distribution Profiles
[dbo].[Consumer_Location_Staging]
Candidate Key Profiles
Column Length Distribution Profiles
Column Null Ratio Profiles
Column Pattern Profiles
Column Statistics Profiles
Column Value Distribution Profiles
[dbo].[Consumer_Staging]
Candidate Key Profiles
Column Length Distribution Profiles
Column Null Ratio Profiles
Column Pattern Profiles
Column Value Distribution Profiles
[dbo].[Country_Region_Staging]
Candidate Key Profiles
Column Length Distribution Profiles
Message

Column	Null Count	Null Percentage
CustomerID	0	0.0000 %
FirstName	0	0.0000 %
LastName	2	0.2522 %
Segment	0	0.0000 %

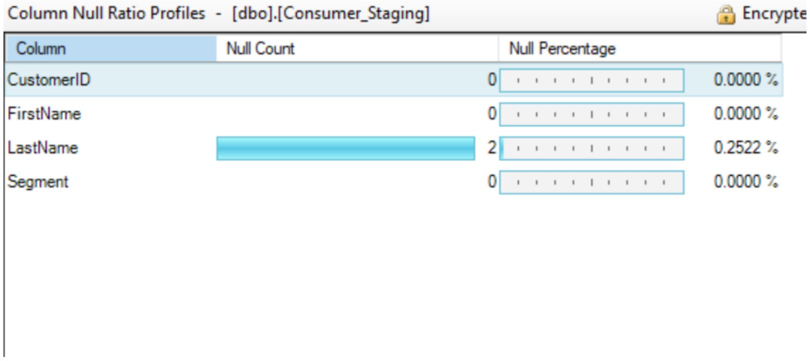
Successfully loaded data profile from ...

Data Transforming and Loading

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

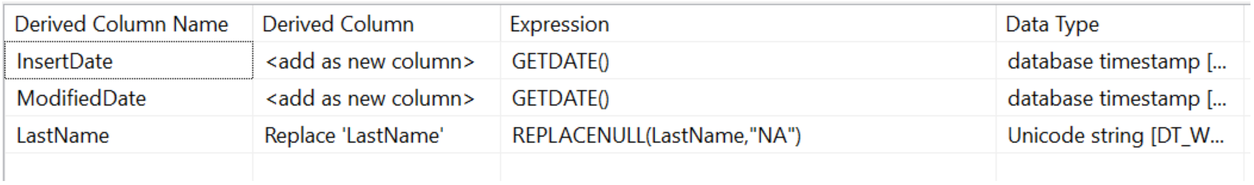
- Consumer table in the staging has been merged into Customer_Location table since there are only few rows in Customer_Location table. Merge has been performed by sorting both the tables using the common filed ‘CustomerID’.

- 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.


Column Null Ratio Profiles - [dbo].[Consumer_Staging]

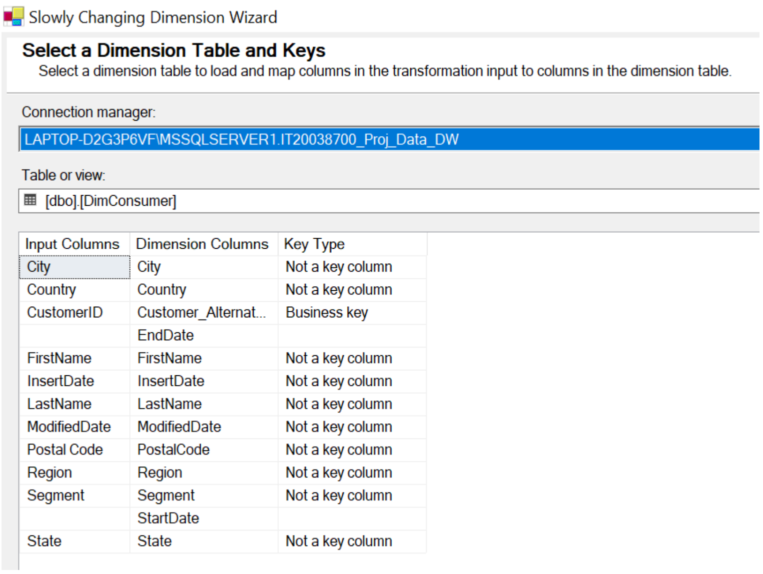
Column	Null Count	Null Percentage
CustomerID	0	0.0000 %
FirstName	0	0.0000 %
LastName	2	0.2522 %
Segment	0	0.0000 %

- Then I Replace the Replace null by using the Derived Column



Derived Column Name	Derived Column	Expression	Data Type
InsertDate	<add as new column>	GETDATE()	database timestamp [...
ModifiedDate	<add as new column>	GETDATE()	database timestamp [...
LastName	Replace 'LastName'	REPLACENULL(LastName,"NA")	Unicode string [DT_W...

- DimConsumer dimension has been identified as a slowly changing dimension. Hence necessary steps have been followed to make DimConsumer a slowly changing dimension.


Slowly Changing Dimension Wizard

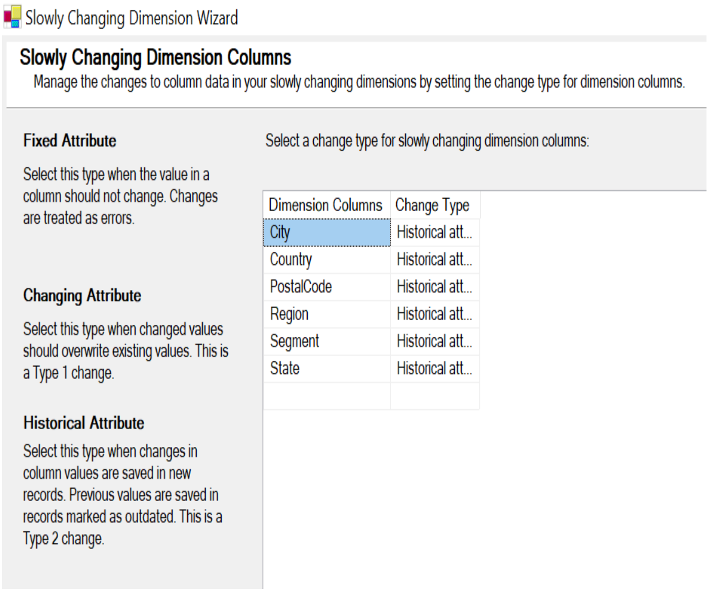
Select a Dimension Table and Keys

Select a dimension table to load and map columns in the transformation input to columns in the dimension table.

Connection manager:
LAPTOP-D2G3P6VF\MSSQLSERVER1.IT20038700_Proj_Data_DW

Table or view:
[dbo].[DimConsumer]

Input Columns	Dimension Columns	Key Type
City	City	Not a key column
Country	Country	Not a key column
CustomerID	Customer_Alternat...	Business key
	EndDate	
FirstName	FirstName	Not a key column
InsertDate	InsertDate	Not a key column
LastName	LastName	Not a key column
ModifiedDate	ModifiedDate	Not a key column
Postal Code	PostalCode	Not a key column
Region	Region	Not a key column
Segment	Segment	Not a key column
	StartDate	
State	State	Not a key column


Slowly Changing Dimension Wizard

Slowly Changing Dimension Columns

Manage the changes to column data in your slowly changing dimensions by setting the change type for dimension columns.

Fixed Attribute

Select this type when the value in a column should not change. Changes are treated as errors.

Changing Attribute

Select this type when changed values should overwrite existing values. This is a Type 1 change.

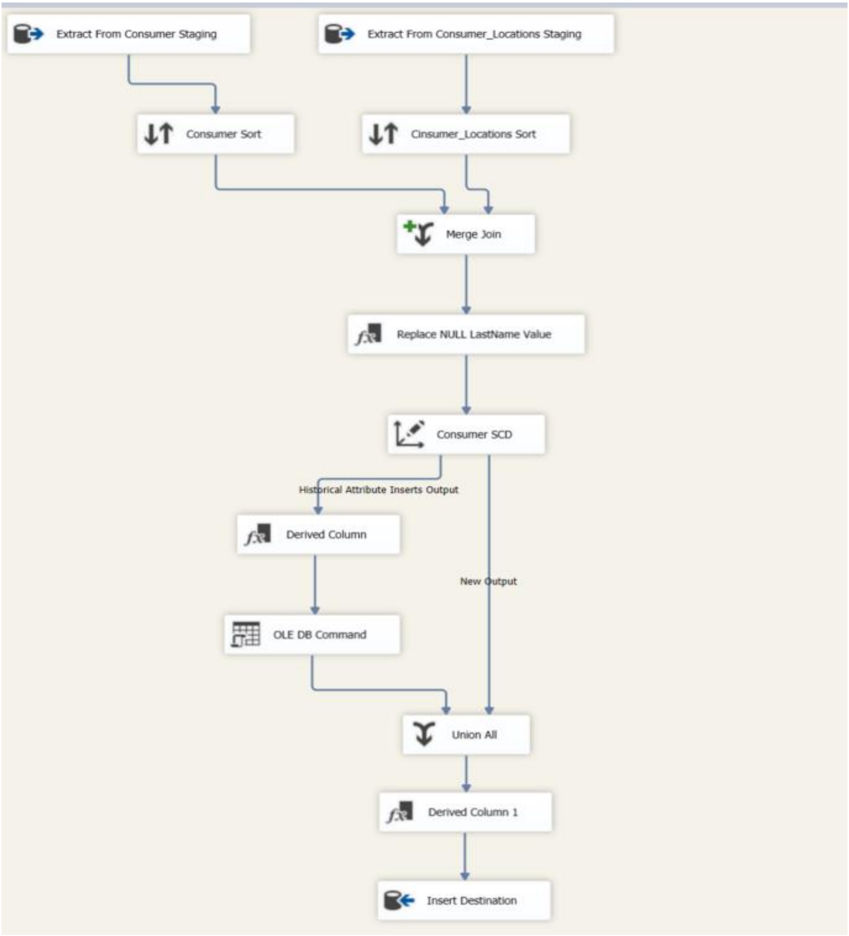
Historical Attribute

Select this type when changes in column values are saved in new records. Previous values are saved in records marked as outdated. This is a Type 2 change.

Select a change type for slowly changing dimension columns:

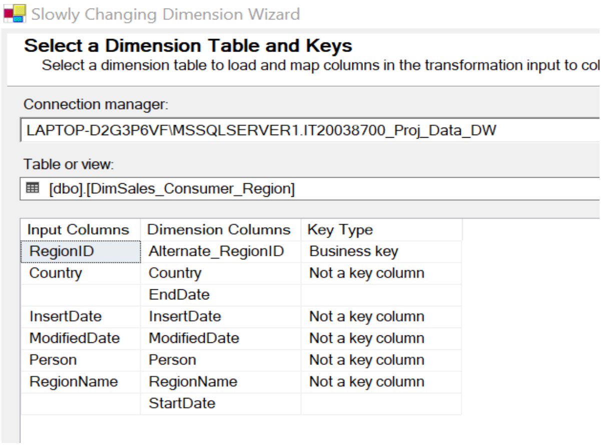
Dimension Columns	Change Type
City	Historical att...
Country	Historical att...
PostalCode	Historical att...
Region	Historical att...
Segment	Historical att...
State	Historical att...

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



2) 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.



Slowly Changing Dimension Wizard

Select a Dimension Table and Keys

Select a dimension table to load and map columns in the transformation input to col

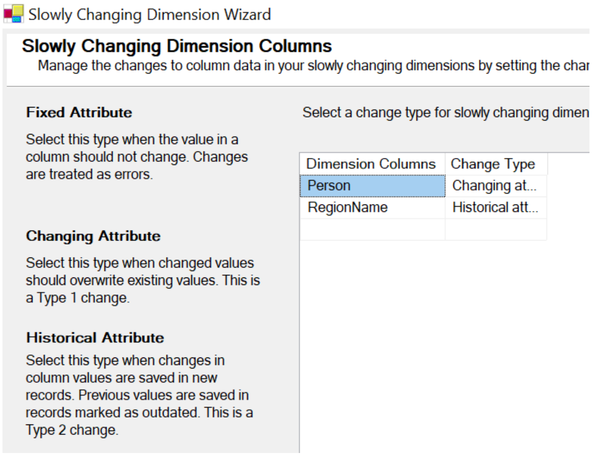
Connection manager:

LAPTOP-D2G3P6VF\MSSQLSERVER1.IT20038700_Proj_Data_DW

Table or view:

[dbo].[DimSales_Consumer_Region]

Input Columns	Dimension Columns	Key Type
RegionID	Alternate_RegionID	Business key
Country	Country	Not a key column
	EndDate	
InsertDate	InsertDate	Not a key column
ModifiedDate	ModifiedDate	Not a key column
Person	Person	Not a key column
RegionName	RegionName	Not a key column
	StartDate	



Slowly Changing Dimension Wizard

Slowly Changing Dimension Columns

Manage the changes to column data in your slowly changing dimensions by setting the char

Fixed Attribute

Select this type when the value in a column should not change. Changes are treated as errors.

Changing Attribute

Select this type when changed values should overwrite existing values. This is a Type 1 change.

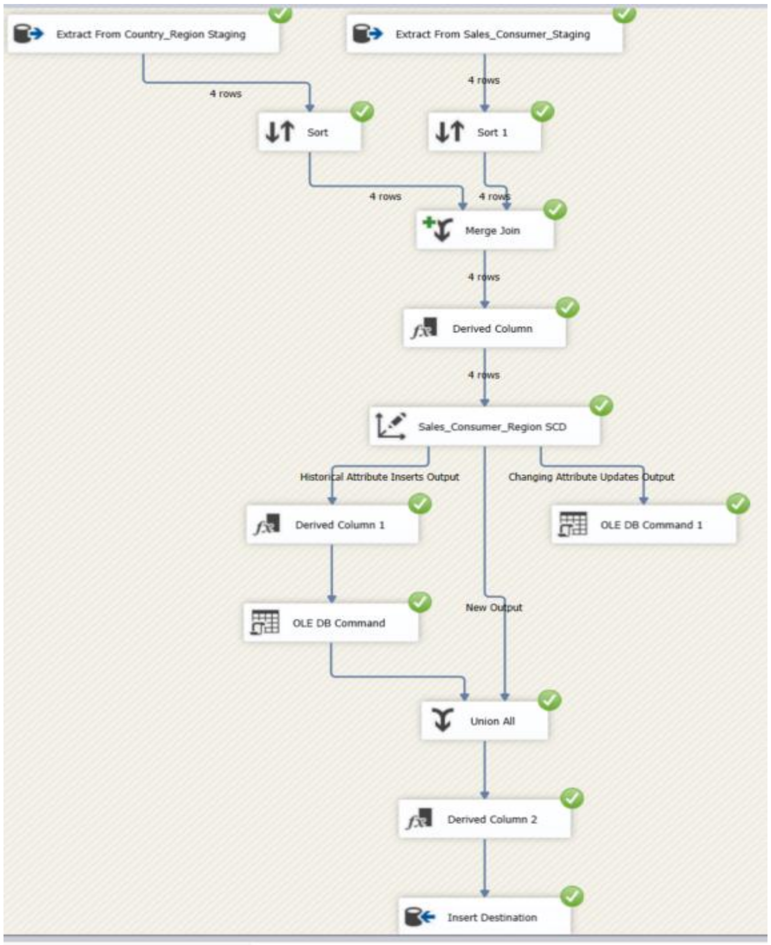
Historical Attribute

Select this type when changes in column values are saved in new records. Previous values are saved in records marked as outdated. This is a Type 2 change.

Select a change type for slowly changing dimen

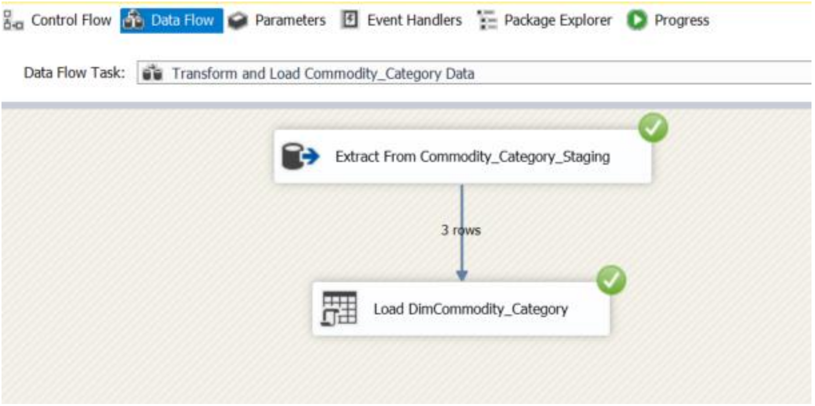
Dimension Columns	Change Type
Person	Changing at...
RegionName	Historical att...

- Following all the steps ,finally SalesConsumer data has been Loaded to DimSales_Consumer_Region table in data warehouse.

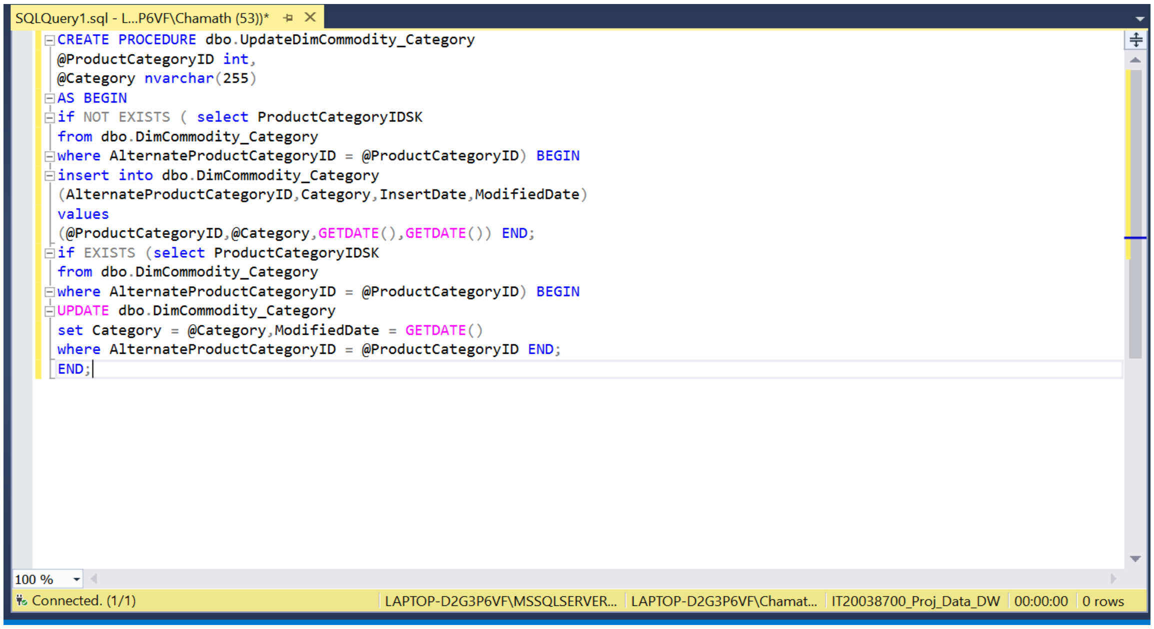


3) Transform and Load Commodity_Category Details

Commodity Category data has been loaded to Dim Commodity_Category


Control Flow
Data Flow
Parameters
Event Handlers
Package Explorer
Progress
Data Flow Task:
Transform and Load Commodity_Category Data
Extract From Commodity_Category_Staging
3 rows
Load DimCommodity_Category

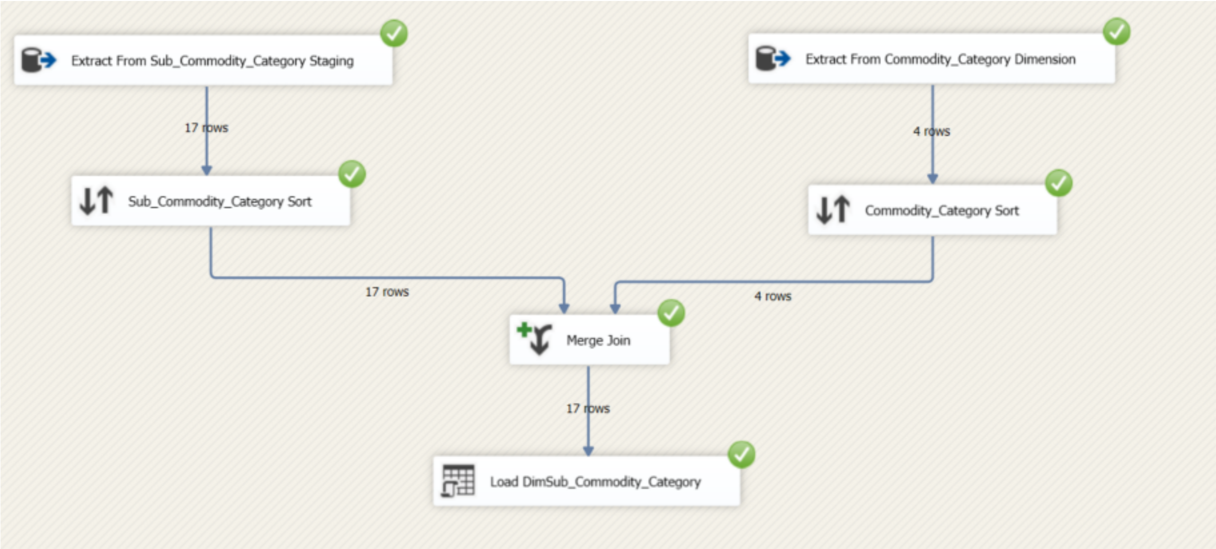
- Commodity Category data has been loaded to DimCommodity_Category table in Datawarehouse. The following procedure is used in order to load data.


SQLQuery1.sql - L...P6VF\Chamath (53))*
100 %
Connected. (1/1)
LAPTOP-D2G3P6VF\MSSQLSERVER...
LAPTOP-D2G3P6VF\Chamat...
IT20038700_Proj_Data_DW
00:00:00
0 rows

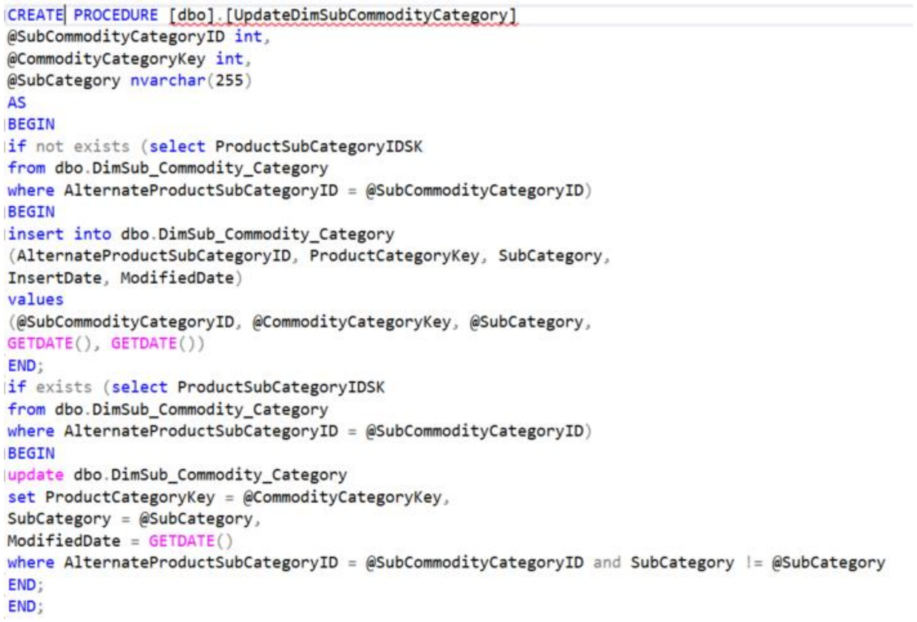
```
CREATE PROCEDURE dbo.UpdateDimCommodity_Category
@ProductCategoryID int,
@Category nvarchar(255)
AS BEGIN
if NOT EXISTS ( select ProductCategoryIDSK
from dbo.DimCommodity_Category
where AlternateProductCategoryID = @ProductCategoryID) BEGIN
insert into dbo.DimCommodity_Category
(AlternateProductCategoryID,Category,InsertDate,ModifiedDate)
values
(@ProductCategoryID,@Category,GETDATE(),GETDATE()) END;
if EXISTS (select ProductCategoryIDSK
from dbo.DimCommodity_Category
where AlternateProductCategoryID = @ProductCategoryID) BEGIN
UPDATE dbo.DimCommodity_Category
set Category = @Category,ModifiedDate = GETDATE()
where AlternateProductCategoryID = @ProductCategoryID END;
END;
```

4) 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


Extract From Sub_Commodity_Category Staging
Extract From Commodity_Category Dimension
17 rows
4 rows
Sub_Commodity_Category Sort
Commodity_Category Sort
17 rows
4 rows
Merge Join
17 rows
Load DimSub_Commodity_Category

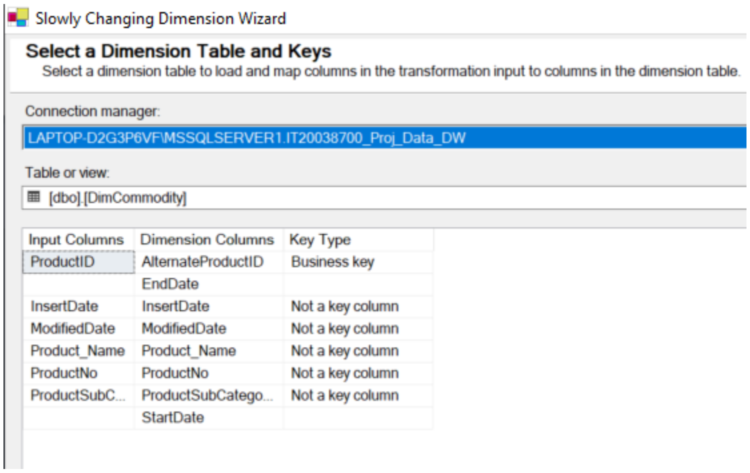
- Sub Commodity Category data has been loaded to DimSub_Commodity_Category table in Datawarehouse. The following procedure is used in order to load data.



```
CREATE PROCEDURE [dbo].[UpdateDimSubCommodityCategory]
@SubCommodityCategoryID int,
@CommodityCategoryKey int,
@SubCategory nvarchar(255)
AS
BEGIN
if not exists (select ProductSubCategoryIDSK
from dbo.DimSub_Commodity_Category
where AlternateProductSubCategoryID = @SubCommodityCategoryID)
BEGIN
insert into dbo.DimSub_Commodity_Category
(AlternateProductSubCategoryID, ProductCategoryKey, SubCategory,
InsertDate, ModifiedDate)
values
(@SubCommodityCategoryID, @CommodityCategoryKey, @SubCategory,
GETDATE(), GETDATE())
END;
if exists (select ProductSubCategoryIDSK
from dbo.DimSub_Commodity_Category
where AlternateProductSubCategoryID = @SubCommodityCategoryID)
BEGIN
update dbo.DimSub_Commodity_Category
set ProductCategoryKey = @CommodityCategoryKey,
SubCategory = @SubCategory,
ModifiedDate = GETDATE()
where AlternateProductSubCategoryID = @SubCommodityCategoryID and SubCategory != @SubCategory
END;
END;
```

5) 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.



Slowly Changing Dimension Wizard

Select a Dimension Table and Keys

Select a dimension table to load and map columns in the transformation input to columns in the dimension table.

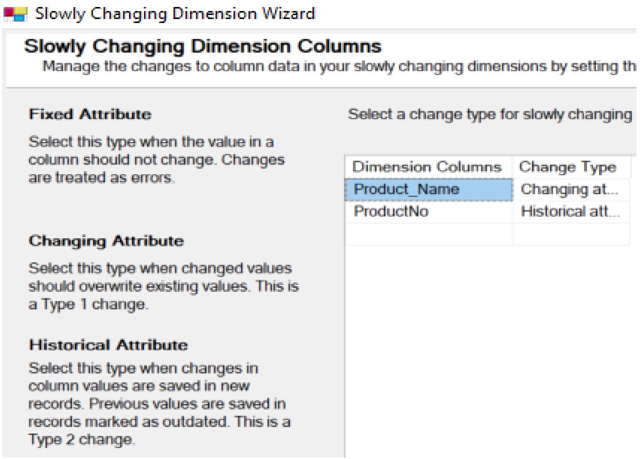
Connection manager:

LAPTOP-D2G3P6VF\MSSQLSERVER1.IT20038700_Proj_Data_DW

Table or view:

[dbo].[DimCommodity]

Input Columns	Dimension Columns	Key Type
ProductID	AlternateProductID	Business key
	EndDate	
InsertDate	InsertDate	Not a key column
ModifiedDate	ModifiedDate	Not a key column
Product_Name	Product_Name	Not a key column
ProductNo	ProductNo	Not a key column
ProductSubC...	ProductSubCatego...	Not a key column
	StartDate	



Slowly Changing Dimension Wizard

Slowly Changing Dimension Columns

Manage the changes to column data in your slowly changing dimensions by setting th

Fixed Attribute

Select this type when the value in a column should not change. Changes are treated as errors.

Changing Attribute

Select this type when changed values should overwrite existing values. This is a Type 1 change.

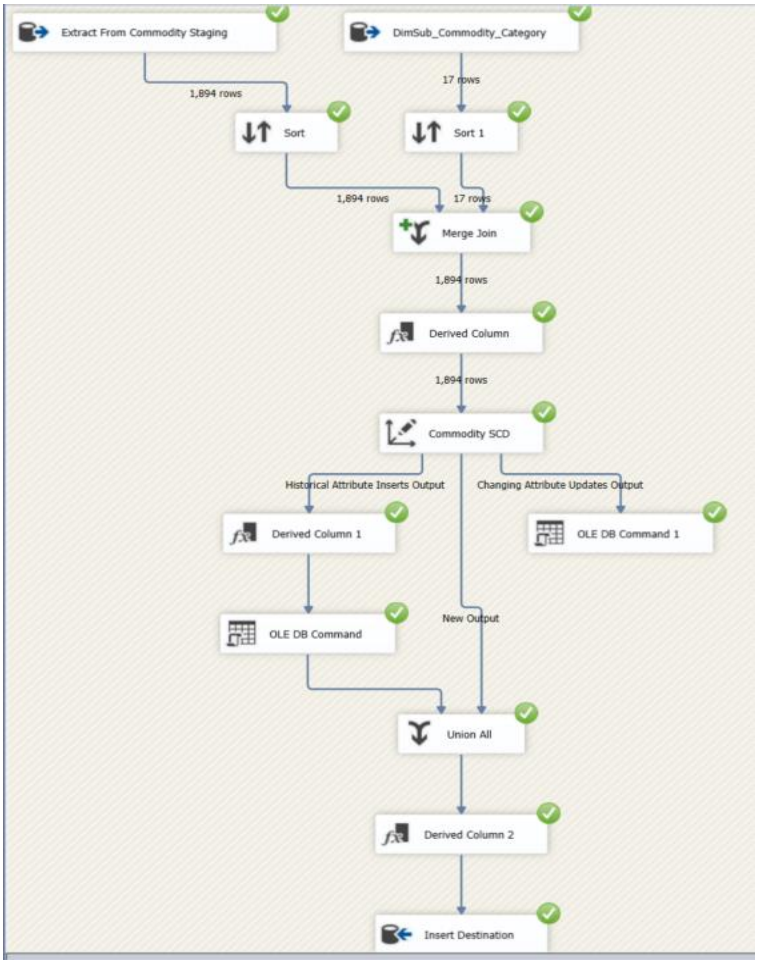
Historical Attribute

Select this type when changes in column values are saved in new records. Previous values are saved in records marked as outdated. This is a Type 2 change.

Select a change type for slowly changing

Dimension Columns	Change Type
Product_Name	Changing at...
ProductNo	Historical att...

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

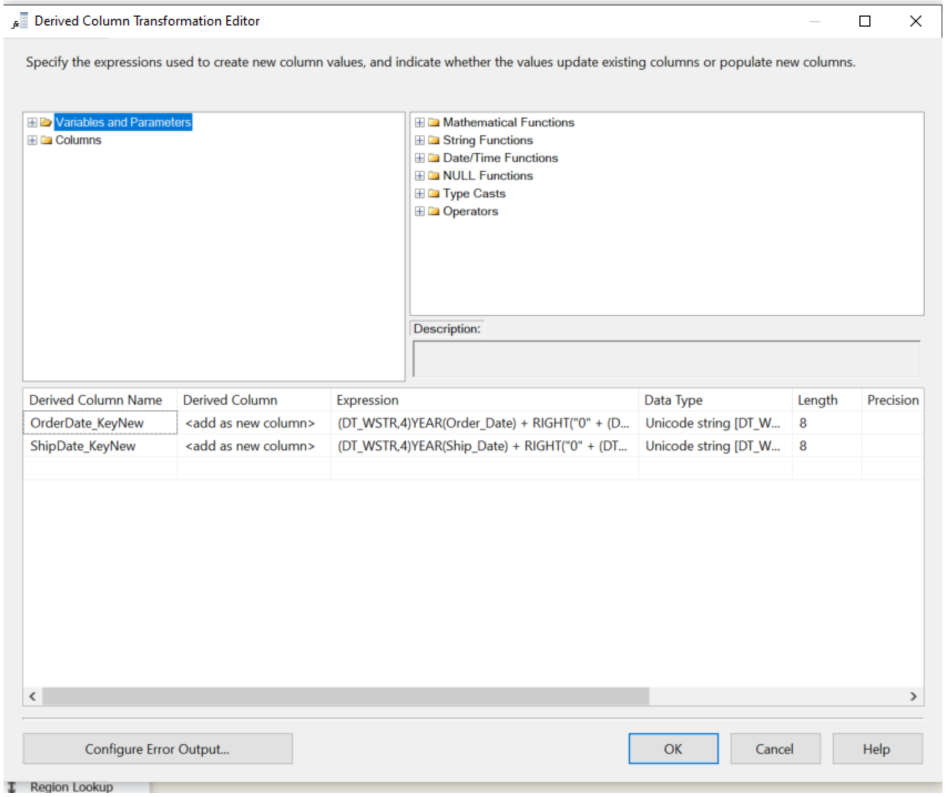


6) Transform and Load SalesFact

- 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.

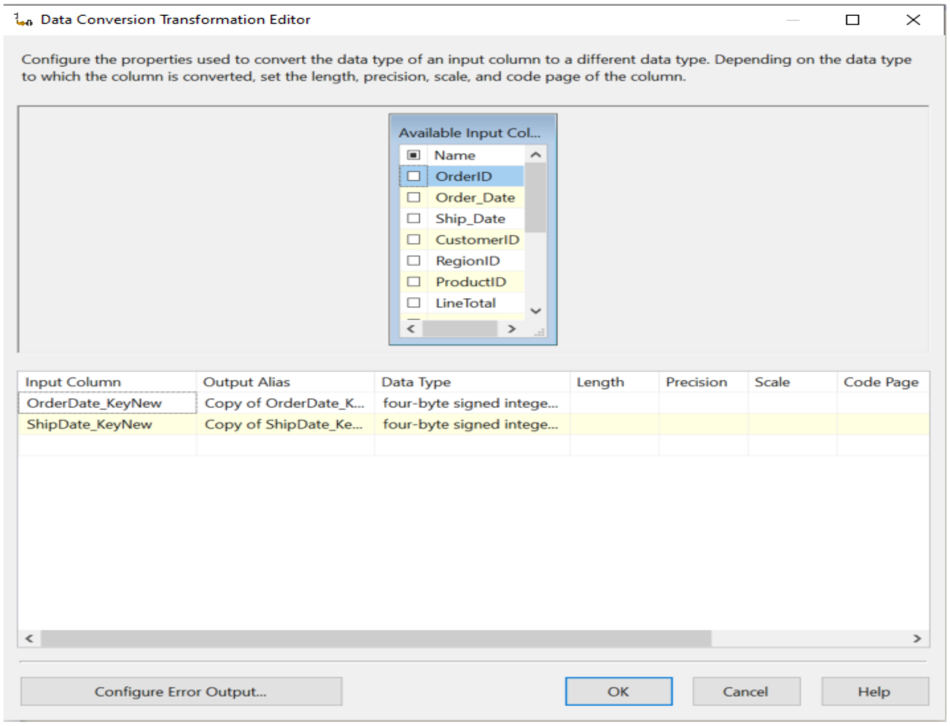


Derived Column Transformation Editor

Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Length	Precision
OrderDate_KeyNew	<add as new column>	(DT_WSTR,4)YEAR(Order_Date) + RIGHT("0" + (D...	Unicode string [DT_W...	8	
ShipDate_KeyNew	<add as new column>	(DT_WSTR,4)YEAR(Ship_Date) + RIGHT("0" + (DT...	Unicode string [DT_W...	8	

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

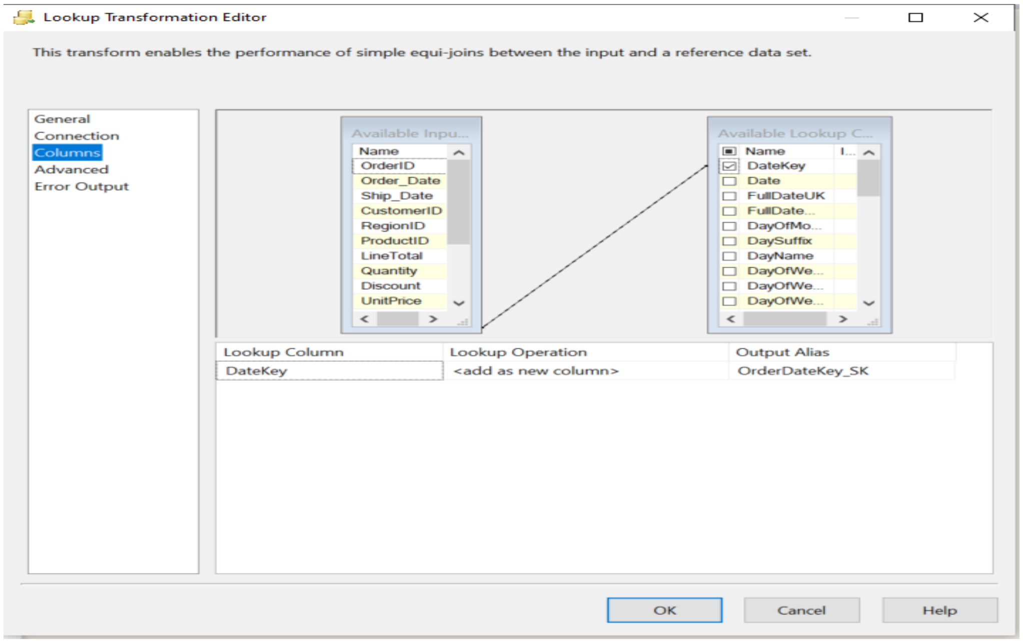


Data Conversion Transformation Editor

Configure the properties used to convert the data type of an input column to a different data type. Depending on the data type to which the column is converted, set the length, precision, scale, and code page of the column.

Input Column	Output Alias	Data Type	Length	Precision	Scale	Code Page
OrderDate_KeyNew	Copy of OrderDate_K...	four-byte signed intege...				
ShipDate_KeyNew	Copy of ShipDate_Ke...	four-byte signed intege...				

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



This transform enables the performance of simple equi-joins between the input and a reference data set.

General
Connection
Columns
Advanced
Error Output

Available Inpu...

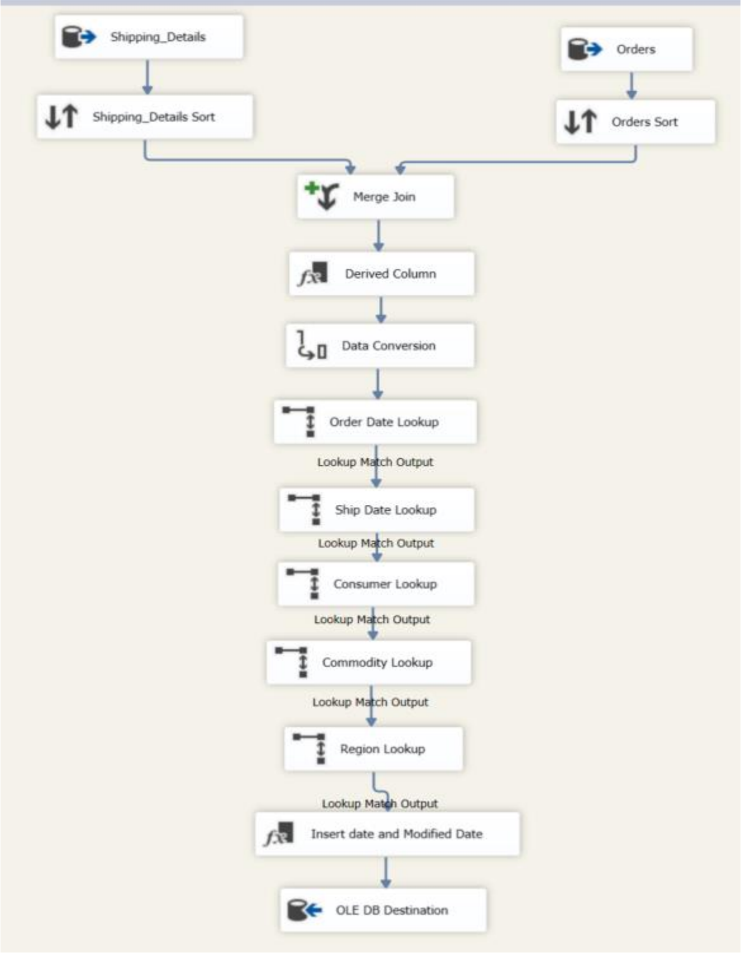
Name
OrderID
Order_Date
Ship_Date
CustomerID
RegionID
ProductID
LineTotal
Quantity
Discount
UnitPrice

Available Lookup C...

Name
☑ DateKey
☐ Date
☐ FullDateUK
☐ FullDate...
☐ DayOfMo...
☐ DaySuffix
☐ DayName
☐ DayOfWe...
☐ DayOfWe...
☐ DayOfWe...

Lookup Column	Lookup Operation	Output Alias
DateKey	<add as new column>	OrderDateKey_SK

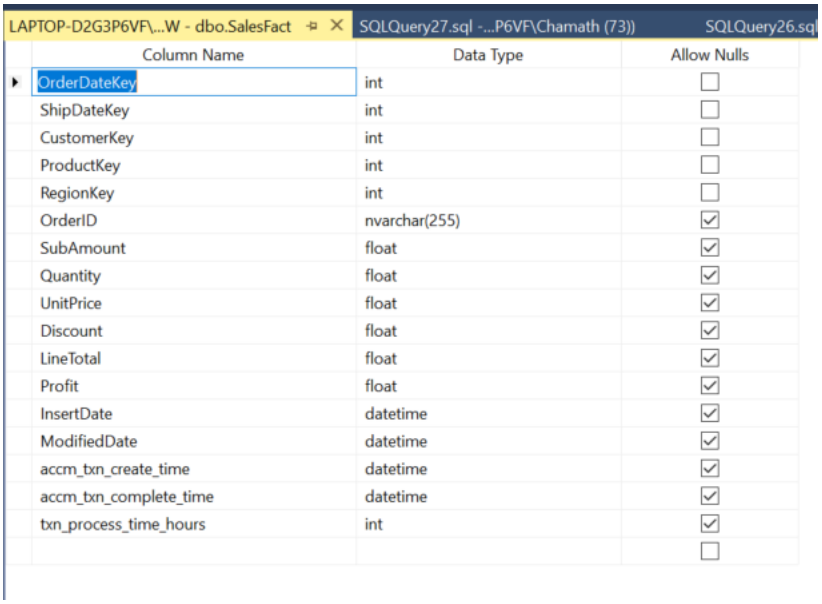
- This section of your data flow should look similar to below



Step 06: ETL Development-Accumulating Fact Table

Step 1:-

- I have extended SalesFact Table with 3 additional columns.

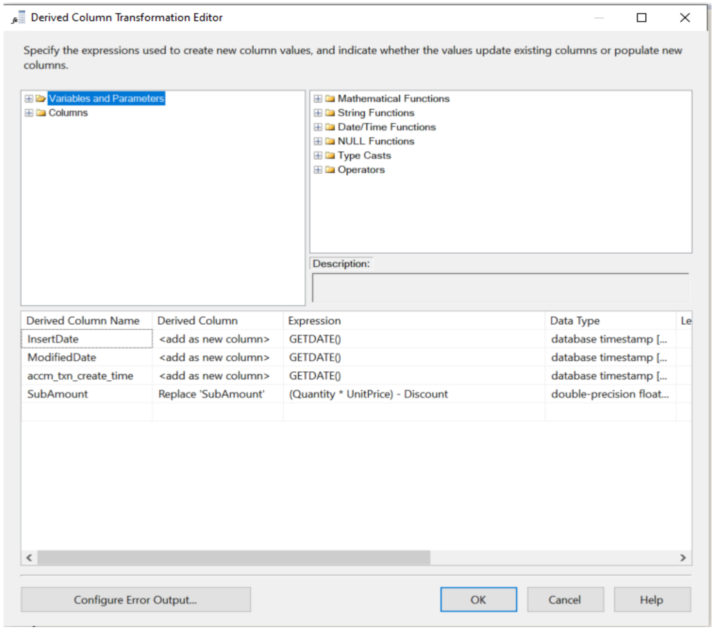


LAPTOP-D2G3P6VF\...W - dbo.SalesFact
SQLQuery27.sql -...P6VF\Chamath (73))
SQLQuery26.sql

Column Name	Data Type	Allow Nulls
OrderDateKey	int	□
ShipDateKey	int	□
CustomerKey	int	□
ProductKey	int	□
RegionKey	int	□
OrderID	nvarchar(255)	☑
SubAmount	float	☑
Quantity	float	☑
UnitPrice	float	☑
Discount	float	☑
LineTotal	float	☑
Profit	float	☑
InsertDate	datetime	☑
ModifiedDate	datetime	☑
accm_txn_create_time	datetime	☑
accm_txn_complete_time	datetime	☑
txn_process_time_hours	int	☑
		□

Step 2:-

- I had Set accm_txn_create_time to be equal to the current system date when load data to Salesfact table.

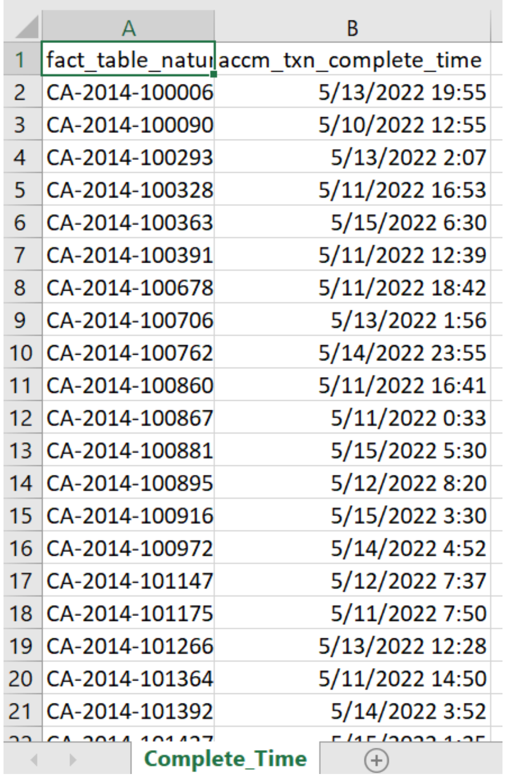


Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
InsertDate	<add as new column>	GETDATE()	database timestamp [...	
ModifiedDate	<add as new column>	GETDATE()	database timestamp [...	
accm_txn_create_time	<add as new column>	GETDATE()	database timestamp [...	
SubAmount	Replace 'SubAmount'	(Quantity * UnitPrice) - Discount	double-precision float...	

Step 3:-

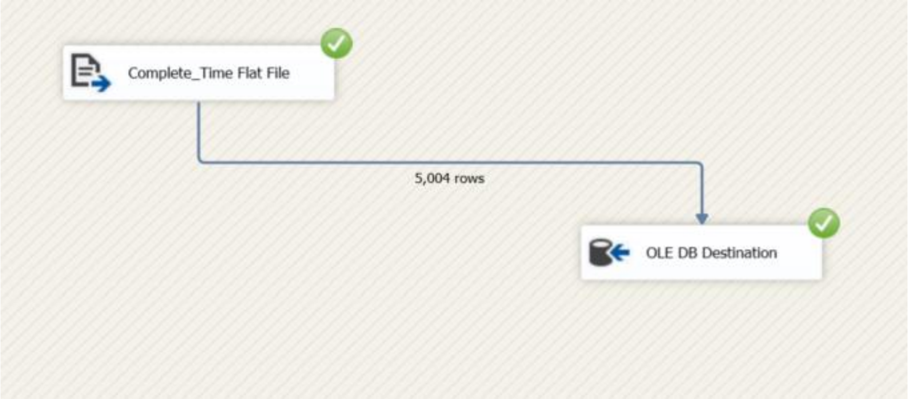
- I had Prepared a separate data set called Complete_Time.csv.



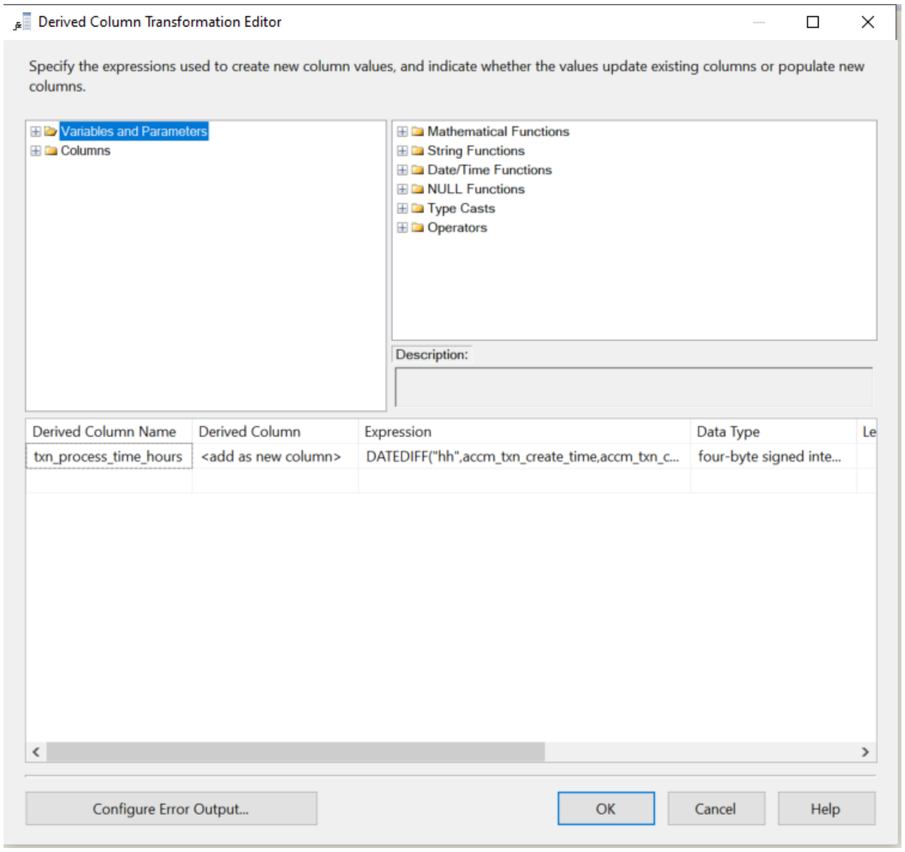
	A	B
1	fact_table_natui	accm_txn_complete_time
2	CA-2014-100006	5/13/2022 19:55
3	CA-2014-100090	5/10/2022 12:55
4	CA-2014-100293	5/13/2022 2:07
5	CA-2014-100328	5/11/2022 16:53
6	CA-2014-100363	5/15/2022 6:30
7	CA-2014-100391	5/11/2022 12:39
8	CA-2014-100678	5/11/2022 18:42
9	CA-2014-100706	5/13/2022 1:56
10	CA-2014-100762	5/14/2022 23:55
11	CA-2014-100860	5/11/2022 16:41
12	CA-2014-100867	5/11/2022 0:33
13	CA-2014-100881	5/15/2022 5:30
14	CA-2014-100895	5/12/2022 8:20
15	CA-2014-100916	5/15/2022 3:30
16	CA-2014-100972	5/14/2022 4:52
17	CA-2014-101147	5/12/2022 7:37
18	CA-2014-101175	5/11/2022 7:50
19	CA-2014-101266	5/13/2022 12:28
20	CA-2014-101364	5/11/2022 14:50
21	CA-2014-101392	5/14/2022 3:52
22	CA-2014-101427	5/15/2022 1:25

Step 4:-

- Designed Separate ETL package and load Complete_Time.csv to Staging.



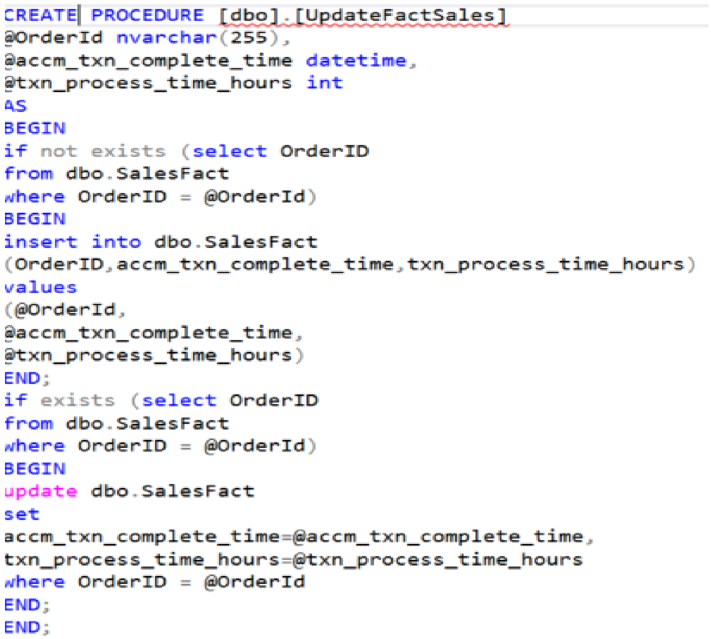
- Used Derived Column to Calculate Processing hours.



Derived Column Transformation Editor

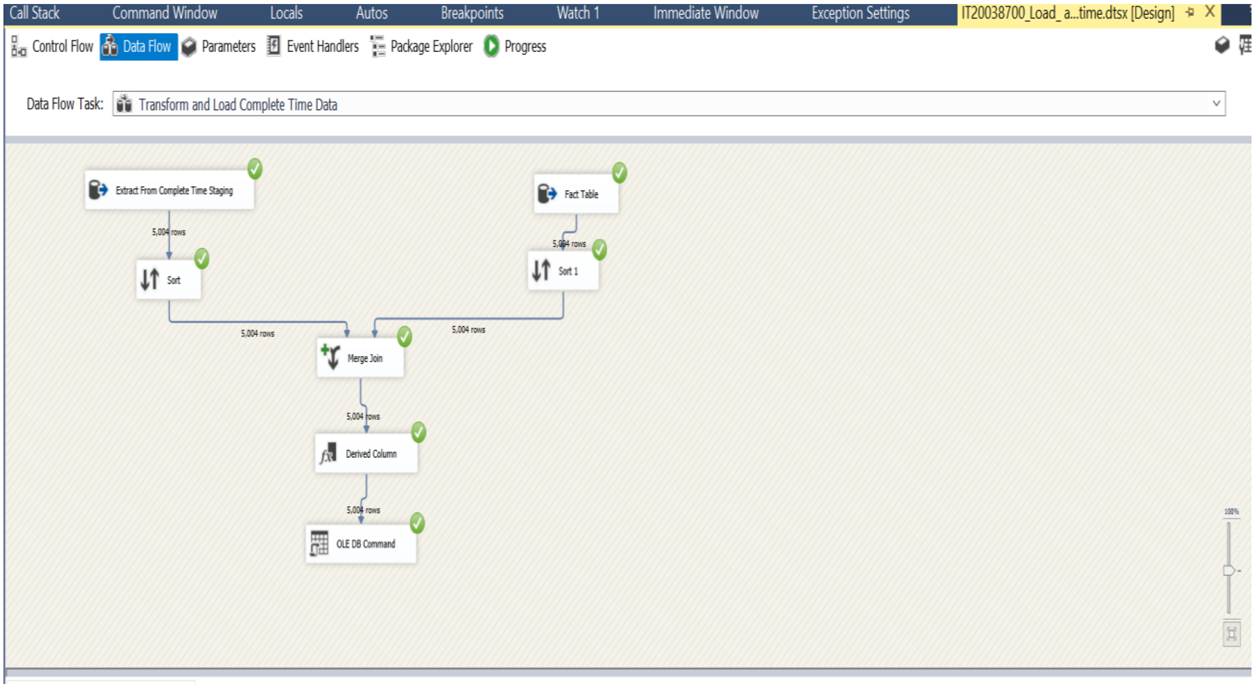
Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
txn_process_time_hours	<add as new column>	DATEDIFF("hh",accm_txn_create_time,accm_txn_c...	four-byte signed inte...	



```
CREATE PROCEDURE [dbo].[UpdateFactSales]
@OrderId nvarchar(255),
@accm_txn_complete_time datetime,
@txn_process_time_hours int
AS
BEGIN
if not exists (select OrderID
from dbo.SalesFact
where OrderID = @OrderId)
BEGIN
insert into dbo.SalesFact
(OrderID,accm_txn_complete_time,txn_process_time_hours)
values
(@OrderId,
@accm_txn_complete_time,
@txn_process_time_hours)
END;
if exists (select OrderID
from dbo.SalesFact
where OrderID = @OrderId)
BEGIN
update dbo.SalesFact
set
accm_txn_complete_time=@accm_txn_complete_time,
txn_process_time_hours=@txn_process_time_hours
where OrderID = @OrderId
END;
END;
```

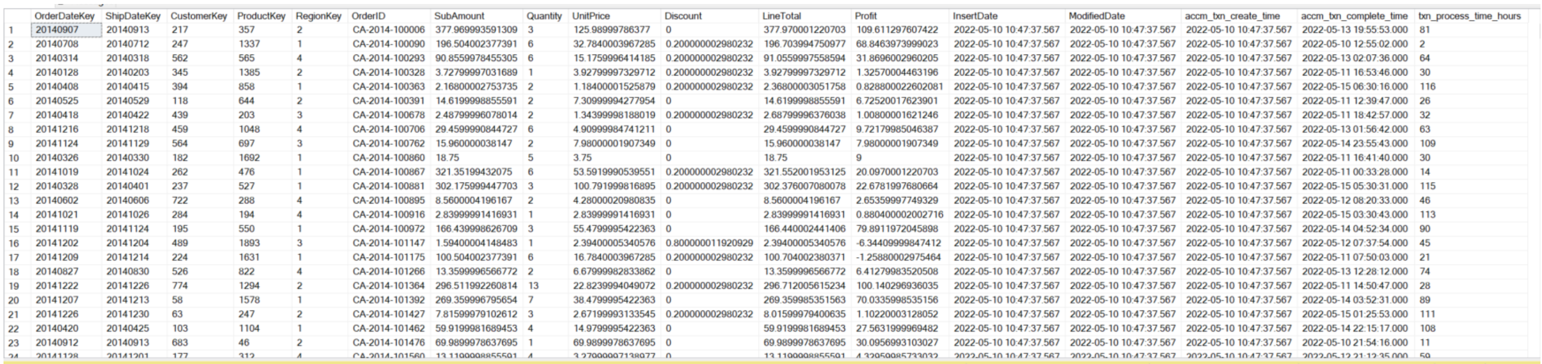
- Update the corresponding accm_txn_complete_time in your DW fact table.



Data Flow Task: Transform and Load Complete Time Data

Step 5:-

- Updated SalesFact Table.



	OrderDateKey	ShipDateKey	CustomerKey	ProductKey	RegionKey	OrderID	SubAmount	Quantity	UnitPrice	Discount	LineTotal	Profit	InsertDate	ModifiedDate	accm_txn_create_time	accm_txn_complete_time	txn_process_time_hours
1	20140907	20140913	217	357	2	CA-2014-100006	377.969993591309	3	125.98999786377	0	377.970001220703	109.611297607422	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-13 19:55:53.000	81
2	20140708	20140712	247	1337	1	CA-2014-100090	196.504002377391	6	32.7840003967285	0.200000002980232	196.703994750977	68.8463973999023	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 12:55:02.000	2
3	20140314	20140318	562	565	4	CA-2014-100293	90.8559978455305	6	15.1759996414185	0.200000002980232	91.0559997558594	31.8696002960205	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-13 02:07:36.000	64
4	20140128	20140203	345	1385	2	CA-2014-100328	3.72799997031689	1	3.92799997329712	0.200000002980232	3.92799997329712	1.32570004463196	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 16:53:46.000	30
5	20140408	20140415	394	858	1	CA-2014-100363	2.1680002753735	2	1.18400001525879	0.200000002980232	2.36800003051758	0.828800022602081	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-15 06:30:16.000	116
6	20140525	20140529	118	644	2	CA-2014-100391	14.6199998855591	2	7.30999994277954	0	14.6199998855591	6.72520017623901	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 12:39:47.000	26
7	20140418	20140422	439	203	3	CA-2014-100678	2.48799996078014	2	1.34399998188019	0.200000002980232	2.68799996376038	1.00800001621246	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 18:42:57.000	32
8	20141216	20141218	459	1048	4	CA-2014-100706	29.4599990844727	6	4.90999984741211	0	29.4599990844727	9.72179985046387	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-13 01:56:42.000	63
9	20141124	20141129	564	697	3	CA-2014-100762	15.960000038147	2	7.98000001907349	0	15.960000038147	7.98000001907349	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-14 23:55:43.000	109
10	20140326	20140330	182	1692	1	CA-2014-100860	18.75	5	3.75	0	18.75	9	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 16:41:40.000	30
11	20141019	20141024	262	476	1	CA-2014-100867	321.35199432075	6	53.5919990539551	0.200000002980232	321.552001953125	20.0970001220703	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 00:33:28.000	14
12	20140328	20140401	237	527	1	CA-2014-100881	302.175999447703	3	100.791999816895	0.200000002980232	302.376007080078	22.6781997680664	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-15 05:30:31.000	115
13	20140602	20140606	722	288	4	CA-2014-100895	8.5600004196167	2	4.28000020980835	0	8.5600004196167	2.65359997749329	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-12 08:20:33.000	46
14	20141021	20141026	284	194	4	CA-2014-100916	2.83999991416931	1	2.83999991416931	0	2.83999991416931	0.880400002002716	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-15 03:30:43.000	113
15	20141119	20141124	195	550	1	CA-2014-100972	166.439998626709	3	55.4799995422363	0	166.440002441406	79.8911972045898	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-14 04:52:34.000	90
16	20141202	20141204	489	1893	3	CA-2014-101147	1.59400004148483	1	2.39400005340576	0.800000011920929	2.39400005340576	-6.34409999847412	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-12 07:37:54.000	45
17	20141209	20141214	224	1631	1	CA-2014-101175	100.504002377391	6	16.7840003967285	0.200000002980232	100.704002380371	-1.25880002975464	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 07:50:03.000	21
18	20140827	20140830	526	822	4	CA-2014-101266	13.3599996566772	2	6.67999982833862	0	13.3599996566772	6.41279983520508	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-13 12:28:12.000	74
19	20141222	20141226	774	1294	2	CA-2014-101364	296.511992260814	13	22.8239994049072	0.200000002980232	296.712005615234	100.140296936035	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-11 14:50:47.000	28
20	20141207	20141213	58	1578	1	CA-2014-101392	269.359996795654	7	38.4799995422363	0	269.359985351563	70.0335998535156	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-14 03:52:31.000	89
21	20141226	20141230	63	247	2	CA-2014-101427	7.81599979102612	3	2.67199993133545	0.200000002980232	8.01599979400635	1.10220003128052	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-15 01:25:53.000	111
22	20140420	20140425	103	1104	1	CA-2014-101462	59.9199981689453	4	14.9799995422363	0	59.9199981689453	27.5631999969482	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-14 22:15:17.000	108
23	20140912	20140913	683	46	2	CA-2014-101476	69.9899978637695	1	69.9899978637695	0	69.9899978637695	30.0956993103027	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 21:54:16.000	11
24	20141128	20141201	177	312	4	CA-2014-101560	13.1199998855591	4	3.27999997138977	0	13.1199998855591	4.32959985733032	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-10 10:47:37.567	2022-05-12 21:12:35.000	59

References

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

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