ΟΙΚΟΝΟΜΙΚΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΑΘΗΝΩΝ

ATHENS UNIVERSITY
OF ECONOMICS
AND BUSINESS

BUSINESS
ANALYTICS
Master of Science

MSc in Business Analytics - Part time

2nd Assignment
Data Management & Business Intelligence

Professor: Mr. Chatziantoniou Damianos

Students: Arseniou Evangelia (p2822026)
Ligkou Sotiria (p2822023)

Table of Contents

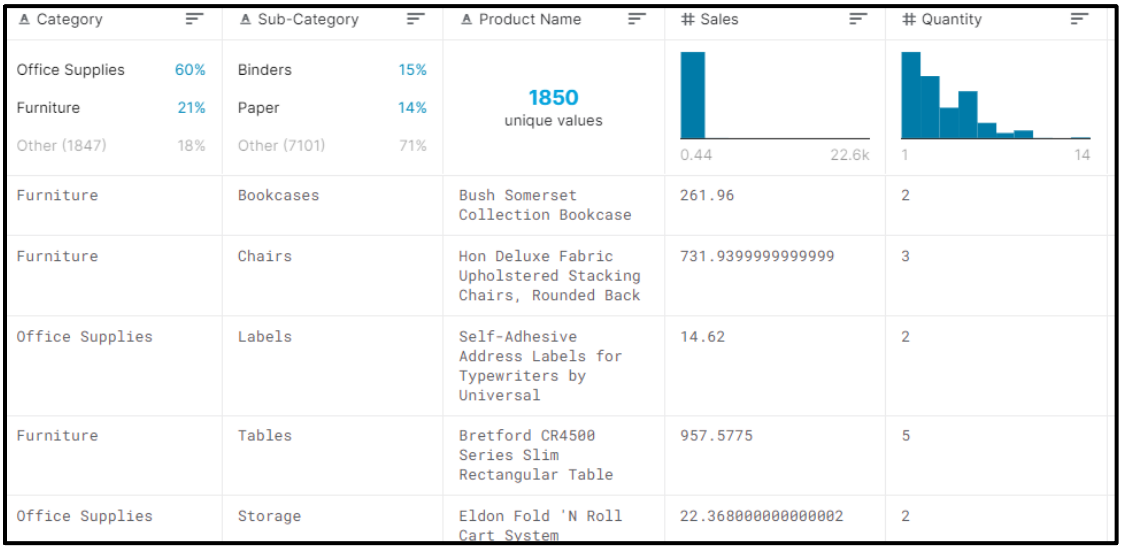
1. Description of The Case

E-commerce is a modern method of promoting products and services with fact growth in the field of Internet. Online stores, such as Amazon and eBay, are now established, and thousands of individual, family and large businesses have incorporated e-commerce into their business activities. The speed and ease with which the buying and selling process can take place, rank e-commerce high in the preferences and activities of internet users. This method allows people to buy products from books, toys, clothes and shoes to food, furniture and other household. With that in mind, an analysis based on products from an online store is worth it to be done, because year by year e-commerce is replacing the purchases that can be done with being physical in a store. In addition, while doing such us analysis, a business like this can identify the products have sold most or what kind of products customers want, so that the store can be equipped to make them more loyal.

The dataset that we have scraped is belong to Kaggle.com. It refers to a Superstore which is an online store based in United States. It contains data from 2014-2018 and describes transactions made in this online store during these years. More specifically, the dataset contains information about **Date of order**, **Shipping** of products, **Customers** and their **Location**, the **Products** that they ordered, **Quantities**, **Total Amount** of each transaction, **Discounts** and the **Profit** that the business gained due to these orders. Products also belong to **Categories** and **Subcategories**.

The main scope of this analysis is to identify the products that have been sold mostly, which customers are more loyal to Superstore, which countries purchase more and when. Also, categories have helped us realize what is trending in this Superstore.

The initial dataset consists of 9.994 rows, each of which is a transaction, and twenty-one columns. The ETL (Extract, Transform, Load) process that have been made, reached to dataset of the same rows and sixteen columns. A small overview of our dataset before transformation and cleaning is shown below:



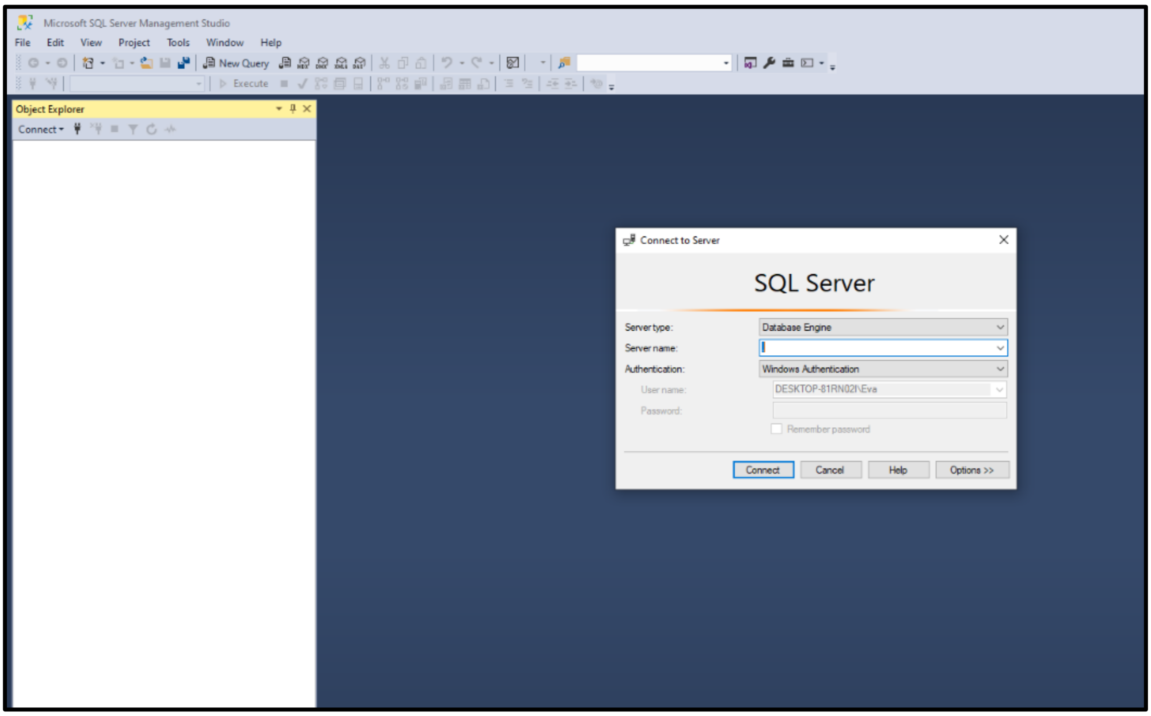
Category	Sub-Category	Product Name	Sales	Quantity
Office Supplies 60% Furniture 21% Other (1847) 18%	Binders 15% Paper 14% Other (7101) 71%	1850 unique values	0.44 22.6k	1 14
Furniture	Bookcases	Bush Somerset Collection Bookcase	261.96	2
Furniture	Chairs	Hon Deluxe Fabric Upholstered Stacking Chairs, Rounded Back	731.9399999999999	3
Office Supplies	Labels	Self-Adhesive Address Labels for Typewriters by Universal	14.62	2
Furniture	Tables	Bretford CR4500 Series Slim Rectangular Table	957.5775	5
Office Supplies	Storage	Eldon Fold 'N Roll Cart System	22.368000000000002	2

Table 1.Overview of data

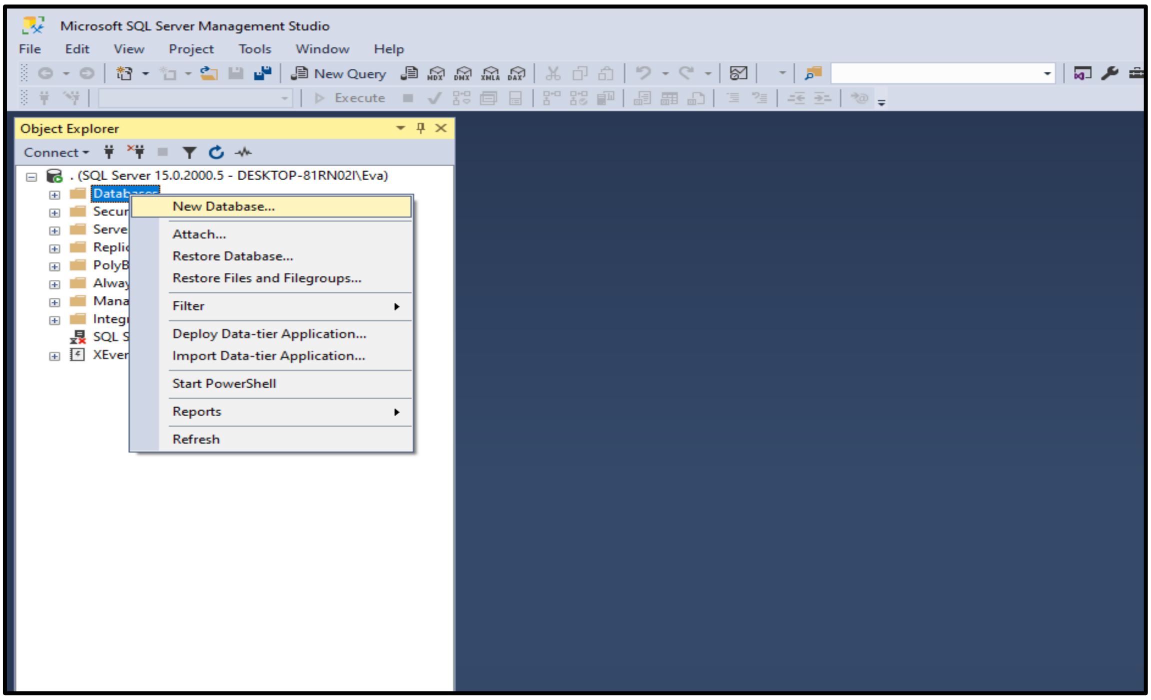
Process of data insertion in SQL Server Management Studio

First of all, we had to connect to Microsoft SQL Server Management Studio and create a new database with the name ‘US Superstore’.

.

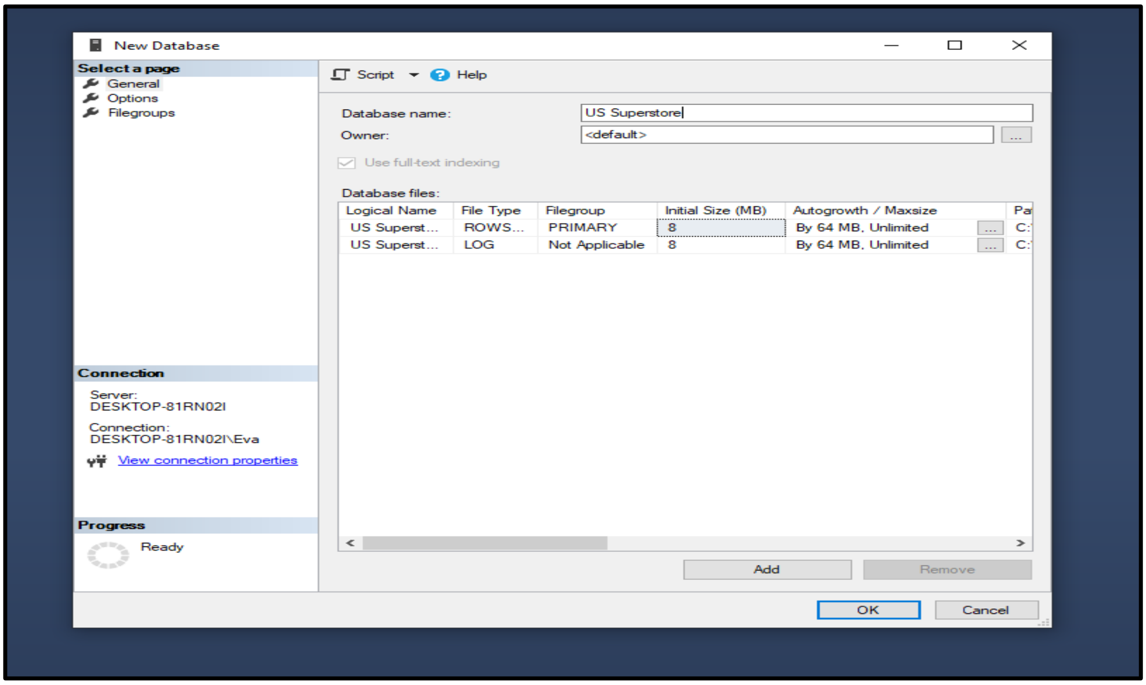


Server type: Database Engine
Server name:
Authentication: Windows Authentication
User name: DESKTOP-81RN02I\Eva
Password:
□ Remember password



Microsoft SQL Server Management Studio
File Edit View Project Tools Window Help
New Query
Execute
Object Explorer
Connect
Datab
Secur
Serve
Replic
PolyB
Alway
Mana
Integ
SQL S
XEver
New Database...
Attach...
Restore Database...
Restore Files and Filegroups...
Filter
Deploy Data-tier Application...
Import Data-tier Application...
Start PowerShell
Reports
Refresh

. (SQL Server 15.0.2000.5 - DESKTOP-81RN02I\Eva)



Database name: US Superstore

Owner: <default>

☑ Use full-text indexing

Database files:

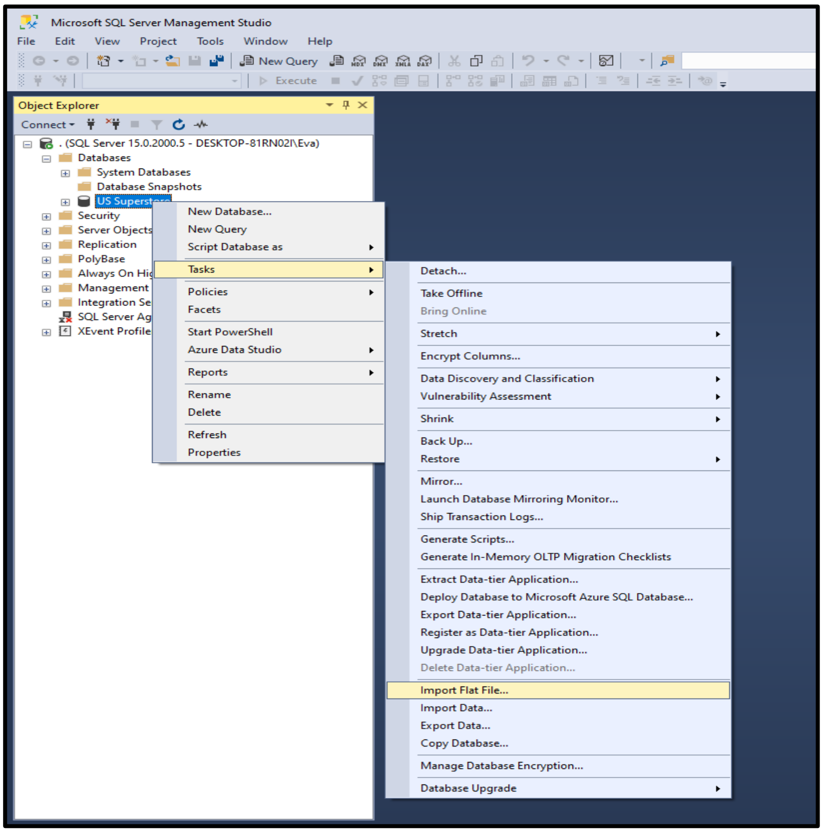
Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize	Pa
US Superst...	ROWS...	PRIMARY	8	By 64 MB, Unlimited	C:
US Superst...	LOG	Not Applicable	8	By 64 MB, Unlimited	C:

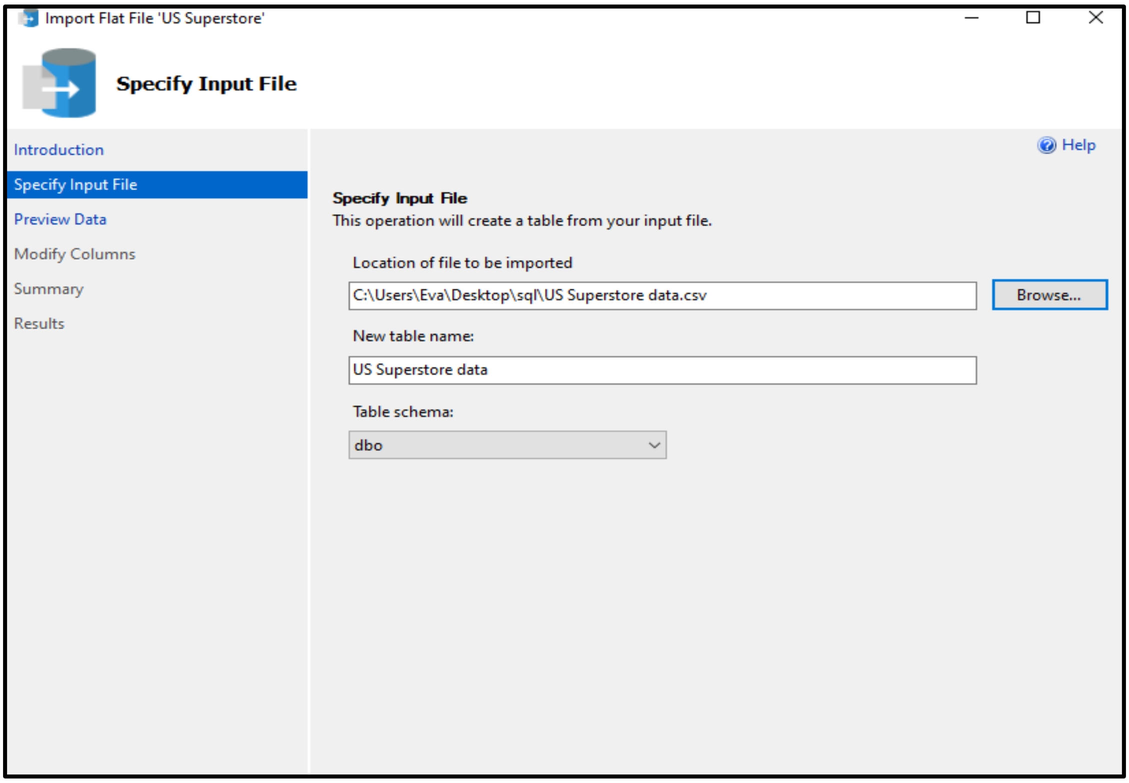
Server:
DESKTOP-81RN02I

Connection:
DESKTOP-81RN02I\Eva

View connection properties

The data were imported as a flat file due to their csv form. Our table which contains all the data is called ‘US Superstore data’.





Specify Input File

Specify Input File

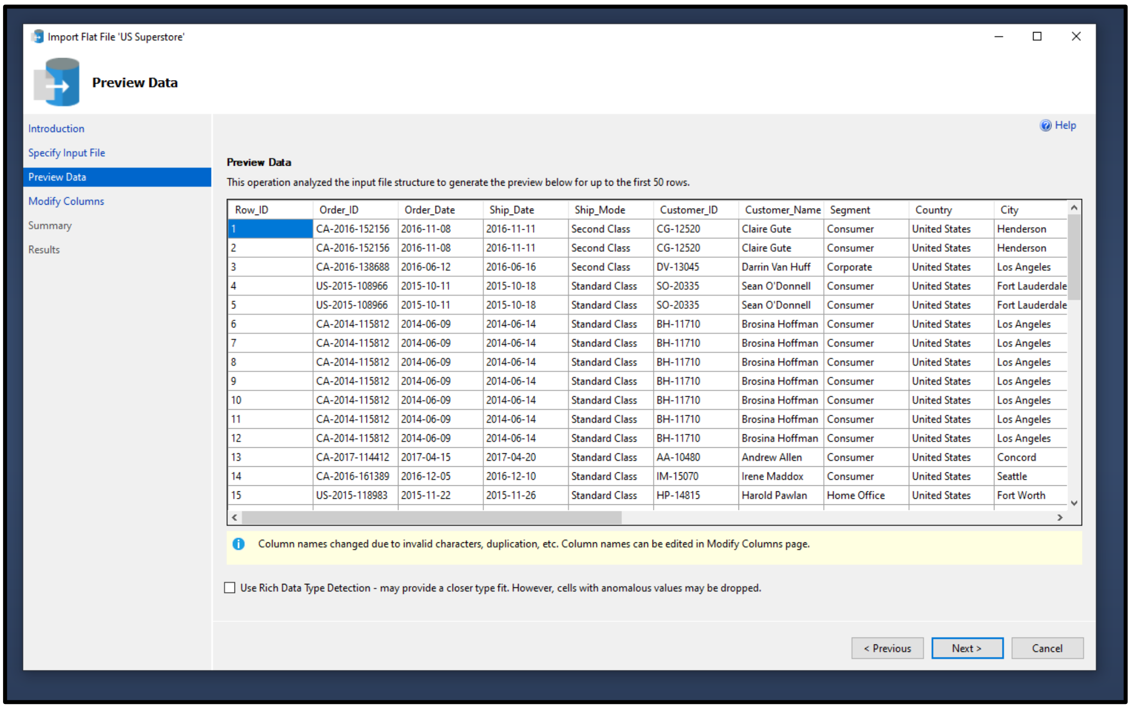
This operation will create a table from your input file.

Location of file to be imported: C:\Users\Eva\Desktop\sql\US Superstore data.csv

New table name: US Superstore data

Table schema: dbo

Next tabs show a small preview of the data in order to check if they were imported well and were specified in the correct type.



Preview Data

Preview Data

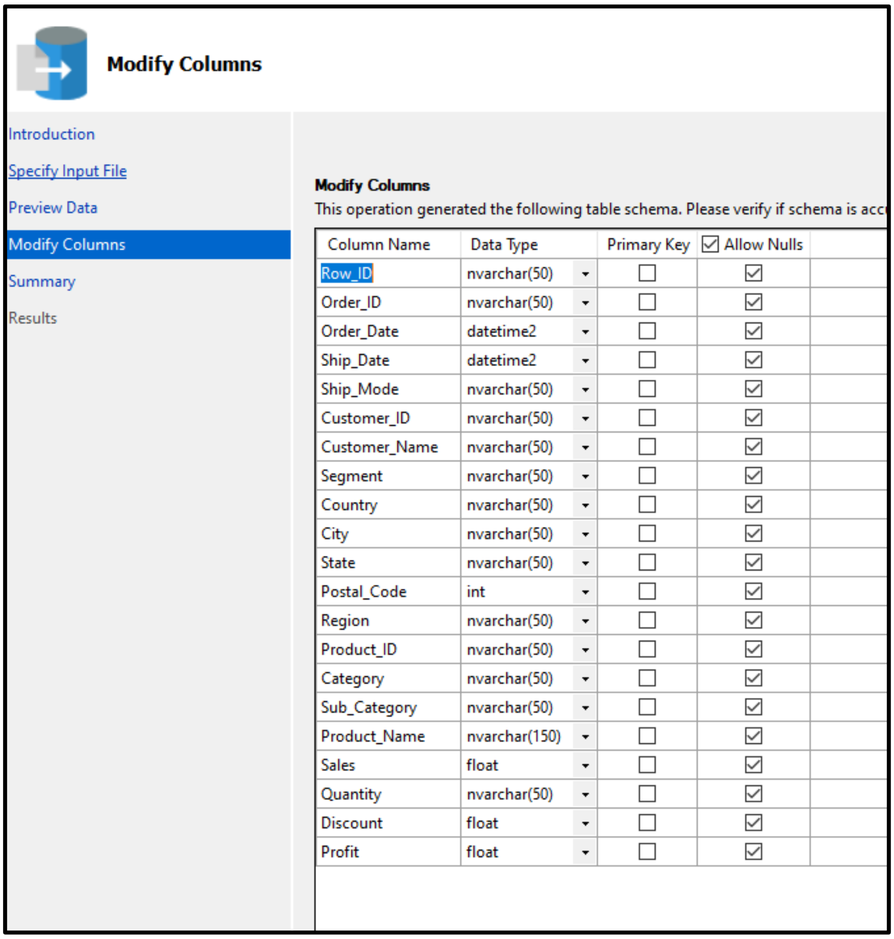
This operation analyzed the input file structure to generate the preview below for up to the first 50 rows.

Row_ID	Order_ID	Order_Date	Ship_Date	Ship_Mode	Customer_ID	Customer_Name	Segment	Country	City
1	CA-2016-152156	2016-11-08	2016-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson
2	CA-2016-152156	2016-11-08	2016-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson
3	CA-2016-138688	2016-06-12	2016-06-16	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	Los Angeles
4	US-2015-108966	2015-10-11	2015-10-18	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale
5	US-2015-108966	2015-10-11	2015-10-18	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale
6	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
7	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
8	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
9	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
10	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
11	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
12	CA-2014-115812	2014-06-09	2014-06-14	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles
13	CA-2017-114412	2017-04-15	2017-04-20	Standard Class	AA-10480	Andrew Allen	Consumer	United States	Concord
14	CA-2016-161389	2016-12-05	2016-12-10	Standard Class	IM-15070	Irene Maddox	Consumer	United States	Seattle
15	US-2015-118983	2015-11-22	2015-11-26	Standard Class	HP-14815	Harold Pawlan	Home Office	United States	Fort Worth

Column names changed due to invalid characters, duplication, etc. Column names can be edited in Modify Columns page.

☐ Use Rich Data Type Detection - may provide a closer type fit. However, cells with anomalous values may be dropped.

In section ‘Modify Columns’ we made the below modifications due to wrong Data types from SQL. Everything except numeric variables were defined as nvarchar. We converted Sales, Discount and Profit into float due to decimal numbers. In addition, Order Date and Ship Date were correctly inserted with datatime2 type. Finally, we allowed nulls in all variables and made some changes later.

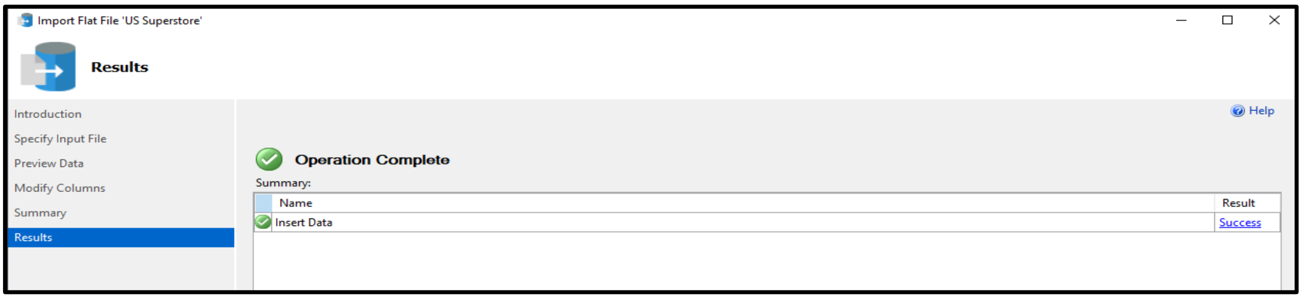


Modify Columns

This operation generated the following table schema. Please verify if schema is acc

Column Name	Data Type	Primary Key	☑ Allow Nulls
Row_ID	nvarchar(50)	□	☑
Order_ID	nvarchar(50)	□	☑
Order_Date	datetime2	□	☑
Ship_Date	datetime2	□	☑
Ship_Mode	nvarchar(50)	□	☑
Customer_ID	nvarchar(50)	□	☑
Customer_Name	nvarchar(50)	□	☑
Segment	nvarchar(50)	□	☑
Country	nvarchar(50)	□	☑
City	nvarchar(50)	□	☑
State	nvarchar(50)	□	☑
Postal_Code	int	□	☑
Region	nvarchar(50)	□	☑
Product_ID	nvarchar(50)	□	☑
Category	nvarchar(50)	□	☑
Sub_Category	nvarchar(50)	□	☑
Product_Name	nvarchar(150)	□	☑
Sales	float	□	☑
Quantity	nvarchar(50)	□	☑
Discount	float	□	☑
Profit	float	□	☑

Our data has imported successfully.

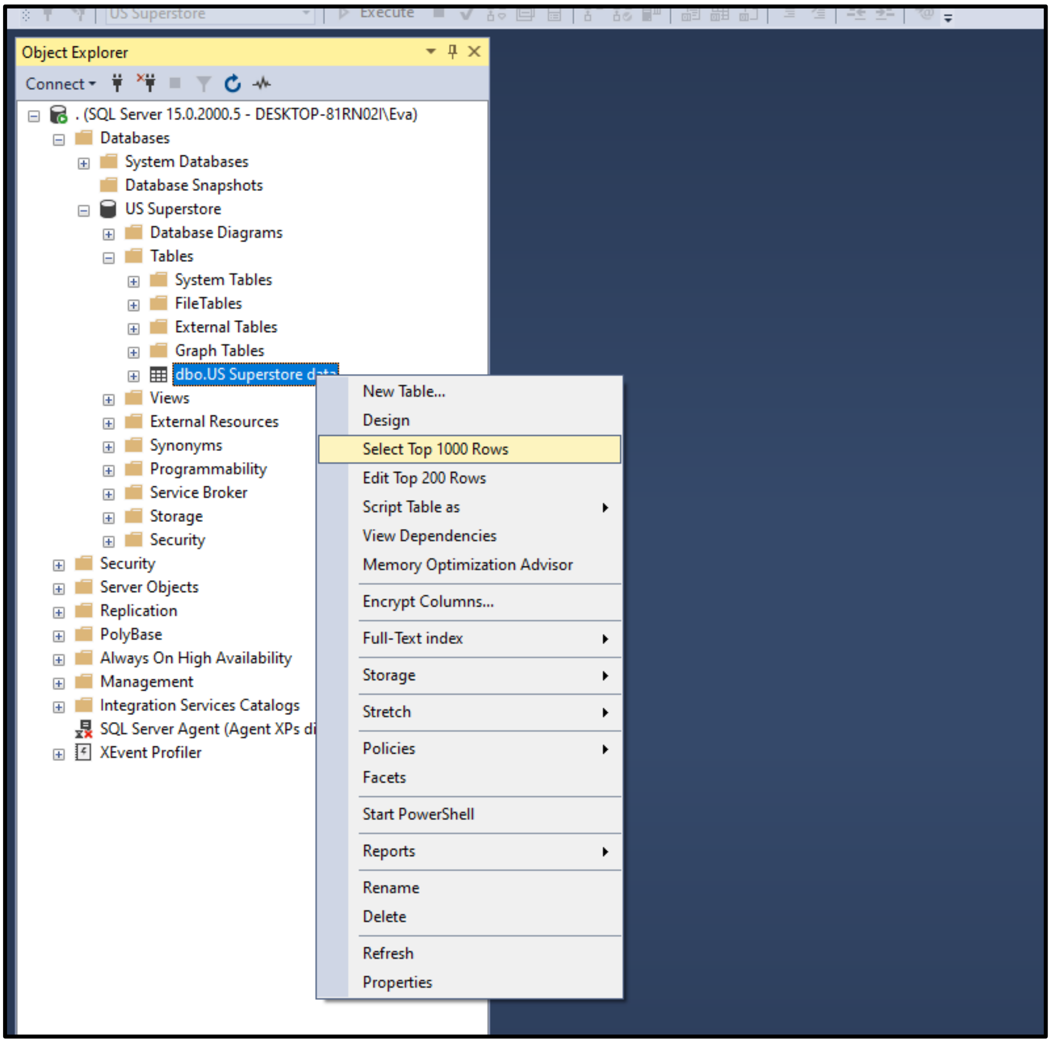


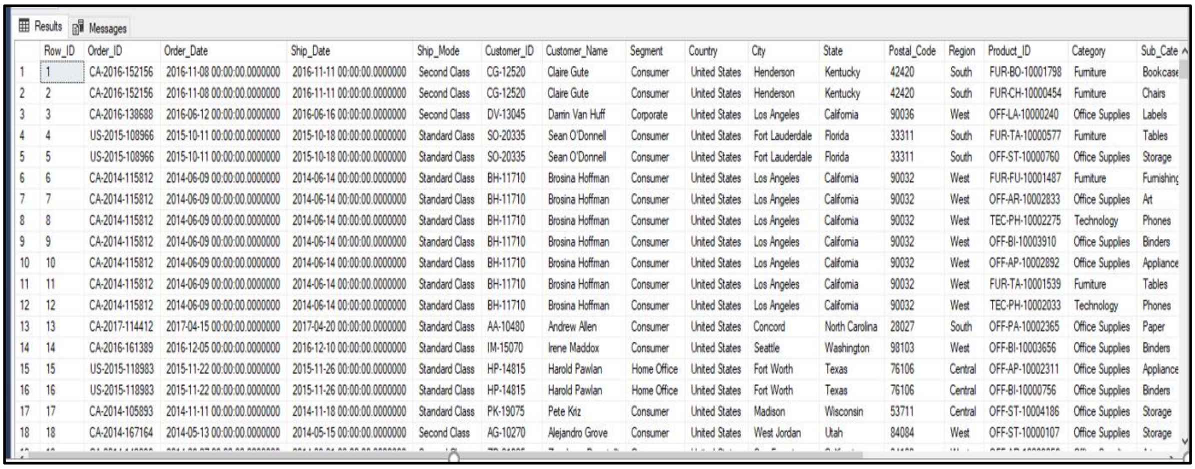
Operation Complete

Summary:

Name	Result
Insert Data	Success

If we select the top 1000 rows from our table, we could easily view the structure of our data as shown below.



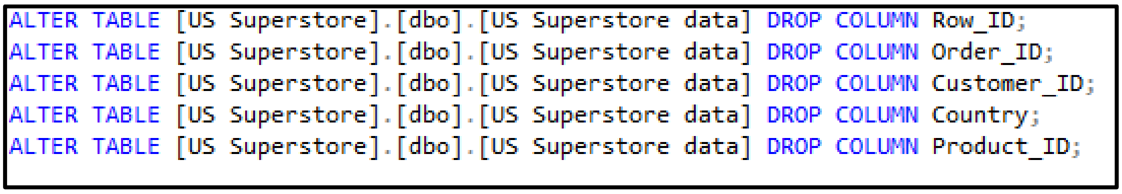


	Row_ID	Order_ID	Order_Date	Ship_Date	Ship_Mode	Customer_ID	Customer_Name	Segment	Country	City	State	Postal_Code	Region	Product_ID	Category	Sub_Cate
1	1	CA-2016-152156	2016-11-08 00:00:00.0000000	2016-11-11 00:00:00.0000000	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	Kentucky	42420	South	FUR-BO-10001798	Furniture	Bookcase
2	2	CA-2016-152156	2016-11-08 00:00:00.0000000	2016-11-11 00:00:00.0000000	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	Kentucky	42420	South	FUR-CH-10000454	Furniture	Chairs
3	3	CA-2016-138688	2016-06-12 00:00:00.0000000	2016-06-16 00:00:00.0000000	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	Los Angeles	California	90036	West	OFF-LA-10000240	Office Supplies	Labels
4	4	US-2015-108966	2015-10-11 00:00:00.0000000	2015-10-18 00:00:00.0000000	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale	Florida	33311	South	FUR-TA-10000577	Furniture	Tables
5	5	US-2015-108966	2015-10-11 00:00:00.0000000	2015-10-18 00:00:00.0000000	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale	Florida	33311	South	OFF-ST-10000760	Office Supplies	Storage
6	6	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	FUR-FU-10001487	Furniture	Furnishing
7	7	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	OFF-AR-10002833	Office Supplies	Art
8	8	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	TEC-PH-10002275	Technology	Phones
9	9	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	OFF-BI-10003910	Office Supplies	Binders
10	10	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	OFF-AP-10002892	Office Supplies	Appliance
11	11	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	FUR-TA-10001539	Furniture	Tables
12	12	CA-2014-115812	2014-06-09 00:00:00.0000000	2014-06-14 00:00:00.0000000	Standard Class	BH-11710	Brosina Hoffman	Consumer	United States	Los Angeles	California	90032	West	TEC-PH-10002033	Technology	Phones
13	13	CA-2017-114412	2017-04-15 00:00:00.0000000	2017-04-20 00:00:00.0000000	Standard Class	AA-10480	Andrew Allen	Consumer	United States	Concord	North Carolina	28027	South	OFF-PA-10002365	Office Supplies	Paper
14	14	CA-2016-161389	2016-12-05 00:00:00.0000000	2016-12-10 00:00:00.0000000	Standard Class	IM-15070	Irene Maddox	Consumer	United States	Seattle	Washington	98103	West	OFF-BI-10003656	Office Supplies	Binders
15	15	US-2015-118983	2015-11-22 00:00:00.0000000	2015-11-26 00:00:00.0000000	Standard Class	HP-14815	Harold Pawlan	Home Office	United States	Fort Worth	Texas	76106	Central	OFF-AP-10002311	Office Supplies	Appliance
16	16	US-2015-118983	2015-11-22 00:00:00.0000000	2015-11-26 00:00:00.0000000	Standard Class	HP-14815	Harold Pawlan	Home Office	United States	Fort Worth	Texas	76106	Central	OFF-BI-10000756	Office Supplies	Binders
17	17	CA-2014-105893	2014-11-11 00:00:00.0000000	2014-11-18 00:00:00.0000000	Standard Class	PK-19075	Pete Kriz	Consumer	United States	Madison	Wisconsin	53711	Central	OFF-ST-10004186	Office Supplies	Storage
18	18	CA-2014-167164	2014-05-13 00:00:00.0000000	2014-05-15 00:00:00.0000000	Second Class	AG-10270	Alejandro Grove	Consumer	United States	West Jordan	Utah	84084	West	OFF-ST-10000107	Office Supplies	Storage

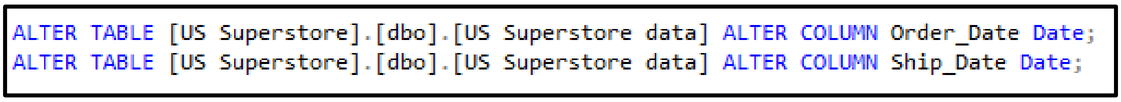
By taking the above table into consideration, we identify that some changes must be done.

- Order_Date and Shipe_Date variable should include only dates so time has to omitted.
- Row_ID, Order_ID, Customer_ID and Product_ID have to removed as they do not give any additional information for our analysis (we created later unique ID's references for these variables).
- Column of Country takes only the value 'United States'. We know that our Superstore is located in United States, so this column is unnecessary.

We created a new query so as to delete and transform our data with the commands shown below:



```
ALTER TABLE [US Superstore].[dbo].[US Superstore data] DROP COLUMN Row_ID;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] DROP COLUMN Order_ID;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] DROP COLUMN Customer_ID;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] DROP COLUMN Country;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] DROP COLUMN Product_ID;
```

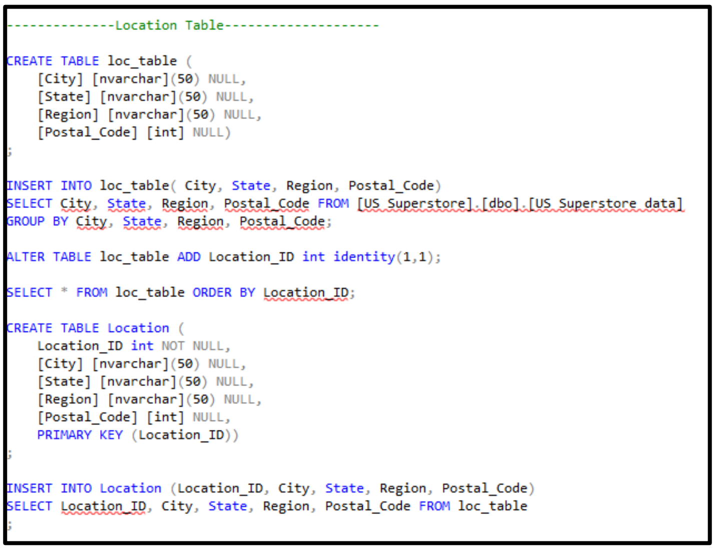


```
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ALTER COLUMN Order_Date Date;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ALTER COLUMN Ship_Date Date;
```

2. Dimensions and Fact Table

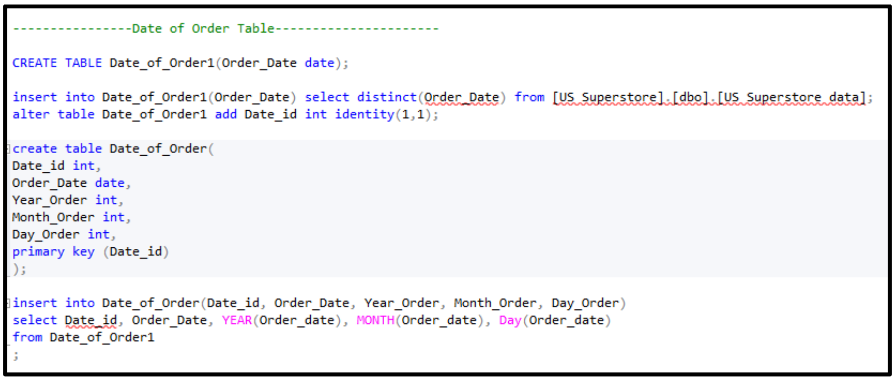
With the SQL Statements below, we created all the dimension tables, which refer to a collection of reference information about a measurable event. All dimension tables have been made in two stages. First, we created a table for each of these only with the attributes that we wanted to include (e.g. Loc_table has City, State, Region, Postal code). Next step was to fill these ‘first’ tables’ with values from ‘US Superstore’ table. In the example of loc_table we have grouped by City, State, Region and Postal Code in order to have all combination of attributed listed inside table. Secondly, we created the dimension tables by adding all the attributes again, plus with a unique ID (e.g. Location_ID) that we added and includes numbers from 1 and so on. Finally, we inserted values to dimension from the ‘first’ table’. ID’s are the primary keys of every dimension table.

Our final dimension tables are Location, Date_of_Order, Date_of_Ship, Ship_Mode_Table, Customer_Table and Category_Table.



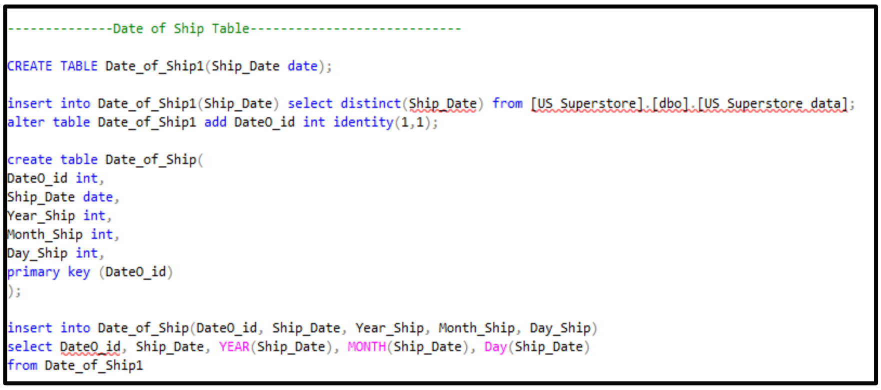
```sql
--------------Location Table-------------------

CREATE TABLE loc_table (
    [City] [nvarchar](50) NULL,
    [State] [nvarchar](50) NULL,
    [Region] [nvarchar](50) NULL,
    [Postal_Code] [int] NULL)
;

INSERT INTO loc_table( City, State, Region, Postal_Code)
SELECT City, State, Region, Postal_Code FROM [US Superstore].[dbo].[US Superstore data]
GROUP BY City, State, Region, Postal_Code;

ALTER TABLE loc_table ADD Location_ID int identity(1,1);

SELECT * FROM loc_table ORDER BY Location_ID;

CREATE TABLE Location (
    Location_ID int NOT NULL,
    [City] [nvarchar](50) NULL,
    [State] [nvarchar](50) NULL,
    [Region] [nvarchar](50) NULL,
    [Postal_Code] [int] NULL,
    PRIMARY KEY (Location_ID))
;

INSERT INTO Location (Location_ID, City, State, Region, Postal_Code)
SELECT Location_ID, City, State, Region, Postal_Code FROM loc_table
;
```

Dimension 1. Location Table



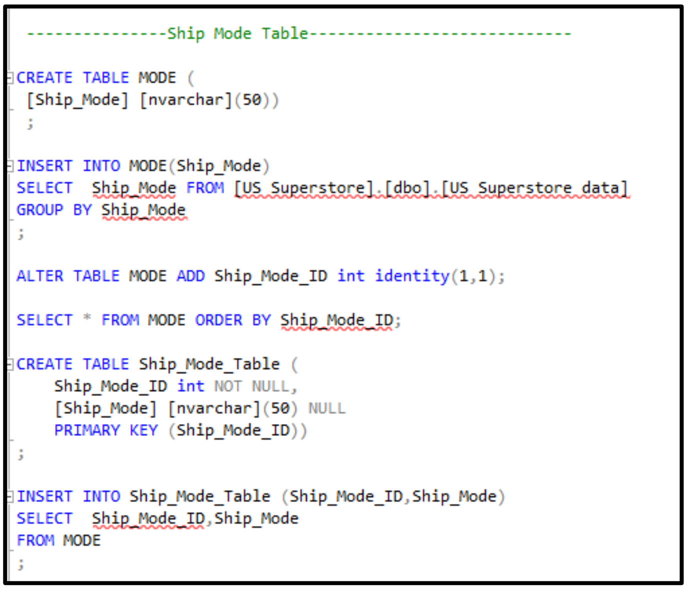
```sql
----------------Date of Order Table---------------------

CREATE TABLE Date_of_Order1(Order_Date date);

insert into Date_of_Order1(Order_Date) select distinct(Order_Date) from [US Superstore].[dbo].[US Superstore data];
alter table Date_of_Order1 add Date_id int identity(1,1);

create table Date_of_Order(
Date_id int,
Order_Date date,
Year_Order int,
Month_Order int,
Day_Order int,
primary key (Date_id)
);

insert into Date_of_Order(Date_id, Order_Date, Year_Order, Month_Order, Day_Order)
select Date_id, Order_Date, YEAR(Order_date), MONTH(Order_date), Day(Order_date)
from Date_of_Order1
;
```

Dimension 2. Date of Order Table



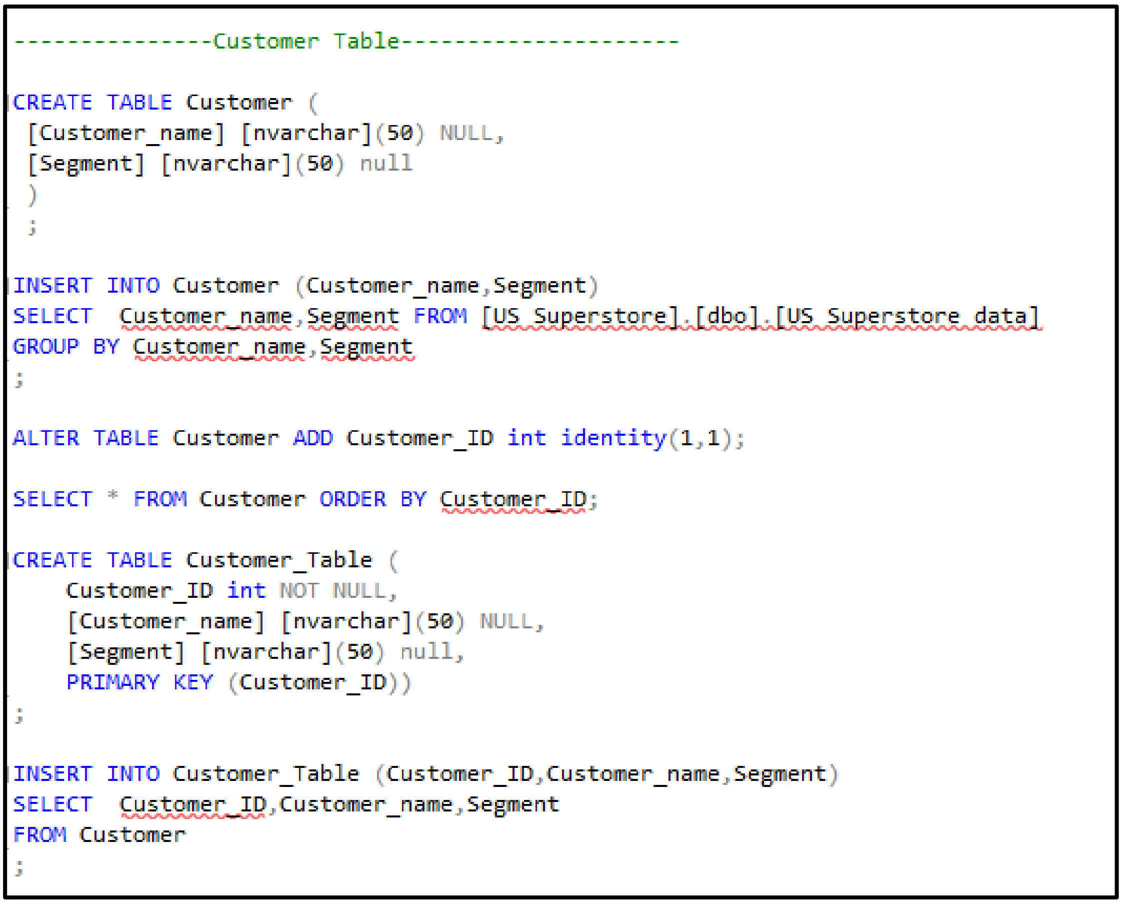
```
-------------Date of Ship Table---------------------------

CREATE TABLE Date_of_Ship1(Ship_Date date);

insert into Date_of_Ship1(Ship_Date) select distinct(Ship_Date) from [US Superstore].[dbo].[US Superstore data];
alter table Date_of_Ship1 add Date0_id int identity(1,1);

create table Date_of_Ship(
Date0_id int,
Ship_Date date,
Year_Ship int,
Month_Ship int,
Day_Ship int,
primary key (Date0_id)
);

insert into Date_of_Ship(Date0_id, Ship_Date, Year_Ship, Month_Ship, Day_Ship)
select Date0_id, Ship_Date, YEAR(Ship_Date), MONTH(Ship_Date), Day(Ship_Date)
from Date_of_Ship1
```

Dimension 3. Date of Ship Table



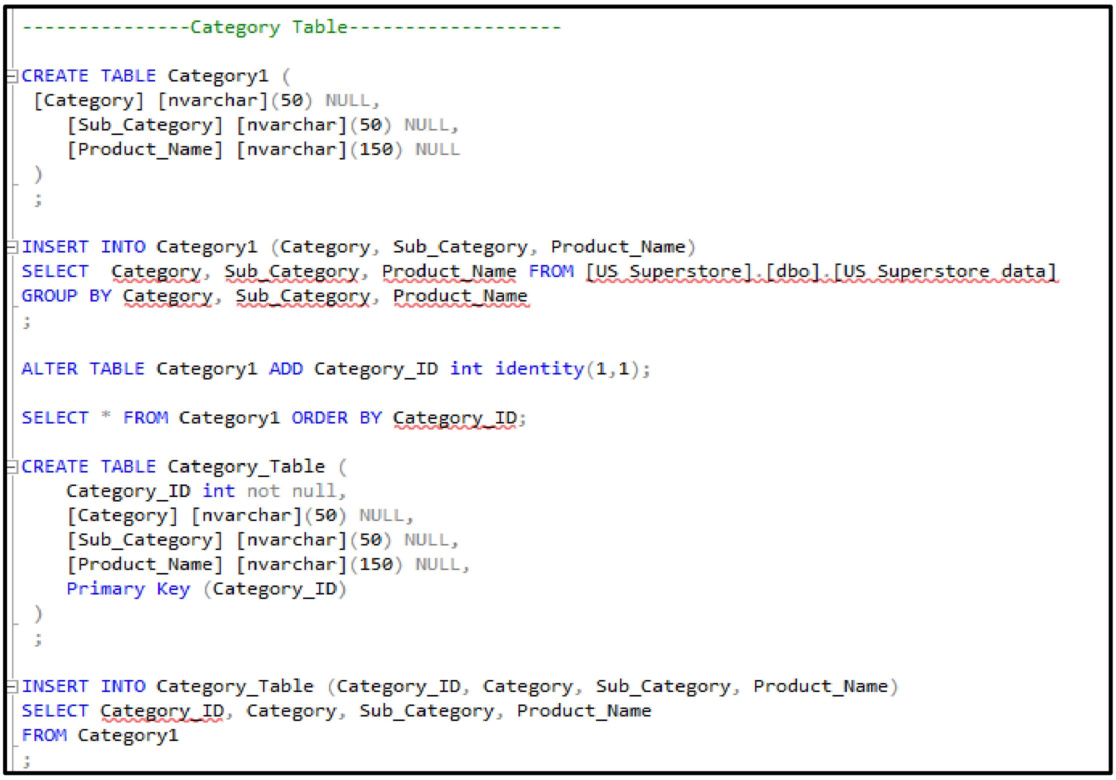
```
 ---------------Ship Mode Table----------------------------

CREATE TABLE MODE (
 [Ship_Mode] [nvarchar](50))
 ;

INSERT INTO MODE(Ship_Mode)
SELECT  Ship_Mode FROM [US Superstore].[dbo].[US Superstore data]
GROUP BY Ship_Mode
;

ALTER TABLE MODE ADD Ship_Mode_ID int identity(1,1);

SELECT * FROM MODE ORDER BY Ship_Mode_ID;

CREATE TABLE Ship_Mode_Table (
    Ship_Mode_ID int NOT NULL,
    [Ship_Mode] [nvarchar](50) NULL
    PRIMARY KEY (Ship_Mode_ID))
;

INSERT INTO Ship_Mode_Table (Ship_Mode_ID,Ship_Mode)
SELECT  Ship_Mode_ID,Ship_Mode
FROM MODE
;
```

Dimension 4. Ship Mode Table



```sql
---------------Customer Table---------------------

CREATE TABLE Customer (
 [Customer_name] [nvarchar](50) NULL,
 [Segment] [nvarchar](50) null
 )
 ;

INSERT INTO Customer (Customer_name,Segment)
SELECT  Customer_name,Segment FROM [US_Superstore].[dbo].[US_Superstore_data]
GROUP BY Customer_name,Segment
;

ALTER TABLE Customer ADD Customer_ID int identity(1,1);

SELECT * FROM Customer ORDER BY Customer_ID;

CREATE TABLE Customer_Table (
    Customer_ID int NOT NULL,
    [Customer_name] [nvarchar](50) NULL,
    [Segment] [nvarchar](50) null,
    PRIMARY KEY (Customer_ID))
;

INSERT INTO Customer_Table (Customer_ID,Customer_name,Segment)
SELECT  Customer_ID,Customer_name,Segment
FROM Customer
;
```

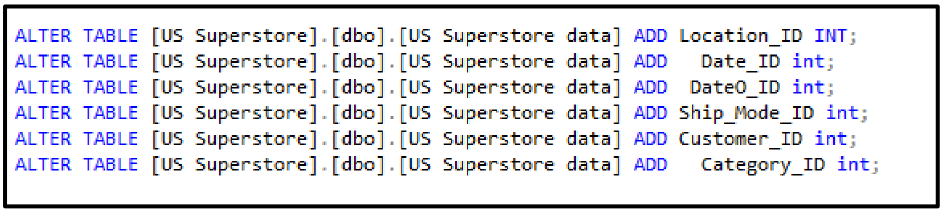
Dimension 5. Customer Table



```sql
--------------Category Table------------------

CREATE TABLE Category1 (
 [Category] [nvarchar](50) NULL,
    [Sub_Category] [nvarchar](50) NULL,
    [Product_Name] [nvarchar](150) NULL
 )
 ;

INSERT INTO Category1 (Category, Sub_Category, Product_Name)
SELECT  Category, Sub_Category, Product_Name FROM [US_Superstore].[dbo].[US_Superstore_data]
GROUP BY Category, Sub_Category, Product_Name
;

ALTER TABLE Category1 ADD Category_ID int identity(1,1);

SELECT * FROM Category1 ORDER BY Category_ID;

CREATE TABLE Category_Table (
    Category_ID int not null,
    [Category] [nvarchar](50) NULL,
    [Sub_Category] [nvarchar](50) NULL,
    [Product_Name] [nvarchar](150) NULL,
    Primary Key (Category_ID)
 )
 ;

INSERT INTO Category_Table (Category_ID, Category, Sub_Category, Product_Name)
SELECT Category_ID, Category, Sub_Category, Product_Name
FROM Category1
;
```

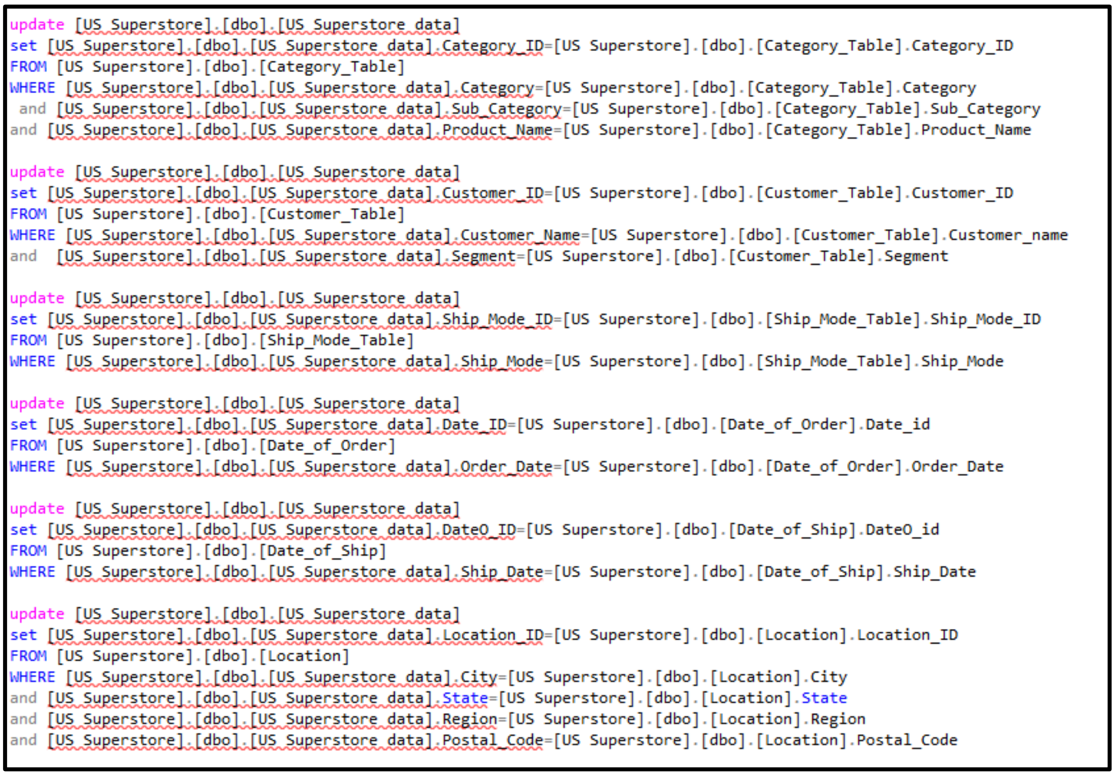
Dimension 6. Category Table

All primary keys from dimension tables were added in the main ‘US Superstore data’ table with the commands below.



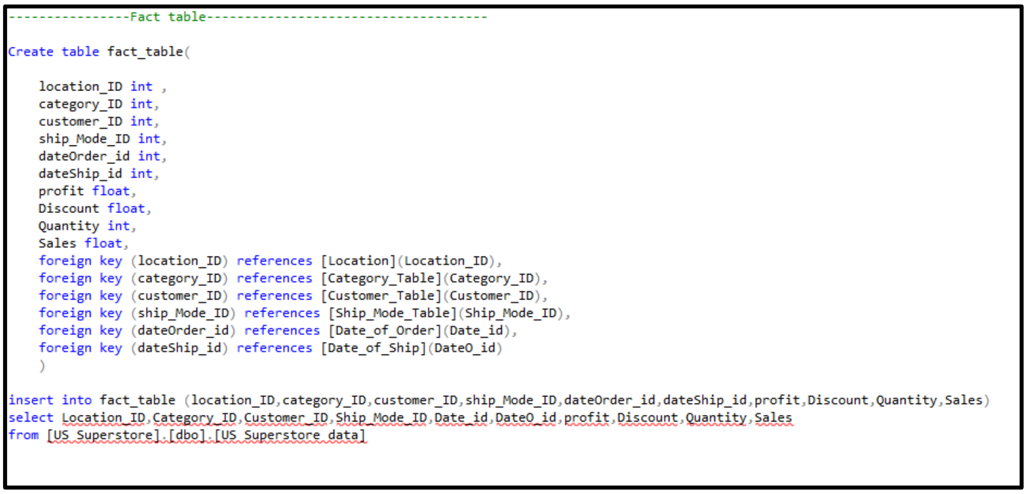
```
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ADD Location_ID INT;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ADD   Date_ID int;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ADD  Date0_ID int;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ADD Ship_Mode_ID int;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ADD Customer_ID int;
ALTER TABLE [US Superstore].[dbo].[US Superstore data] ADD   Category_ID int;
```

Prior to fact table’s creation, we inserted in the ‘US Superstore data’ table the values of the alternation of every dimension table that we did in the previous step.



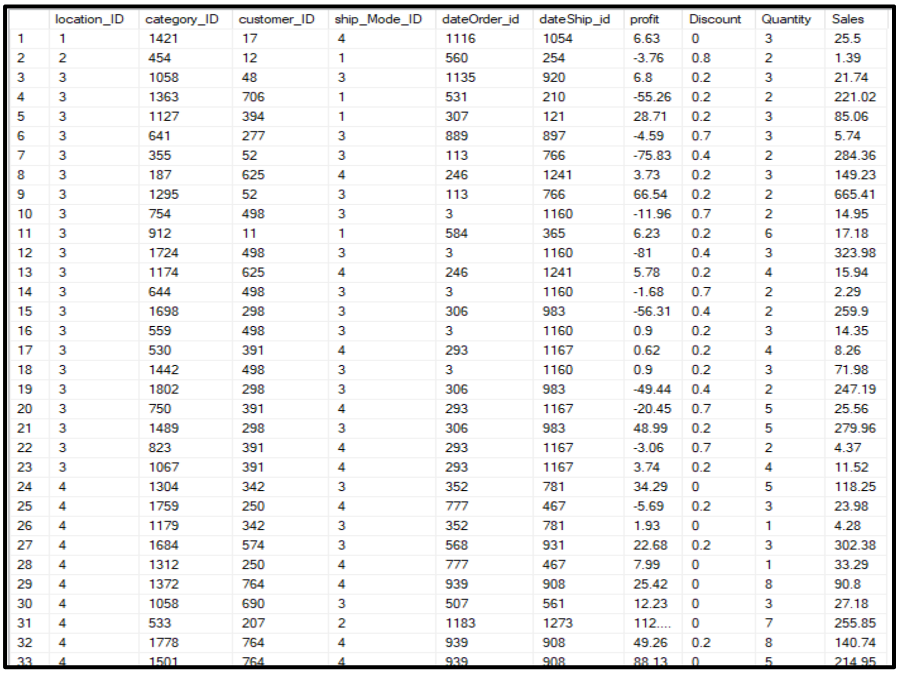
```
update [US Superstore].[dbo].[US Superstore data]
set [US Superstore].[dbo].[US Superstore data].Category_ID=[US Superstore].[dbo].[Category_Table].Category_ID
FROM [US Superstore].[dbo].[Category_Table]
WHERE [US Superstore].[dbo].[US Superstore data].Category=[US Superstore].[dbo].[Category_Table].Category
 and [US Superstore].[dbo].[US Superstore data].Sub_Category=[US Superstore].[dbo].[Category_Table].Sub_Category
and [US Superstore].[dbo].[US Superstore data].Product_Name=[US Superstore].[dbo].[Category_Table].Product_Name

update [US Superstore].[dbo].[US Superstore data]
set [US Superstore].[dbo].[US Superstore data].Customer_ID=[US Superstore].[dbo].[Customer_Table].Customer_ID
FROM [US Superstore].[dbo].[Customer_Table]
WHERE [US Superstore].[dbo].[US Superstore data].Customer_Name=[US Superstore].[dbo].[Customer_Table].Customer_name
and  [US Superstore].[dbo].[US Superstore data].Segment=[US Superstore].[dbo].[Customer_Table].Segment

update [US Superstore].[dbo].[US Superstore data]
set [US Superstore].[dbo].[US Superstore data].Ship_Mode_ID=[US Superstore].[dbo].[Ship_Mode_Table].Ship_Mode_ID
FROM [US Superstore].[dbo].[Ship_Mode_Table]
WHERE [US Superstore].[dbo].[US Superstore data].Ship_Mode=[US Superstore].[dbo].[Ship_Mode_Table].Ship_Mode

update [US Superstore].[dbo].[US Superstore data]
set [US Superstore].[dbo].[US Superstore data].Date_ID=[US Superstore].[dbo].[Date_of_Order].Date_id
FROM [US Superstore].[dbo].[Date_of_Order]
WHERE [US Superstore].[dbo].[US Superstore data].Order_Date=[US Superstore].[dbo].[Date_of_Order].Order_Date

update [US Superstore].[dbo].[US Superstore data]
set [US Superstore].[dbo].[US Superstore data].Date0_ID=[US Superstore].[dbo].[Date_of_Ship].Date0_id
FROM [US Superstore].[dbo].[Date_of_Ship]
WHERE [US Superstore].[dbo].[US Superstore data].Ship_Date=[US Superstore].[dbo].[Date_of_Ship].Ship_Date

update [US Superstore].[dbo].[US Superstore data]
set [US Superstore].[dbo].[US Superstore data].Location_ID=[US Superstore].[dbo].[Location].Location_ID
FROM [US Superstore].[dbo].[Location]
WHERE [US Superstore].[dbo].[US Superstore data].City=[US Superstore].[dbo].[Location].City
and [US Superstore].[dbo].[US Superstore data].State=[US Superstore].[dbo].[Location].State
and [US Superstore].[dbo].[US Superstore data].Region=[US Superstore].[dbo].[Location].Region
and [US Superstore].[dbo].[US Superstore data].Postal_Code=[US Superstore].[dbo].[Location].Postal_Code
```

Finally, we created the fact_table which contains all numerical variables (profit, discount, quantity and sales), ID's (location_ID, category_ID, customer_ID, Ship_Mode_id, DateOrder_id and dateShip_id) and foreign keys that refer to every dimension table. Last but not least, we inserted the values from the main table 'US Superstore data'. A brief preview of fact_table is shown in Table 8.



```
---------------Fact table-----------------------------------

Create table fact_table(

    location_ID int ,
    category_ID int,
    customer_ID int,
    ship_Mode_ID int,
    dateOrder_id int,
    dateShip_id int,
    profit float,
    Discount float,
    Quantity int,
    Sales float,
    foreign key (location_ID) references [Location](Location_ID),
    foreign key (category_ID) references [Category_Table](Category_ID),
    foreign key (customer_ID) references [Customer_Table](Customer_ID),
    foreign key (ship_Mode_ID) references [Ship_Mode_Table](Ship_Mode_ID),
    foreign key (dateOrder_id) references [Date_of_Order](Date_id),
    foreign key (dateShip_id) references [Date_of_Ship](DateO_id)
    )

insert into fact_table (location_ID,category_ID,customer_ID,ship_Mode_ID,dateOrder_id,dateShip_id,profit,Discount,Quantity,Sales)
select Location_ID,Category_ID,Customer_ID,Ship_Mode_ID,Date_id,DateO_id,profit,Discount,Quantity,Sales
from [US Superstore].[dbo].[US Superstore data]
```

7. Fact Table



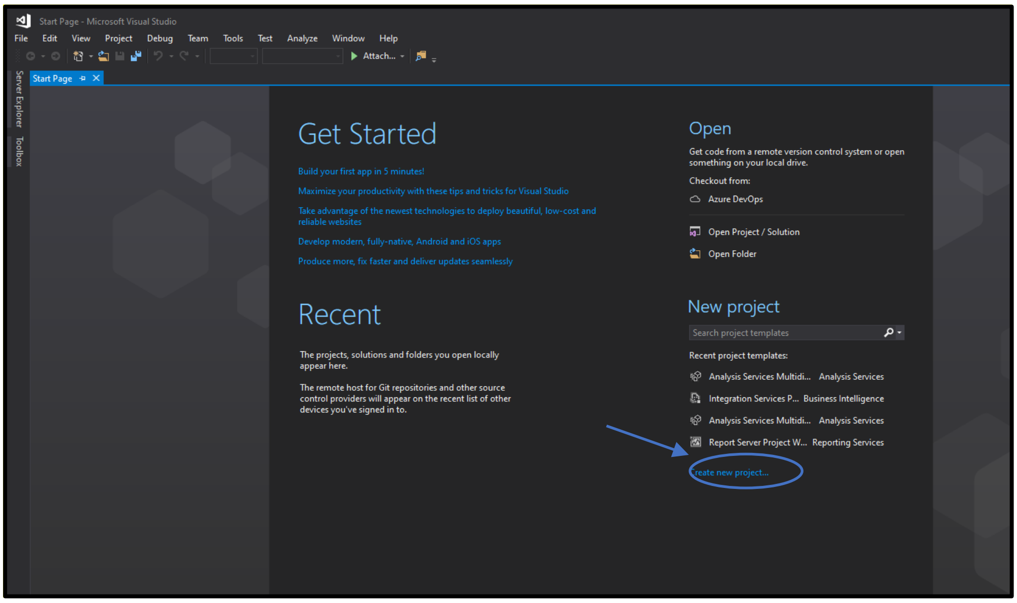
	location_ID	category_ID	customer_ID	ship_Mode_ID	dateOrder_id	dateShip_id	profit	Discount	Quantity	Sales
1	1	1421	17	4	1116	1054	6.63	0	3	25.5
2	2	454	12	1	560	254	-3.76	0.8	2	1.39
3	3	1058	48	3	1135	920	6.8	0.2	3	21.74
4	3	1363	706	1	531	210	-55.26	0.2	2	221.02
5	3	1127	394	1	307	121	28.71	0.2	3	85.06
6	3	641	277	3	889	897	-4.59	0.7	3	5.74
7	3	355	52	3	113	766	-75.83	0.4	2	284.36
8	3	187	625	4	246	1241	3.73	0.2	3	149.23
9	3	1295	52	3	113	766	66.54	0.2	2	665.41
10	3	754	498	3	3	1160	-11.96	0.7	2	14.95
11	3	912	11	1	584	365	6.23	0.2	6	17.18
12	3	1724	498	3	3	1160	-81	0.4	3	323.98
13	3	1174	625	4	246	1241	5.78	0.2	4	15.94
14	3	644	498	3	3	1160	-1.68	0.7	2	2.29
15	3	1698	298	3	306	983	-56.31	0.4	2	259.9
16	3	559	498	3	3	1160	0.9	0.2	3	14.35
17	3	530	391	4	293	1167	0.62	0.2	4	8.26
18	3	1442	498	3	3	1160	0.9	0.2	3	71.98
19	3	1802	298	3	306	983	-49.44	0.4	2	247.19
20	3	750	391	4	293	1167	-20.45	0.7	5	25.56
21	3	1489	298	3	306	983	48.99	0.2	5	279.96
22	3	823	391	4	293	1167	-3.06	0.7	2	4.37
23	3	1067	391	4	293	1167	3.74	0.2	4	11.52
24	4	1304	342	3	352	781	34.29	0	5	118.25
25	4	1759	250	4	777	467	-5.69	0.2	3	23.98
26	4	1179	342	3	352	781	1.93	0	1	4.28
27	4	1684	574	3	568	931	22.68	0.2	3	302.38
28	4	1312	250	4	777	467	7.99	0	1	33.29
29	4	1372	764	4	939	908	25.42	0	8	90.8
30	4	1058	690	3	507	561	12.23	0	3	27.18
31	4	533	207	2	1183	1273	112....	0	7	255.85
32	4	1778	764	4	939	908	49.26	0.2	8	140.74
33	4	1501	764	4	939	908	88.13	0	5	214.95

8. Preview of Fact Table

3. Cube Creation using Visual Studio

Online analytical processing (OLAP) cubes are a feature in Service Manager that use the existing data warehouse infrastructure to provide self-service business intelligence capabilities to end users. An **OLAP cube** is a data structure that overcomes the limitations of relational databases by providing rapid analysis of data. Cubes can display and sum large amounts of data while also providing users with searchable access to any data points. This way, the data can be rolled up, drilled down, sliced and diced as needed to handle the widest variety of questions that are relevant to a user's area of interest. These cubes are stored in SQL Server Analysis Services (SSAS) and you can use them to analyze the data from multiple perspectives.

For the purpose of our analysis, we created an analysis services multidimensional project. Firstly, we choose the data source as shown below.


Start Page - Microsoft Visual Studio
File
Edit
View
Project
Debug
Team
Tools
Test
Analyze
Window
Help
Attach...
Server Explorer
Toolbox
Start Page
Get Started
Open
Recent
New project
Search project templates
Checkout from:
Azure DevOps
Open Project / Solution
Open Folder
Recent project templates:
Analysis Services Multidi... Analysis Services
Integration Services P... Business Intelligence
Analysis Services Multidi... Analysis Services
Report Server Project W... Reporting Services
Create new project...

Build your first app in 5 minutes!

Maximize your productivity with these tips and tricks for Visual Studio

Take advantage of the newest technologies to deploy beautiful, low-cost and reliable websites

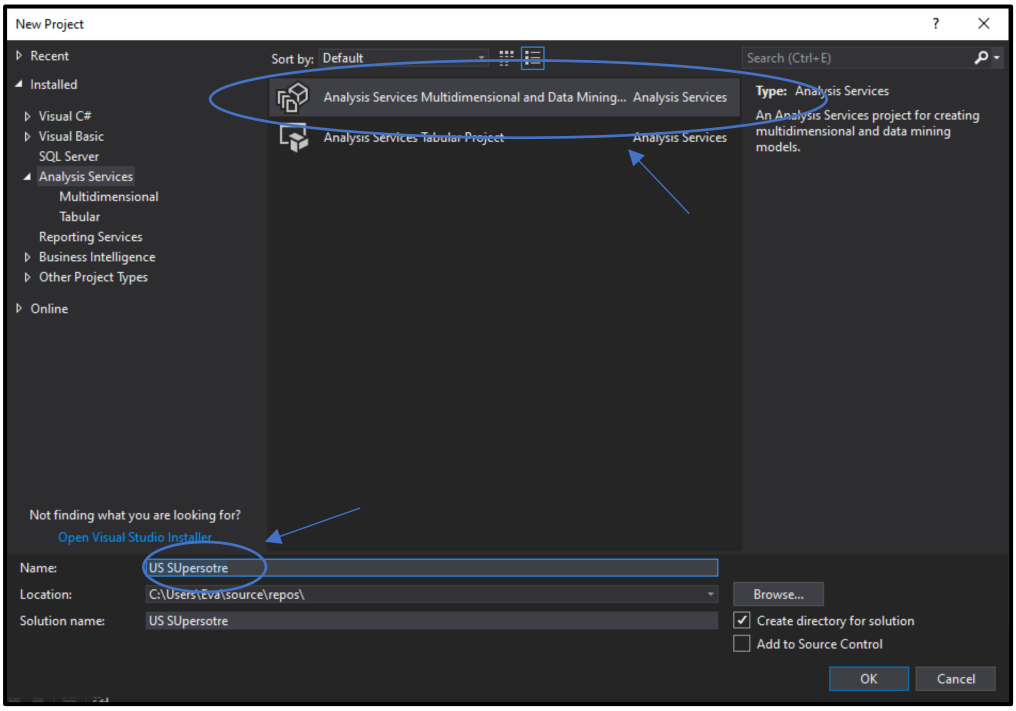
Develop modern, fully-native, Android and iOS apps

Produce more, fix faster and deliver updates seamlessly

Get code from a remote version control system or open something on your local drive.

The projects, solutions and folders you open locally appear here.

The remote host for Git repositories and other source control providers will appear on the recent list of other devices you've signed in to.



Analysis Services Multidimensional and Data Mining... Analysis Services

Analysis Services Tabular Project Analysis Services

Type: Analysis Services

An Analysis Services project for creating multidimensional and data mining models.

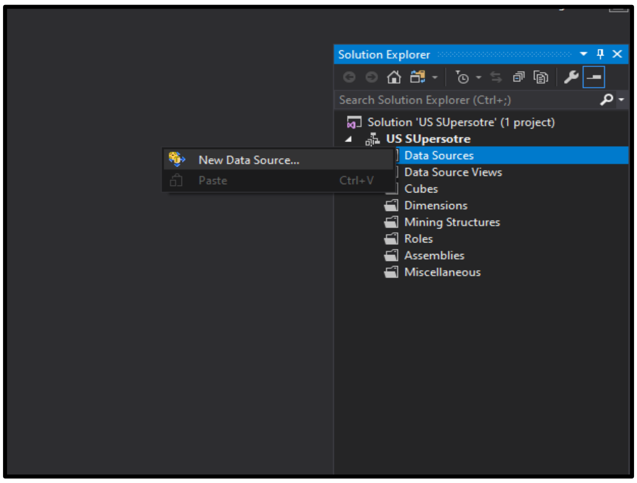
Name: US SUpersotre

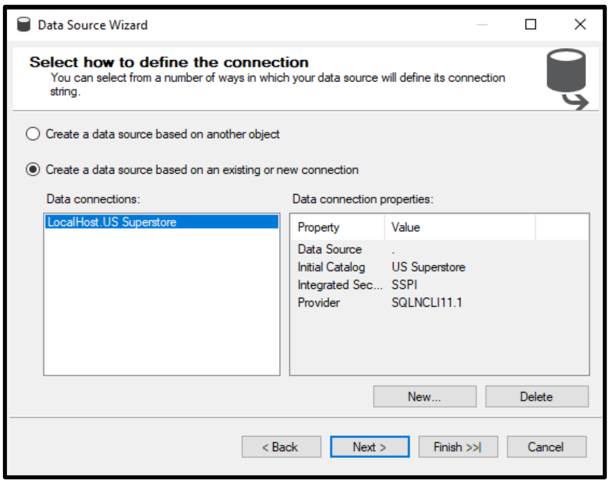
Location: C:\Users\Eva\source\repos\

Solution name: US SUpersotre

☑ Create directory for solution

□ Add to Source Control





Select how to define the connection

You can select from a number of ways in which your data source will define its connection string.

○ Create a data source based on another object

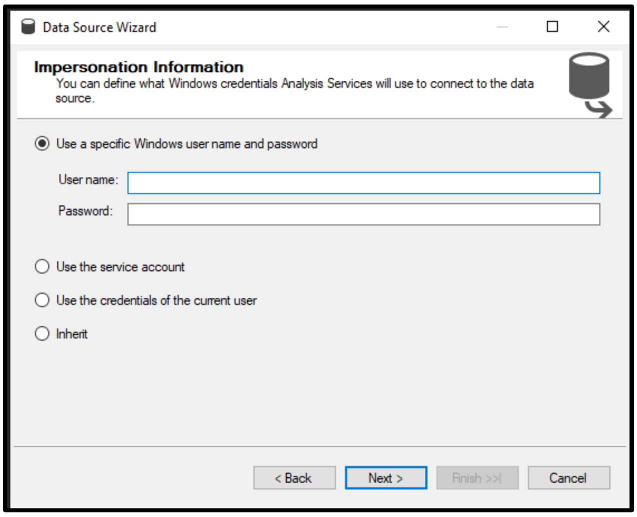
● Create a data source based on an existing or new connection

Data connections:

LocalHost.US Superstore

Data connection properties:

Property	Value
Data Source	.
Initial Catalog	US Superstore
Integrated Sec...	SSPI
Provider	SQLNCLI11.1



Impersonation Information

You can define what Windows credentials Analysis Services will use to connect to the data source.

● Use a specific Windows user name and password

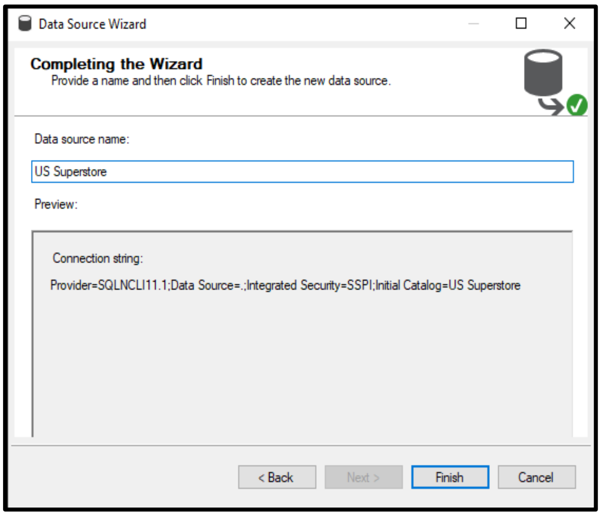
User name:

Password:

○ Use the service account

○ Use the credentials of the current user

○ Inherit



Completing the Wizard

Provide a name and then click Finish to create the new data source.

Data source name:

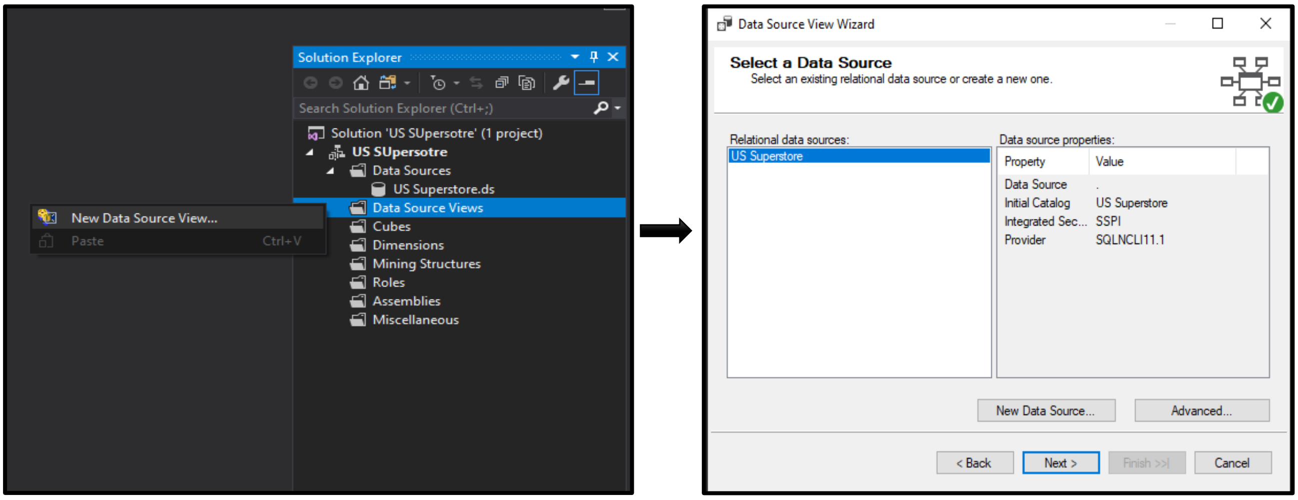
US Superstore

Preview:

Connection string:

Provider=SQLNCLI11.1;Data Source=.;Integrated Security=SSPI;Initial Catalog=US Superstore

After, we had to create a new data source view and transfer our fact table and its related dimensions in the data source views.



Select a Data Source

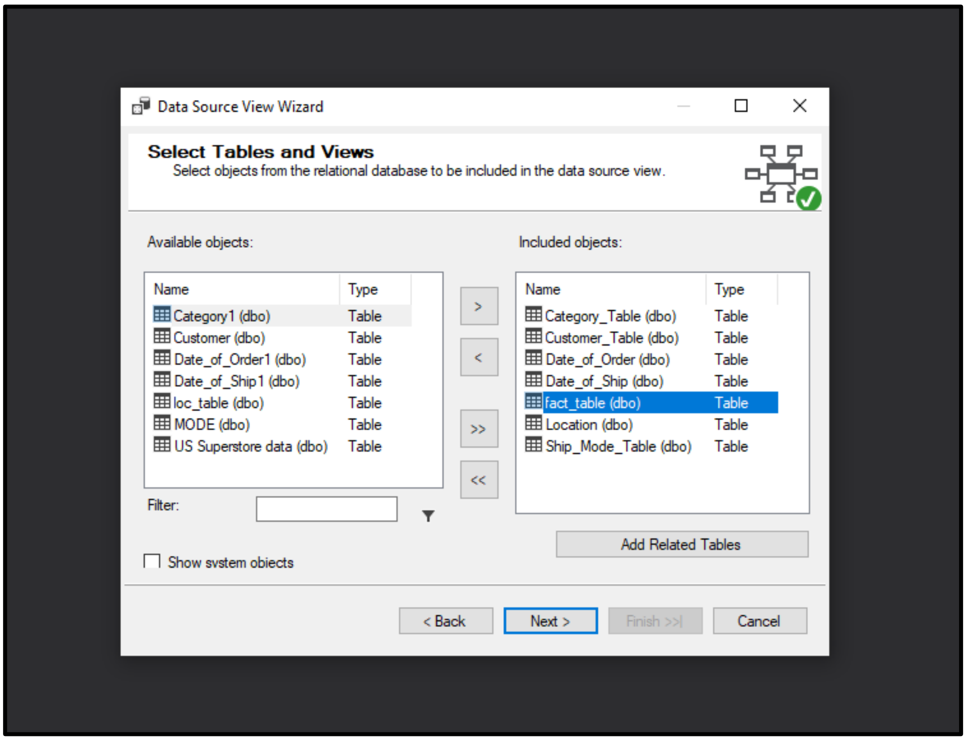
Select an existing relational data source or create a new one.

Relational data sources:

US Superstore

Data source properties:

Property	Value
Data Source	.
Initial Catalog	US Superstore
Integrated Sec...	SSPI
Provider	SQLNCLI11.1



Select Tables and Views

Select objects from the relational database to be included in the data source view.

Available objects:

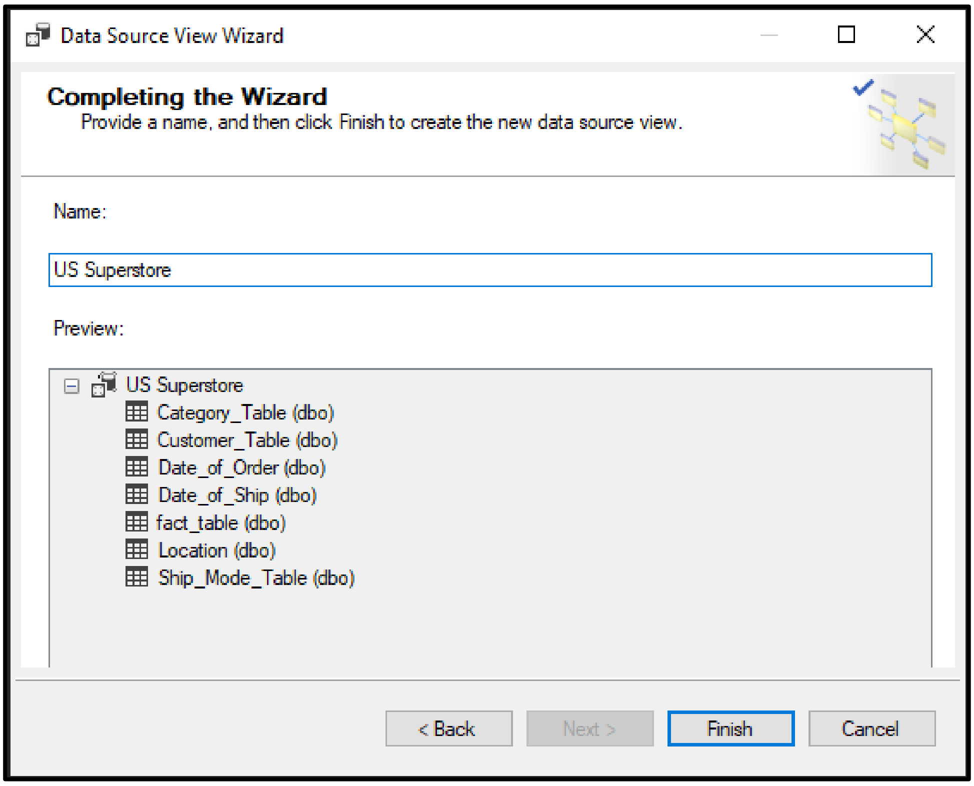
Name	Type
Category1 (dbo)	Table
Customer (dbo)	Table
Date_of_Order1 (dbo)	Table
Date_of_Ship1 (dbo)	Table
loc_table (dbo)	Table
MODE (dbo)	Table
US Superstore data (dbo)	Table

Included objects:

Name	Type
Category_Table (dbo)	Table
Customer_Table (dbo)	Table
Date_of_Order (dbo)	Table
Date_of_Ship (dbo)	Table
fact_table (dbo)	Table
Location (dbo)	Table
Ship_Mode_Table (dbo)	Table

Filter:

□ Show system objects



Data Source View Wizard

Completing the Wizard

Provide a name, and then click Finish to create the new data source view.

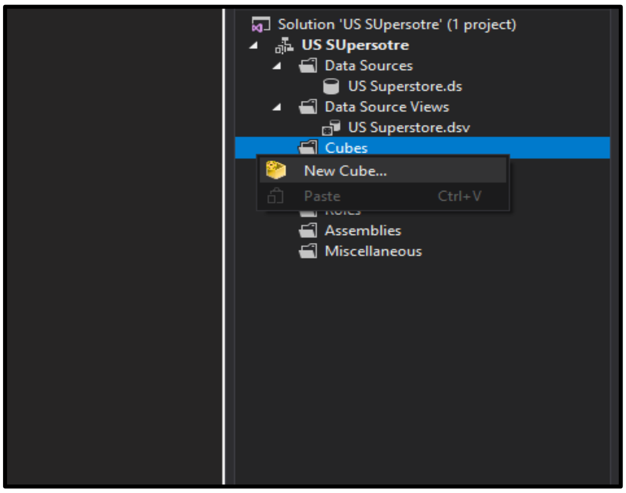
Name:

US Superstore

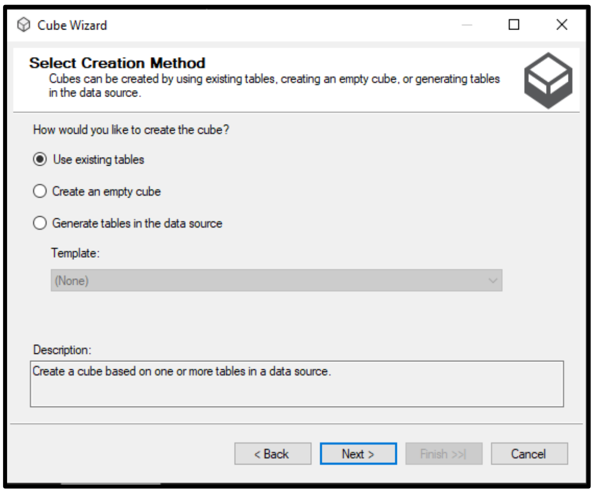
Preview:

```
⊟ US Superstore
    Category_Table (dbo)
    Customer_Table (dbo)
    Date_of_Order (dbo)
    Date_of_Ship (dbo)
    fact_table (dbo)
    Location (dbo)
    Ship_Mode_Table (dbo)
```

We continued with the creation of the cube:



```
Solution 'US SUpersotre' (1 project)
◢ US SUpersotre
   ◢ Data Sources
        US Superstore.ds
   ◢ Data Source Views
        US Superstore.dsv
     Cubes
```



Cube Wizard

Select Creation Method

Cubes can be created by using existing tables, creating an empty cube, or generating tables in the data source.

How would you like to create the cube?

● Use existing tables

○ Create an empty cube

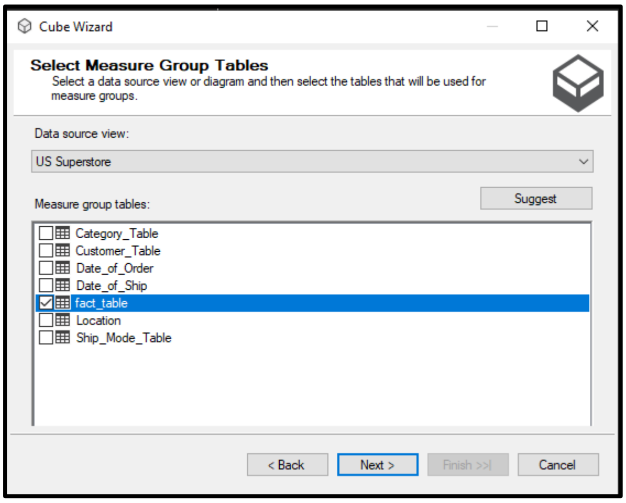
○ Generate tables in the data source

Template:

(None)

Description:

Create a cube based on one or more tables in a data source.



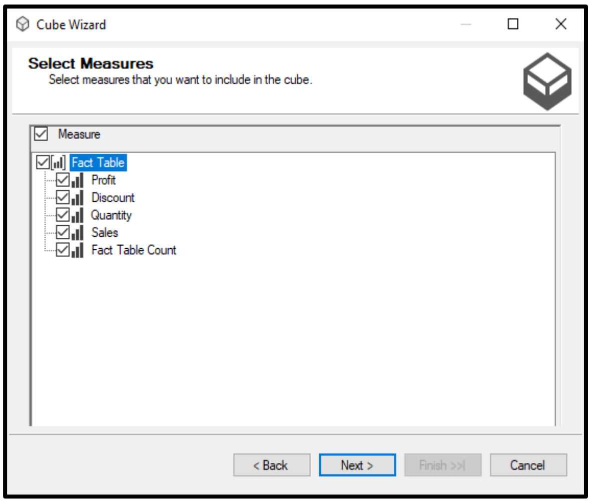
Cube Wizard

Select Measure Group Tables
Select a data source view or diagram and then select the tables that will be used for measure groups.

Data source view: US Superstore

Measure group tables:

- ☐ Category_Table
- ☐ Customer_Table
- ☐ Date_of_Order
- ☐ Date_of_Ship
- ☑ fact_table
- ☐ Location
- ☐ Ship_Mode_Table

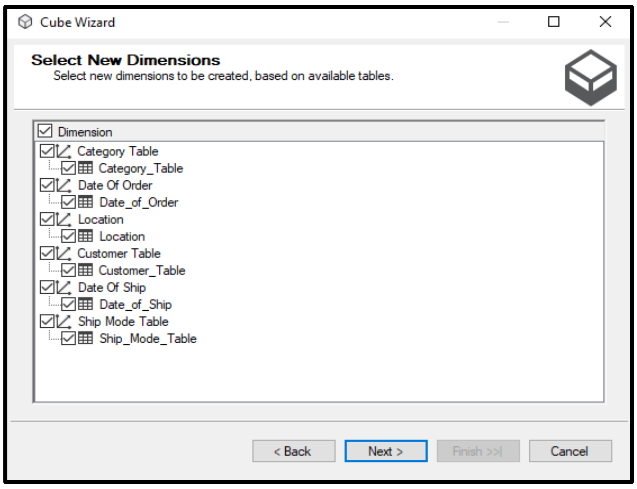


Cube Wizard

Select Measures
Select measures that you want to include in the cube.

- ☑ Measure
  - ☑ Fact Table
    - ☑ Profit
    - ☑ Discount
    - ☑ Quantity
    - ☑ Sales
    - ☑ Fact Table Count

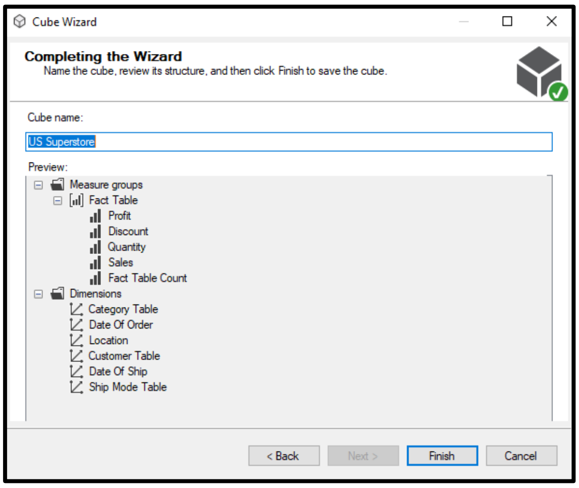
After the choice of the fact table, dimensions were added:



Cube Wizard

Select New Dimensions
Select new dimensions to be created, based on available tables.

- ☑ Dimension
  - ☑ Category Table
    - ☑ Category_Table
  - ☑ Date Of Order
    - ☑ Date_of_Order
  - ☑ Location
    - ☑ Location
  - ☑ Customer Table
    - ☑ Customer_Table
  - ☑ Date Of Ship
    - ☑ Date_of_Ship
  - ☑ Ship Mode Table
    - ☑ Ship_Mode_Table



Cube Wizard

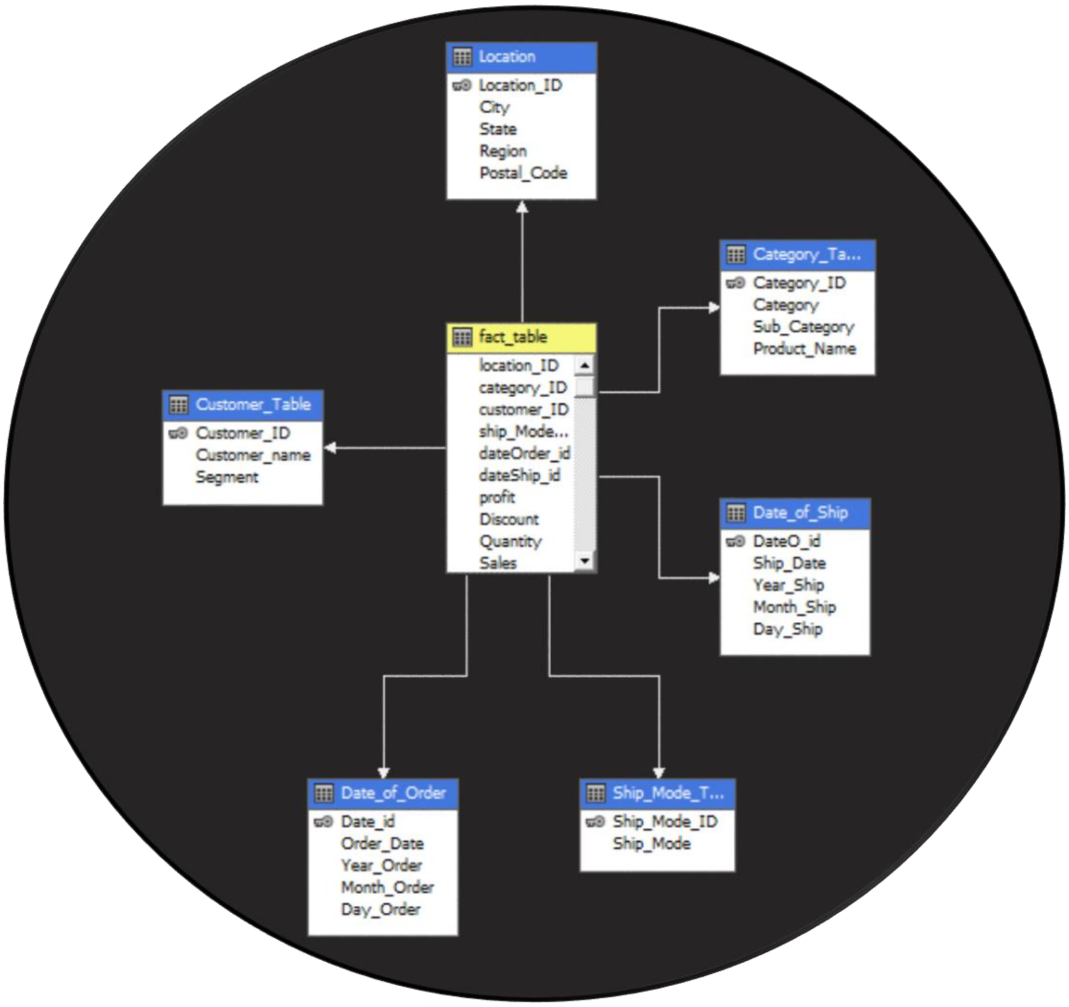
Completing the Wizard
Name the cube, review its structure, and then click Finish to save the cube.

Cube name: US Superstore

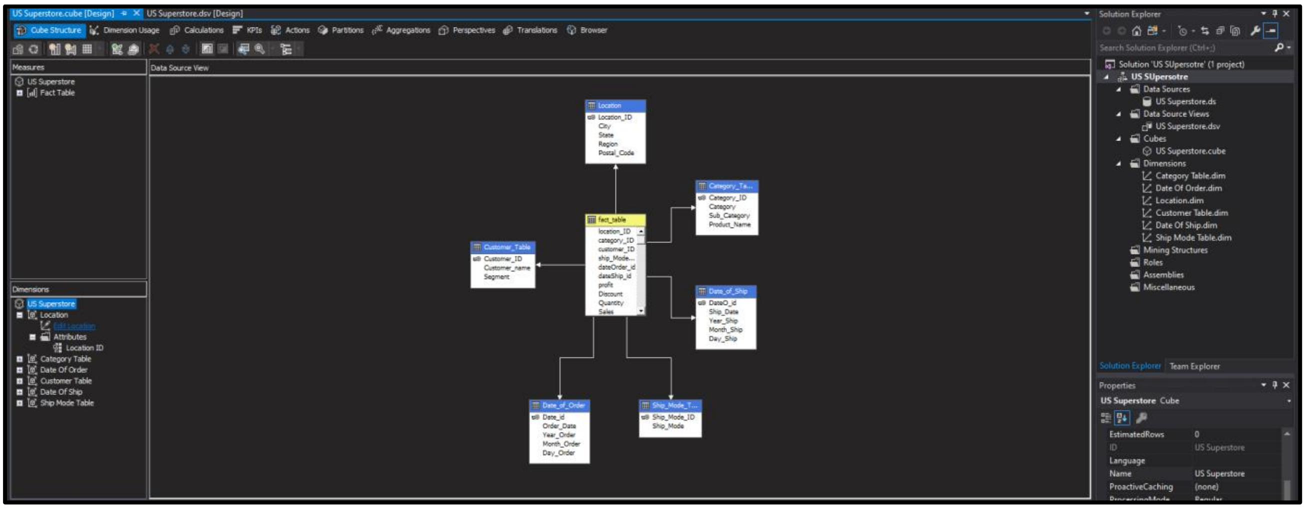
Preview:

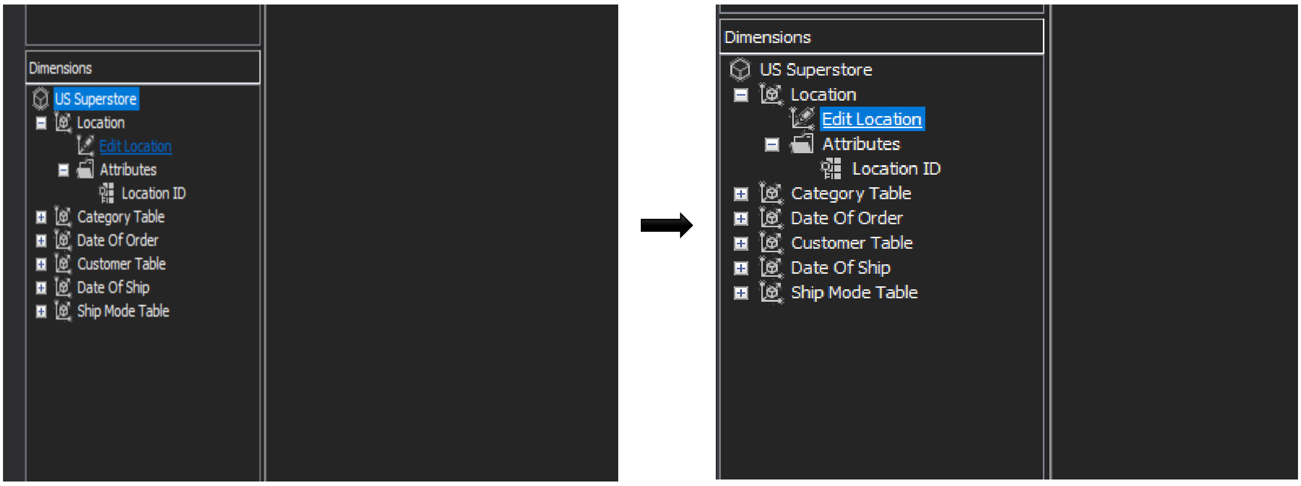
- Measure groups
  - Fact Table
    - Profit
    - Discount
    - Quantity
    - Sales
    - Fact Table Count
- Dimensions
  - Category Table
  - Date Of Order
  - Location
  - Customer Table
  - Date Of Ship
  - Ship Mode Table

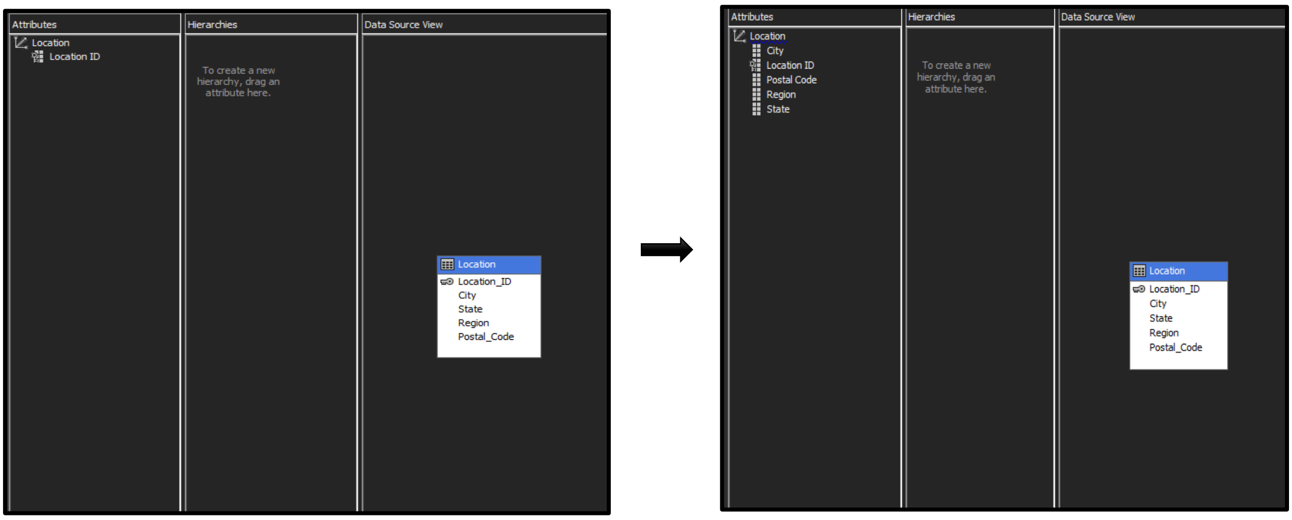
Star – Schema


Location
Location_ID
City
State
Region
Postal_Code
Category_Ta...
Category_ID
Category
Sub_Category
Product_Name
fact_table
location_ID
category_ID
customer_ID
ship_Mode...
dateOrder_id
dateShip_id
profit
Discount
Quantity
Sales
Customer_Table
Customer_ID
Customer_name
Segment
Date_of_Ship
DateO_id
Ship_Date
Year_Ship
Month_Ship
Day_Ship
Date_of_Order
Date_id
Order_Date
Year_Order
Month_Order
Day_Order
Ship_Mode_T...
Ship_Mode_ID
Ship_Mode

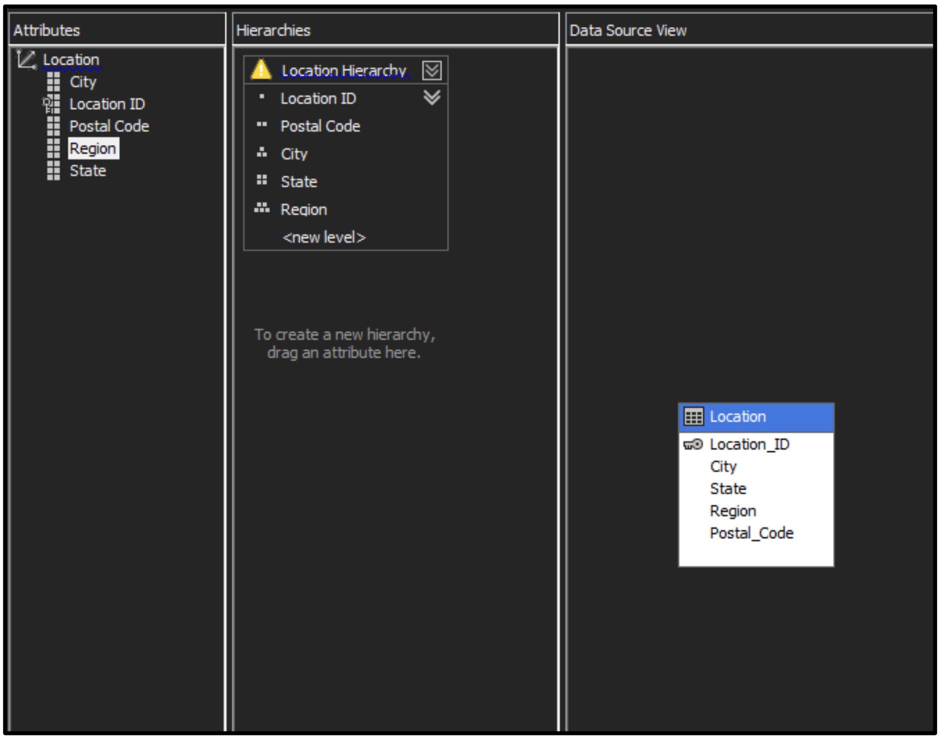
The next step was to edit all the attributes in each dimension tables (only ID’s were included) as shown below.

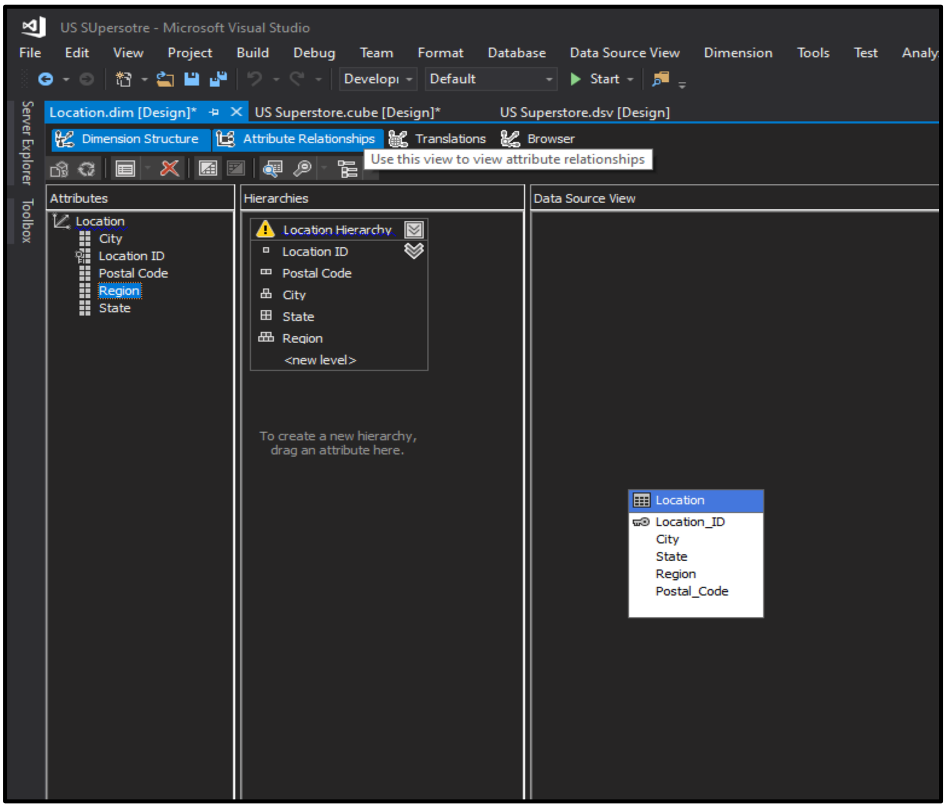


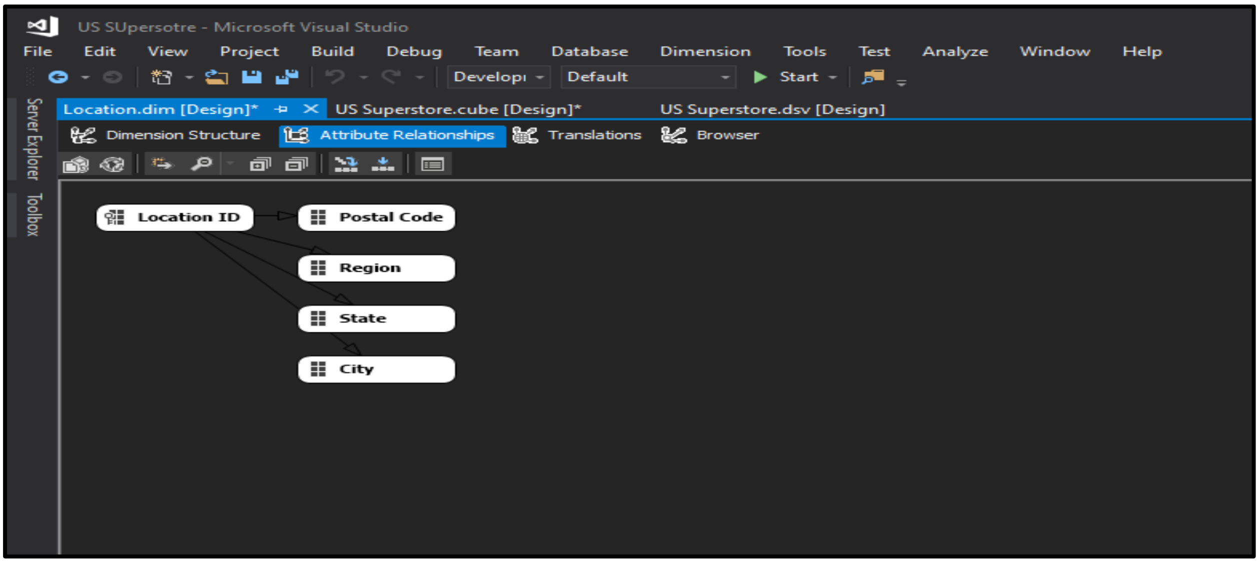


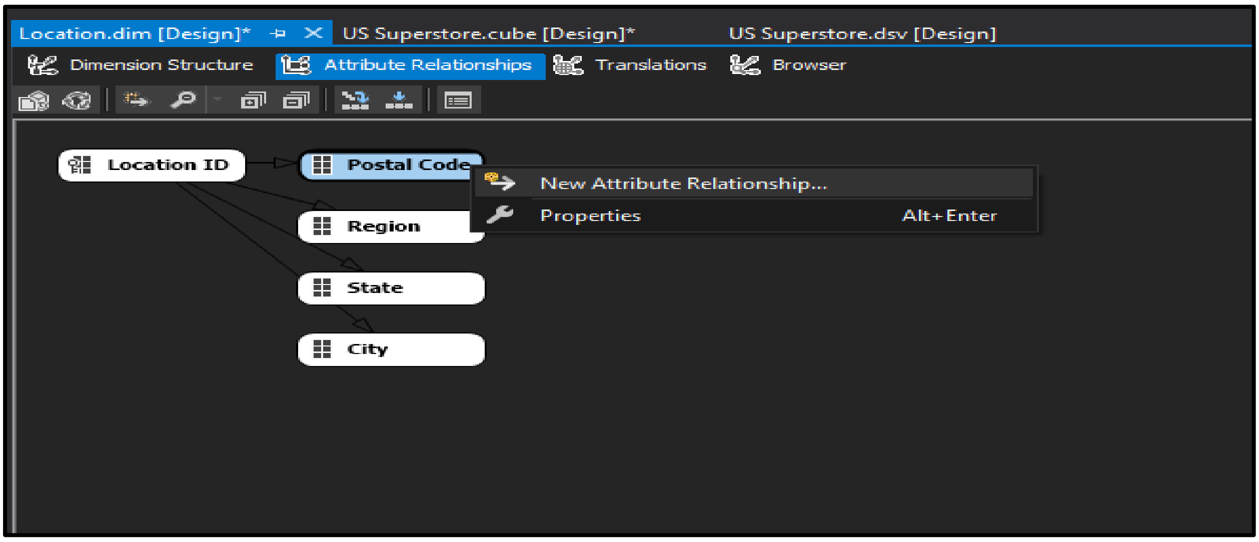


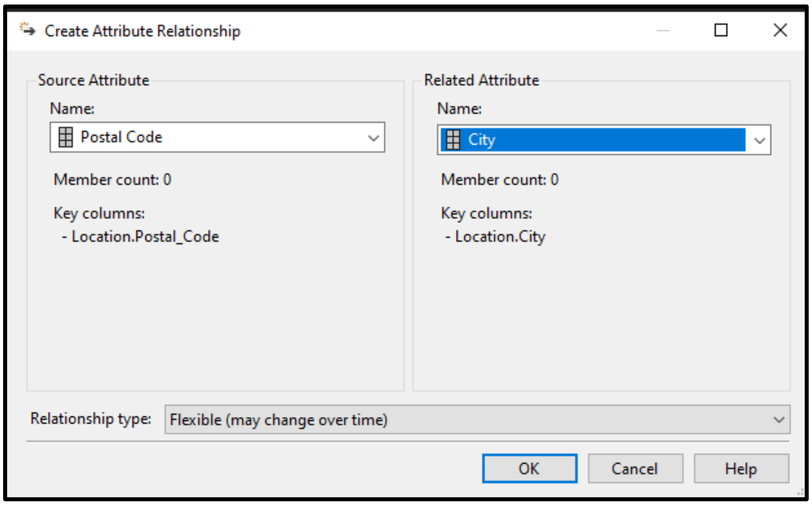
Hierarchy in the dimension "Location" has also to be defined as follows:






US SUpersotre - Microsoft Visual Studio
File Edit View Project Build Debug Team Database Dimension Tools Test Analyze Window Help
Developı Default Start
Location.dim [Design]* US Superstore.cube [Design]* US Superstore.dsv [Design]
Dimension Structure Attribute Relationships Translations Browser
Server Explorer Toolbox
Location ID
Postal Code
Region
State
City


Location.dim [Design]* US Superstore.cube [Design]* US Superstore.dsv [Design]
Dimension Structure Attribute Relationships Translations Browser
Location ID
Postal Code
Region
State
City
New Attribute Relationship...
Properties Alt+Enter



Source Attribute

Name: Postal Code

Member count: 0

Key columns:
- Location.Postal_Code

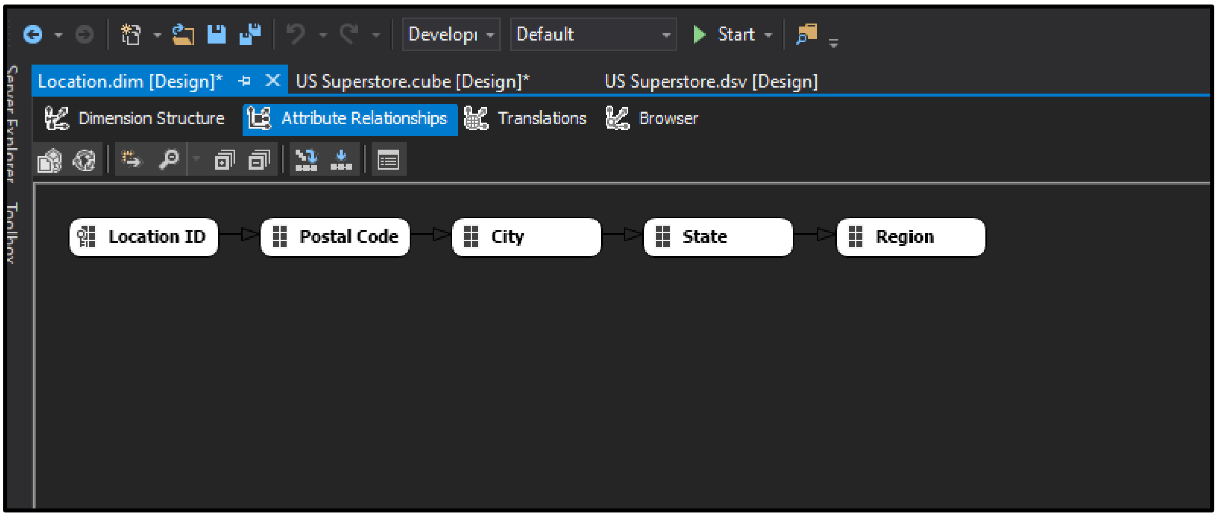
Related Attribute

Name: City

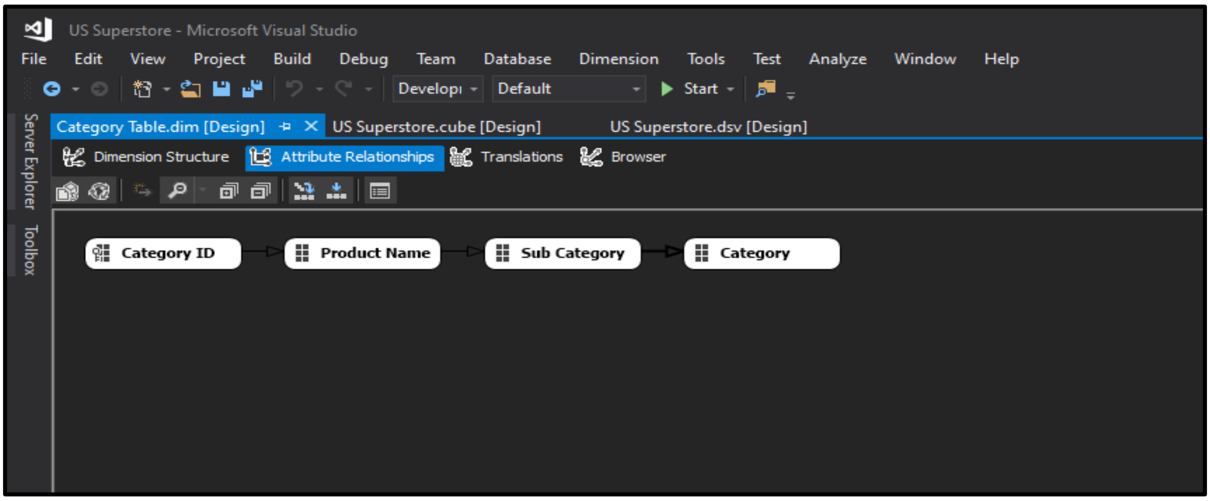
Member count: 0

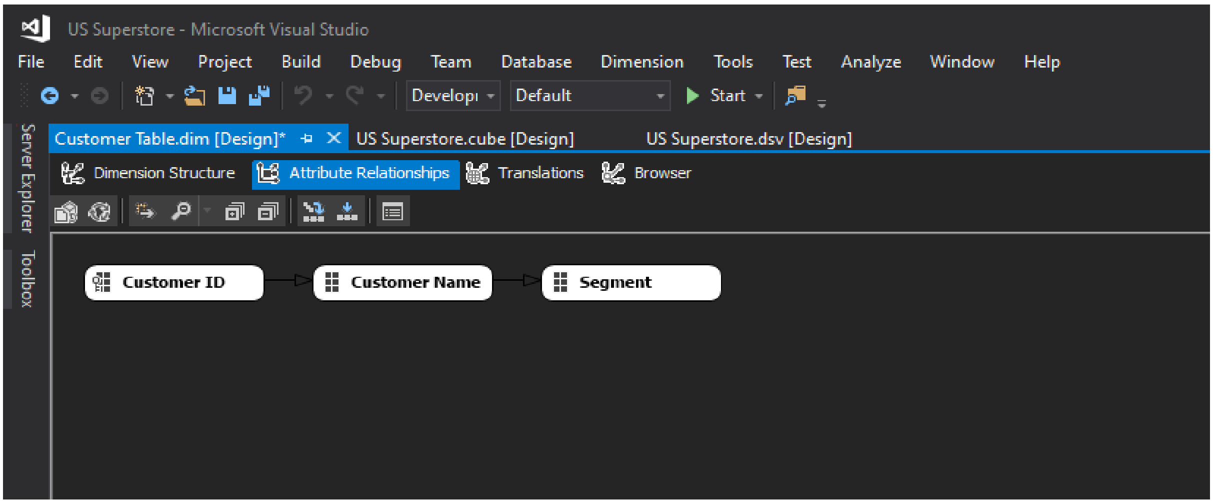
Key columns:
- Location.City

Relationship type: Flexible (may change over time)

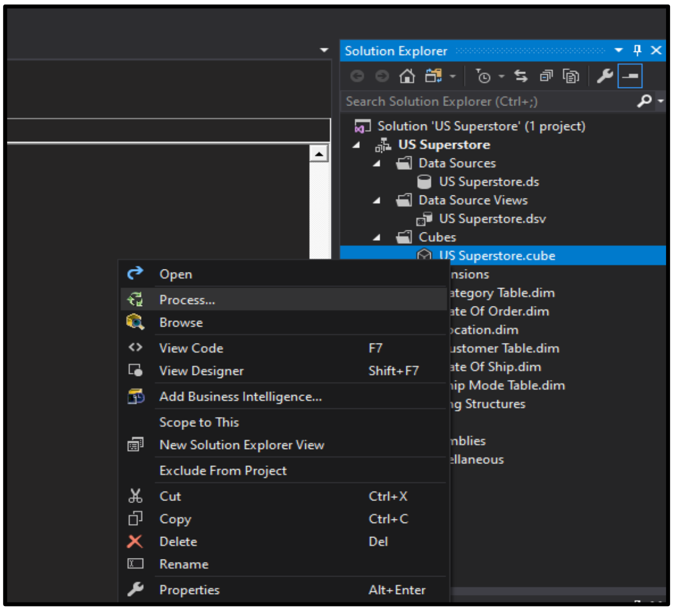


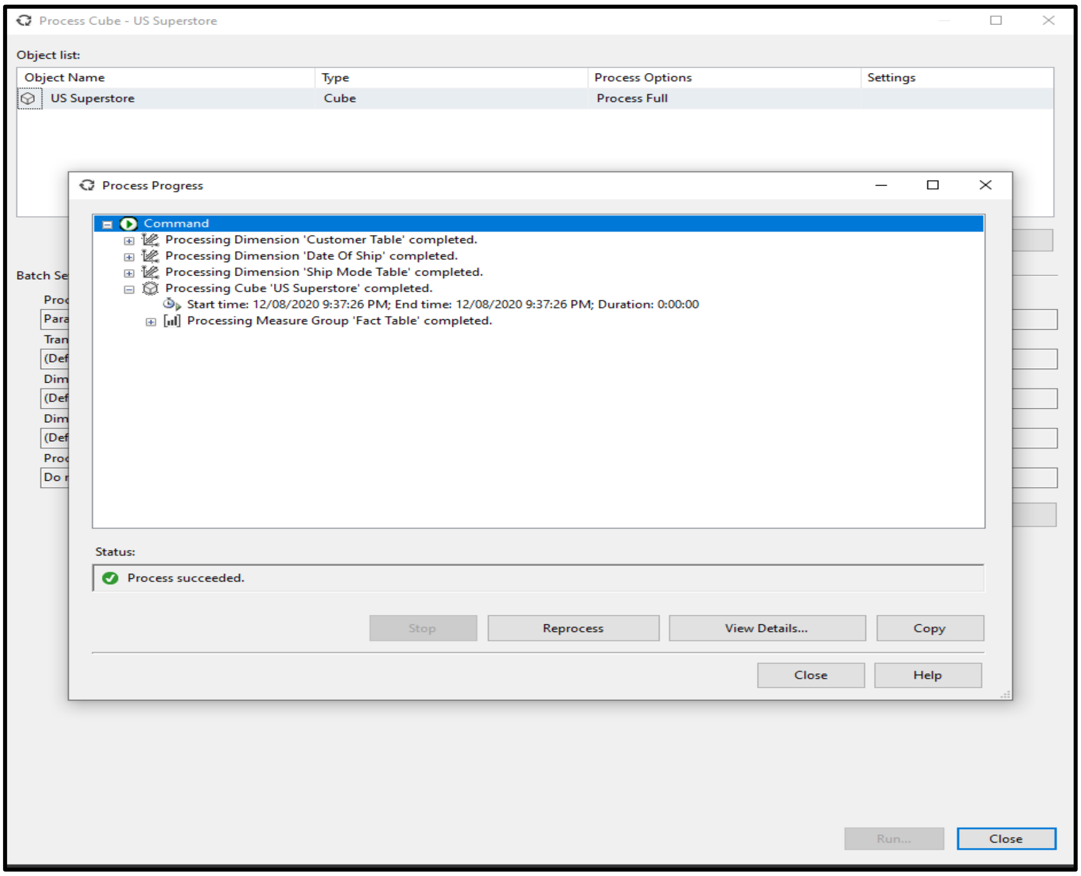
The same procedure was followed for all dimensions. However, they are not reported here for brevity reasons but their hierarchies were created as below:


US Superstore - Microsoft Visual Studio
File Edit View Project Build Debug Team Database Dimension Tools Test Analyze Window Help
Category Table.dim [Design]
US Superstore.cube [Design]
US Superstore.dsv [Design]
Dimension Structure
Attribute Relationships
Translations
Browser
Category ID
Product Name
Sub Category
Category


US Superstore - Microsoft Visual Studio
File Edit View Project Build Debug Team Database Dimension Tools Test Analyze Window Help
Customer Table.dim [Design]*
US Superstore.cube [Design]
US Superstore.dsv [Design]
Dimension Structure
Attribute Relationships
Translations
Browser
Customer ID
Customer Name
Segment

Finally, cube was proccessed and deployed the successfully.





Object Name	Type	Process Options	Settings
US Superstore	Cube	Process Full	

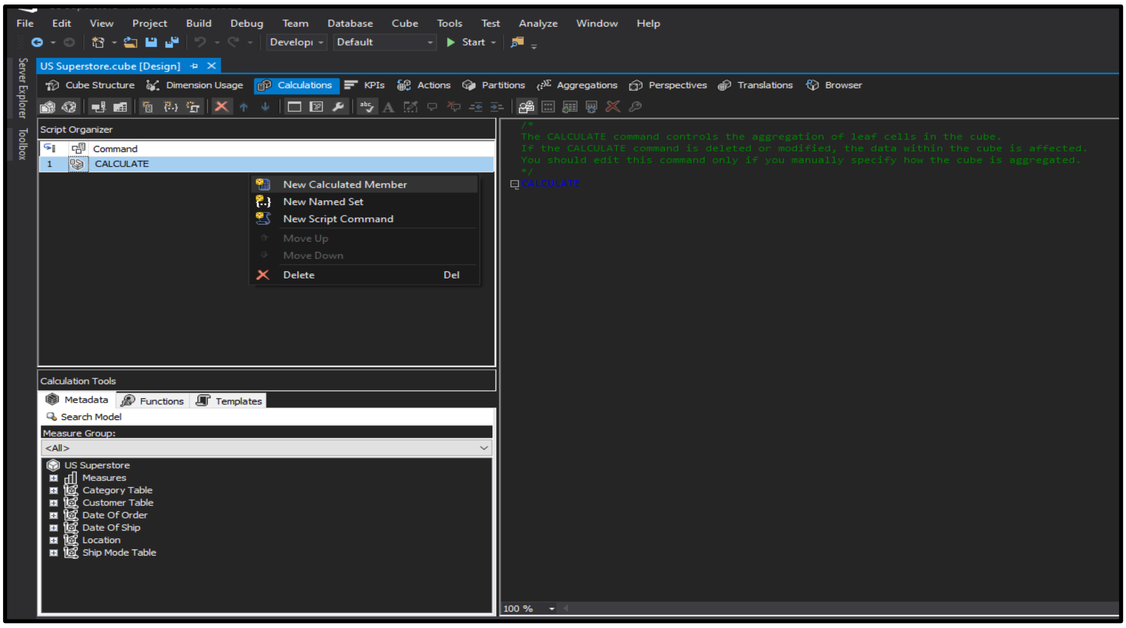
```
Command
  Processing Dimension 'Customer Table' completed.
  Processing Dimension 'Date Of Ship' completed.
  Processing Dimension 'Ship Mode Table' completed.
  Processing Cube 'US Superstore' completed.
    Start time: 12/08/2020 9:37:26 PM; End time: 12/08/2020 9:37:26 PM; Duration: 0:00:00
    Processing Measure Group 'Fact Table' completed.
```

Status:

Process succeeded.

4. Calculations

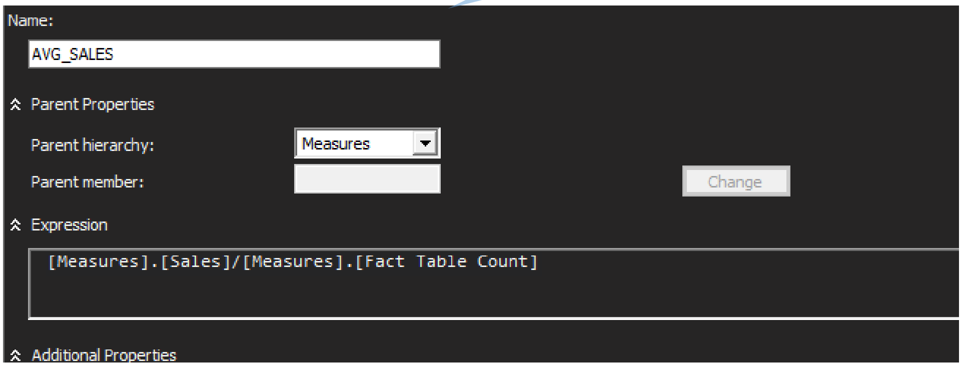
A calculation tool is also provided from Visual Studio. Taking advantage of this opportunity, we created some useful measurements which helped us to end up to some interesting conclusions for our analysis.



```
/*
The CALCULATE command controls the aggregation of leaf cells in the cube.
If the CALCULATE command is deleted or modified, the data within the cube is affected.
You should edit this command only if you manually specify how the cube is aggregated.
*/
CALCULATE
```

- **Average Sales**

$$Avg(Sales) = \frac{Sales}{Total\ Sales}$$



Name:

AVG_SALES

Parent Properties

Parent hierarchy: Measures

Parent member:

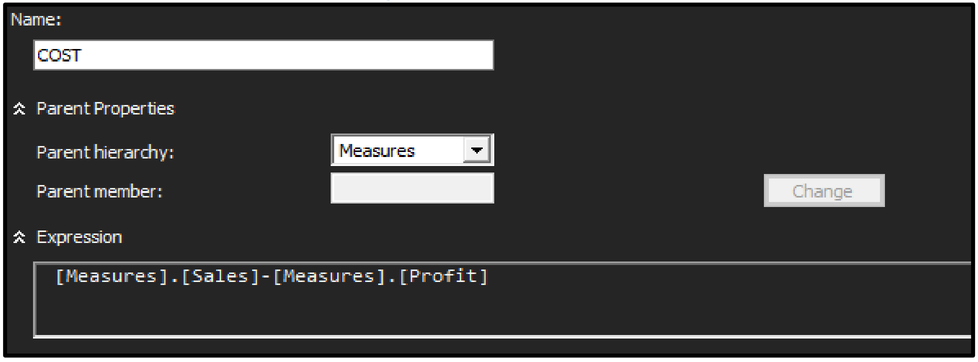
Expression

```
[Measures].[Sales]/[Measures].[Fact Table Count]
```

Additional Properties

- **Cost**

$$Cost = Sales - Profit$$



Name: COST

Parent Properties

Parent hierarchy: Measures

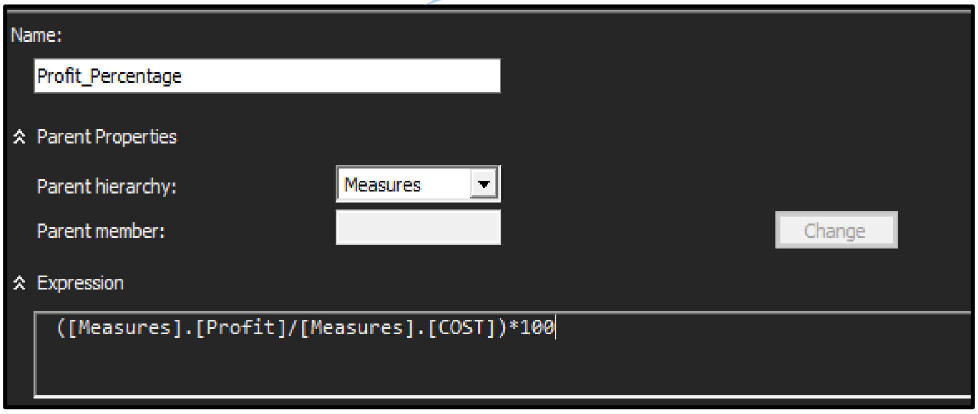
Parent member:

Expression

```
[Measures].[Sales]-[Measures].[Profit]
```

- **Profit_Percentage**

$$Profit\% = \frac{Profit}{Cost} * 100$$



Name: Profit_Percentage

Parent Properties

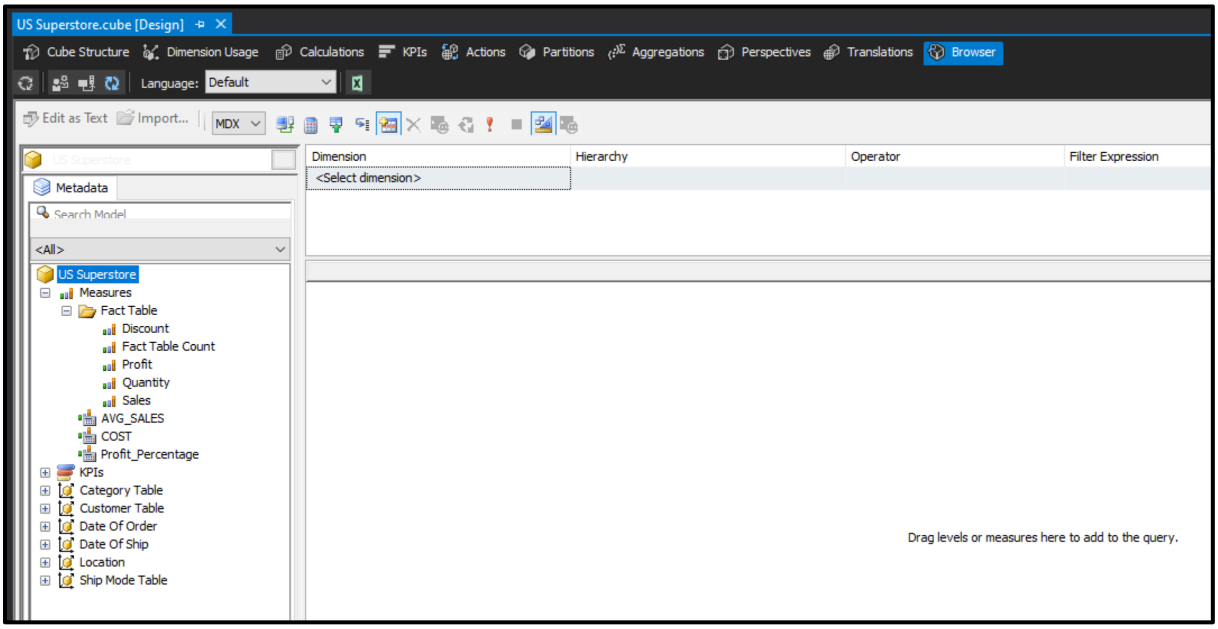
Parent hierarchy: Measures

Parent member:

Expression

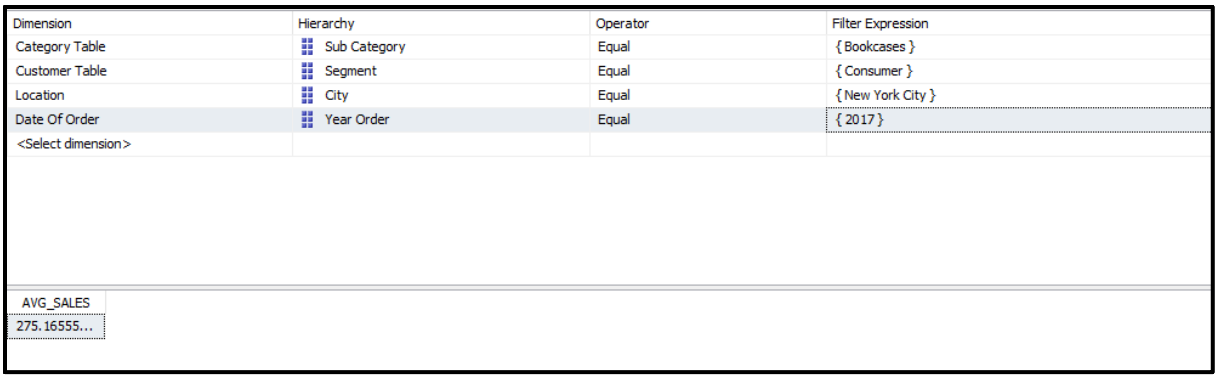
```
([Measures].[Profit]/[Measures].[COST])*100
```

5. OLAP Reports



Drill Down

In the following report we could see the average sales for bookcases in New York for **2017** year which are referred only to consumers.

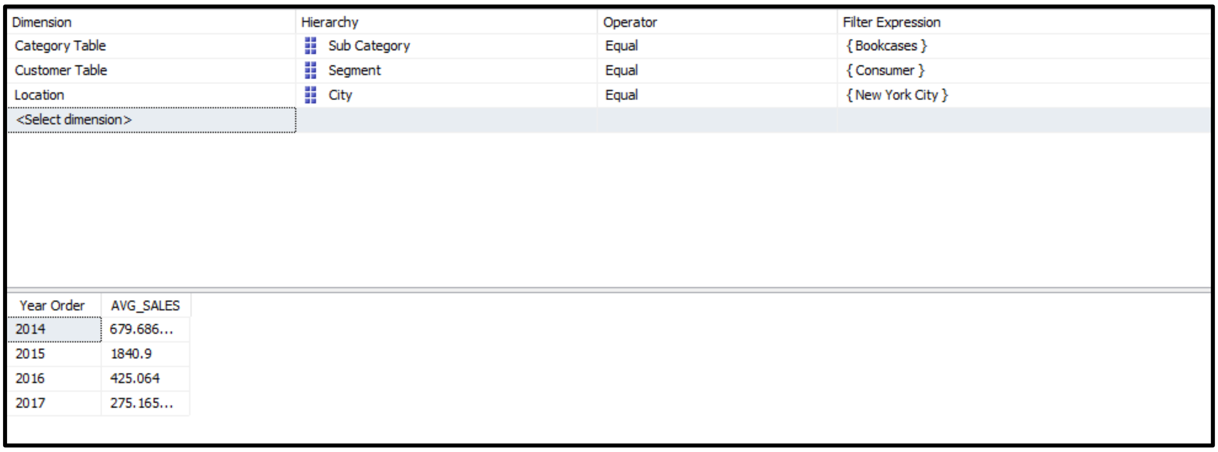


Dimension	Hierarchy	Operator	Filter Expression
Category Table	Sub Category	Equal	{ Bookcases }
Customer Table	Segment	Equal	{ Consumer }
Location	City	Equal	{ New York City }
Date Of Order	Year Order	Equal	{ 2017 }
<Select dimension>			

AVG_SALES
275.16555...

Roll Up

By using roll up, we could see the average sales of consumers for **all years** in New York city who bought bookcases.

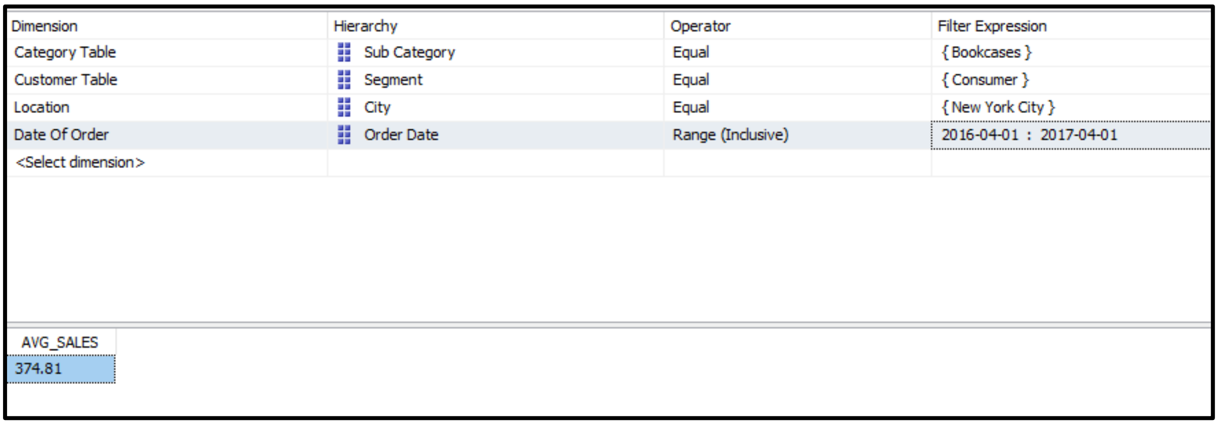


Dimension	Hierarchy	Operator	Filter Expression
Category Table	Sub Category	Equal	{ Bookcases }
Customer Table	Segment	Equal	{ Consumer }
Location	City	Equal	{ New York City }
<Select dimension>			

Year Order	AVG_SALES
2014	679.686...
2015	1840.9
2016	425.064
2017	275.165...

Slice and dice

Finally, a slice and dice example which gives us the average sales of consumers in New York city who bought bookcases between 01/04/2016 and 01/04/2017.



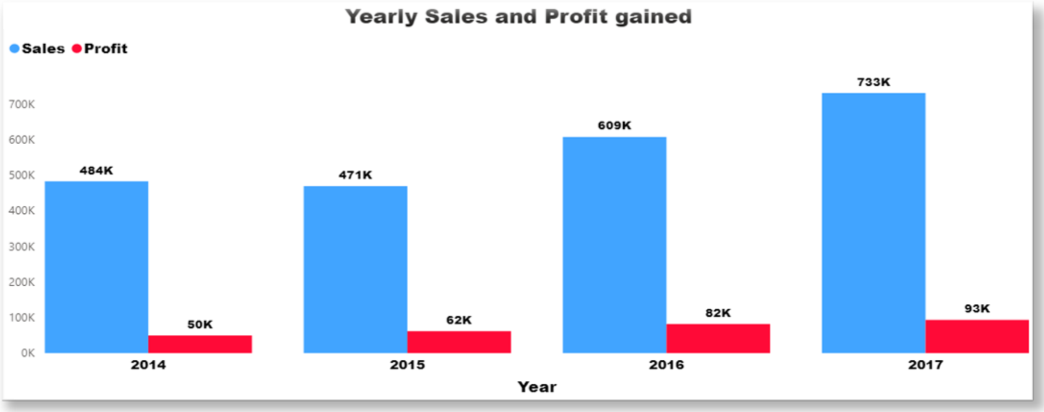
Dimension	Hierarchy	Operator	Filter Expression
Category Table	Sub Category	Equal	{ Bookcases }
Customer Table	Segment	Equal	{ Consumer }
Location	City	Equal	{ New York City }
Date Of Order	Order Date	Range (Inclusive)	2016-04-01 : 2017-04-01
<Select dimension>			

AVG_SALES
374.81

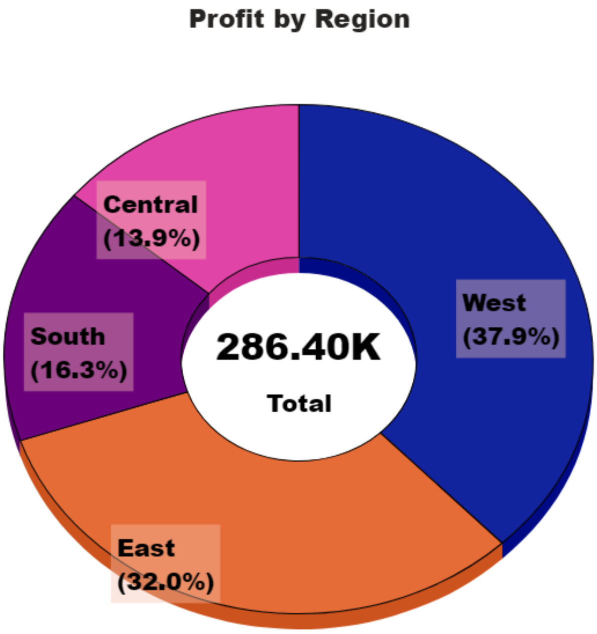
6. Power Bi

We use the Power Bi program in order to make the visualization analysis of our data. Power BI is a cloud-based analysis service that provides rapid insight and is used to extract and visualize data.

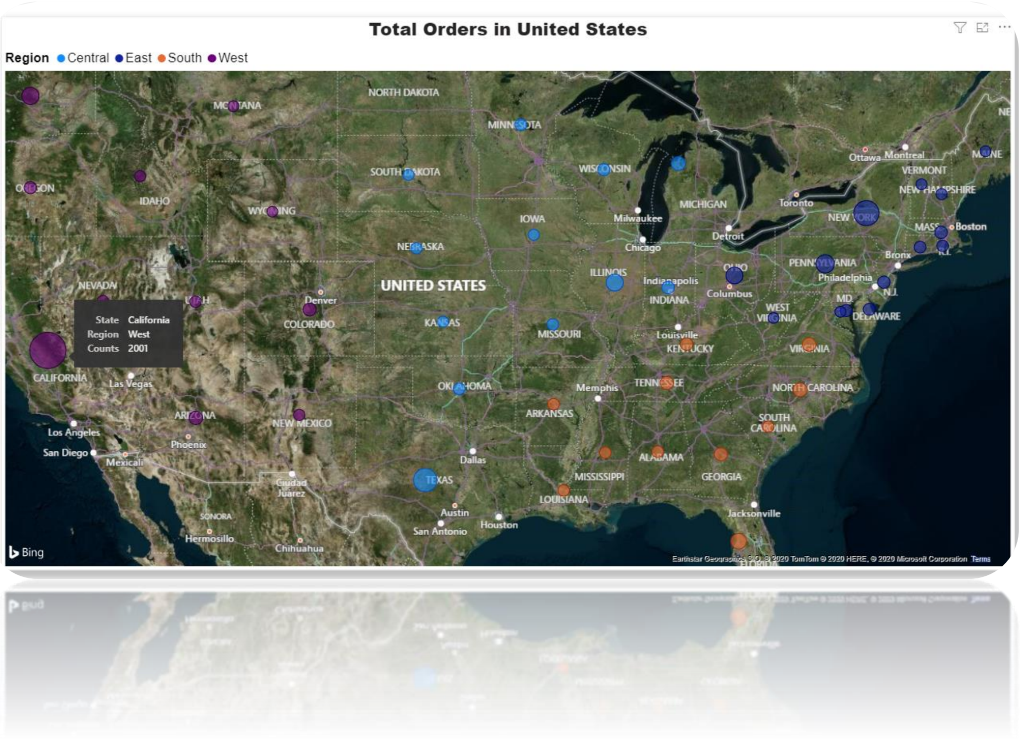
First of all, we would like to have an overview of the Superstore's sales and profits. This could easily depict to a bar char as shown below:



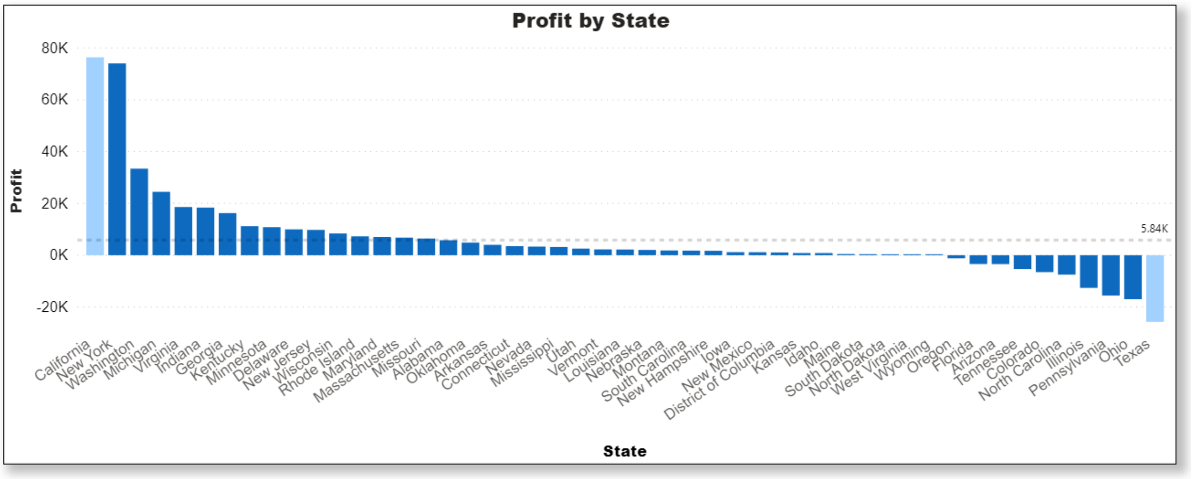
Sales of the store has increased every year resulting in high profit margin by the end of 2017.



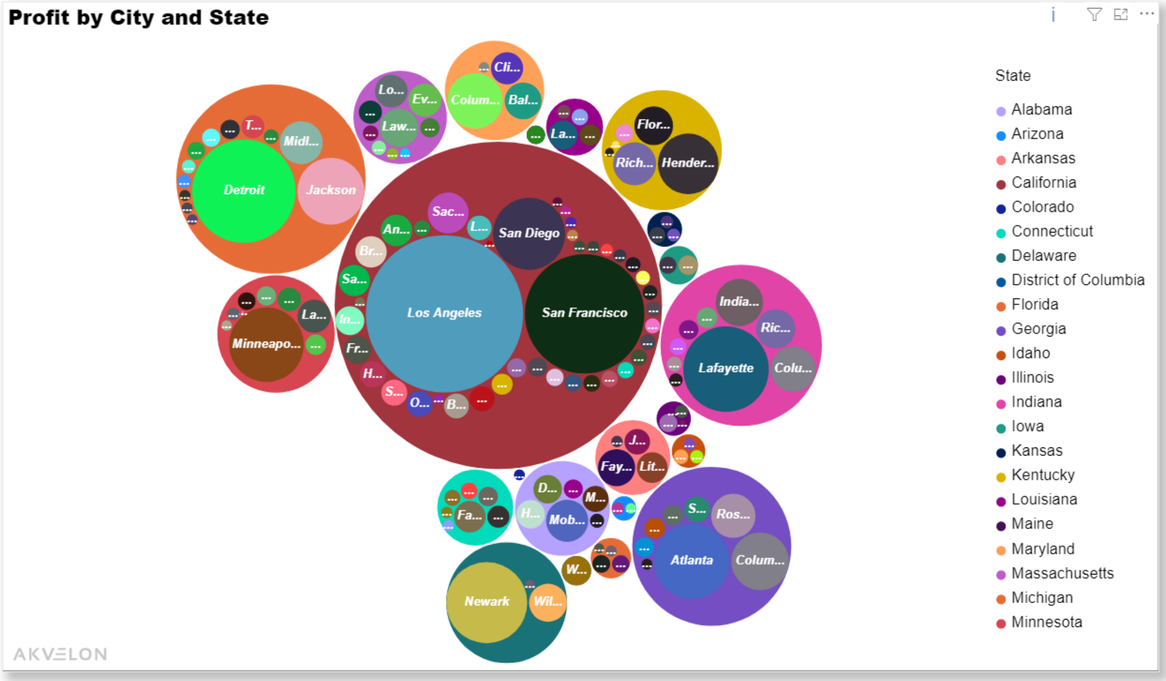
The most profitable region seems to be the Western with 37.9%.


Total Orders in United States
Region ● Central ● East ● South ● West
State California
Region West
Counts 2001
UNITED STATES
Bing
Earthstar Geographics SIO, © 2020 TomTom © 2020 HERE, © 2020 Microsoft Corporation Terms

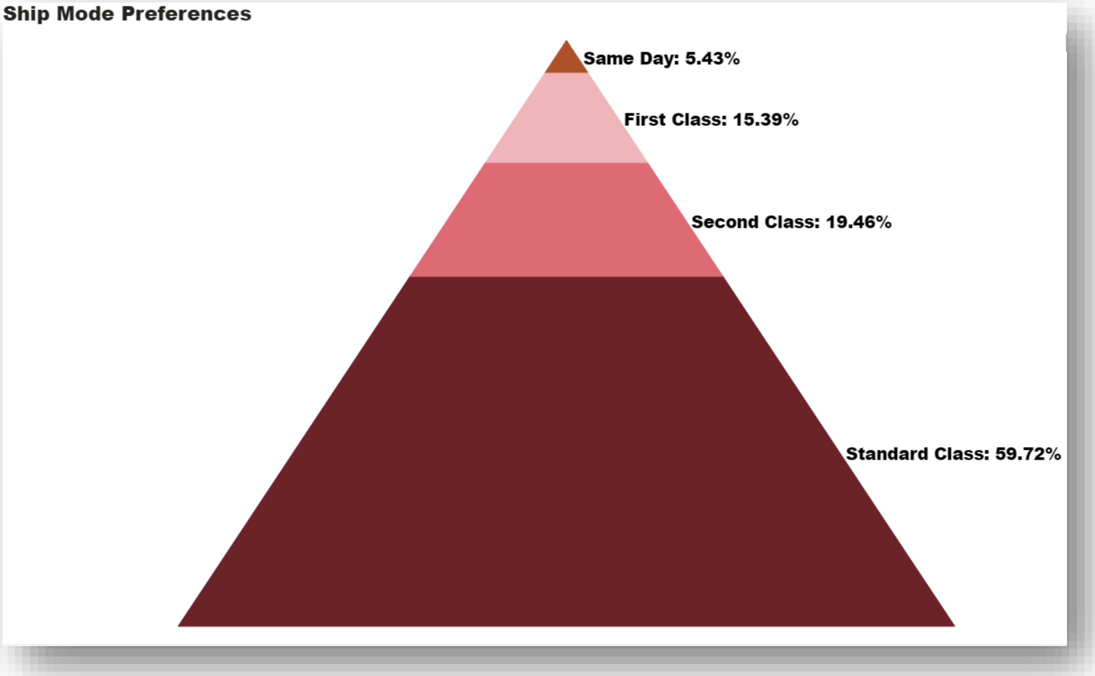
People residing in Western part of US tend to order more from the Superstore with California to be the state with the most orders during the years.



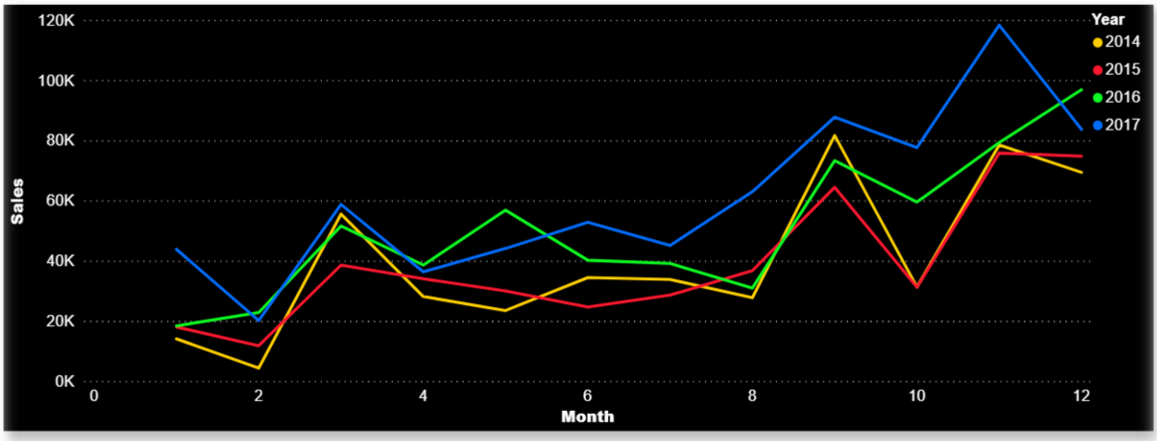
In contrast with California, Texas is the State with the lowest profit (negative).



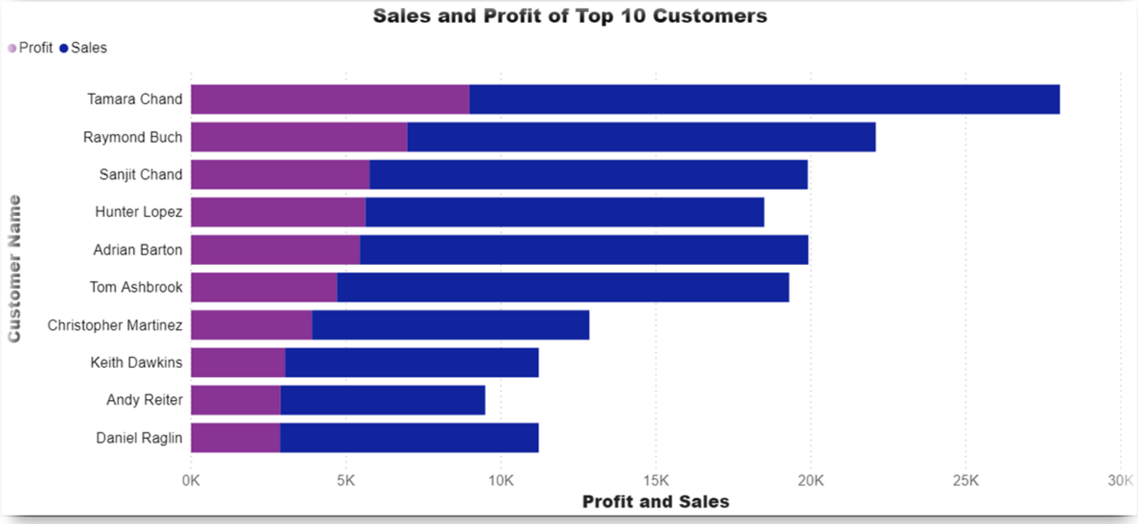
In California the most profitable cities are Los Angeles and San Francisco.



As regards to the ship mode preferences, the majority of people prefer standard class shipment (59,72%) while only 5,43% choose the same day shipping.

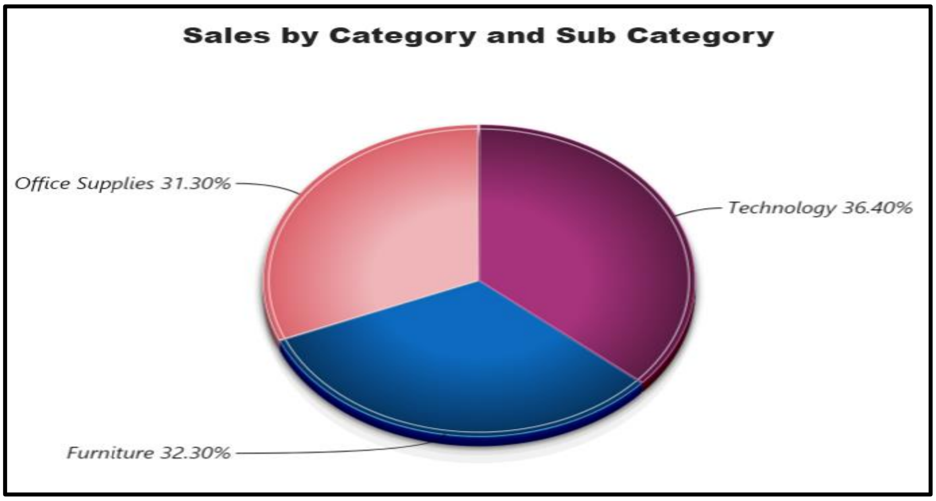


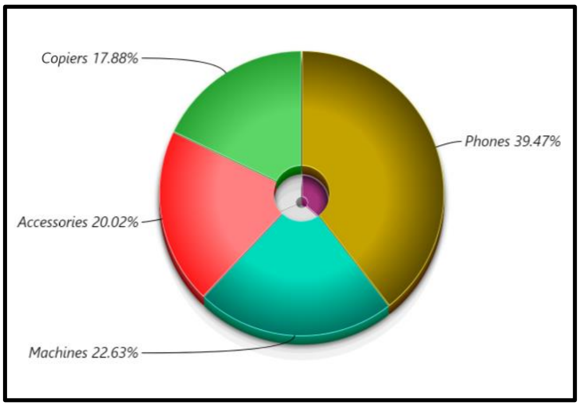
There is an increasing tendency of sales by the years with peak season to be during autumn time.



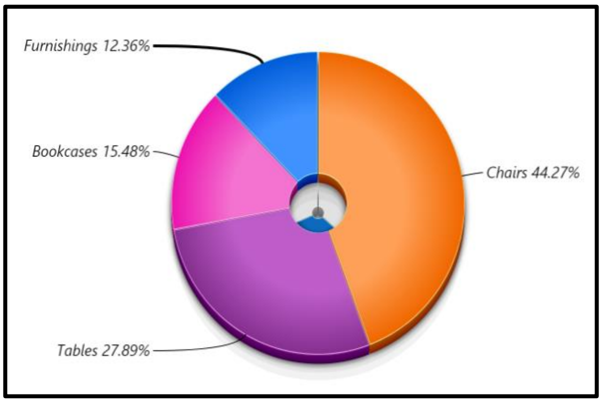
Tamara Chand seems to be the most loyal customer.

We would also like to analyze the products of the US Superstore. For this reason, drill down would be useful in order to come up with some interesting conclusions.

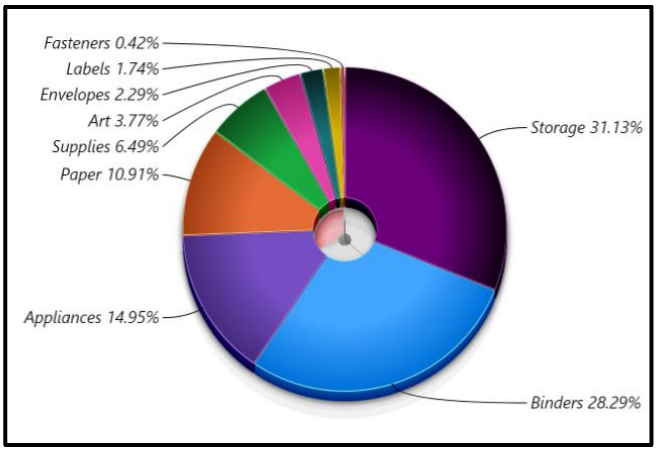




1. ***Technology***

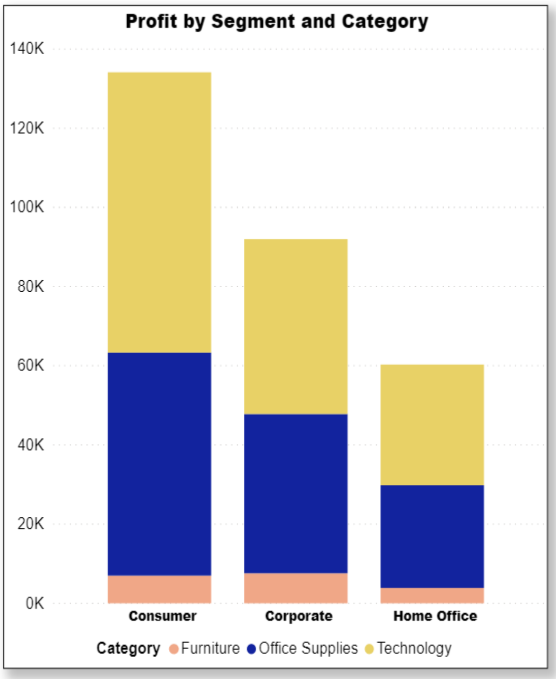


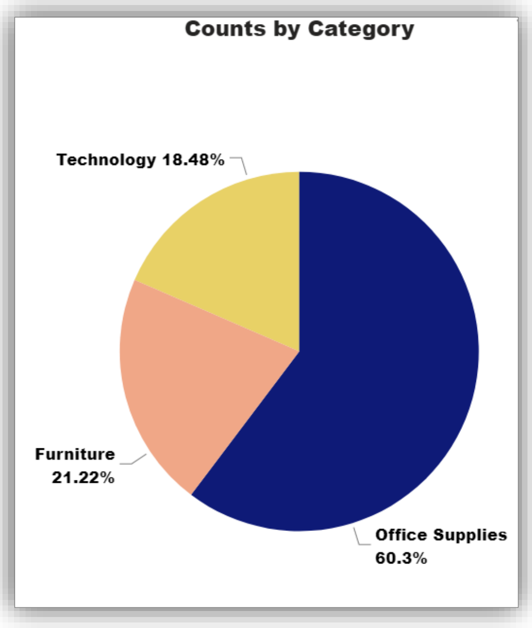
2. Furniture



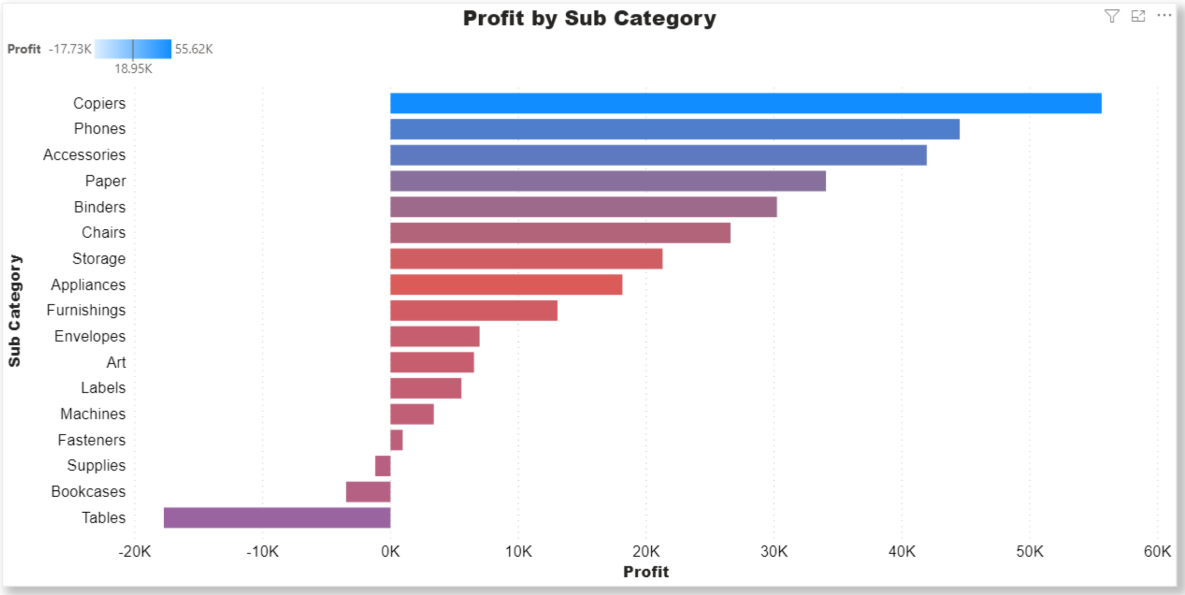
3. Office Supplies

Technology category has the most sales (up to 36.40%) with phones to be the highest desired product. Chairs and storage from furniture and office supplies respectively are the products of the greatest sales.

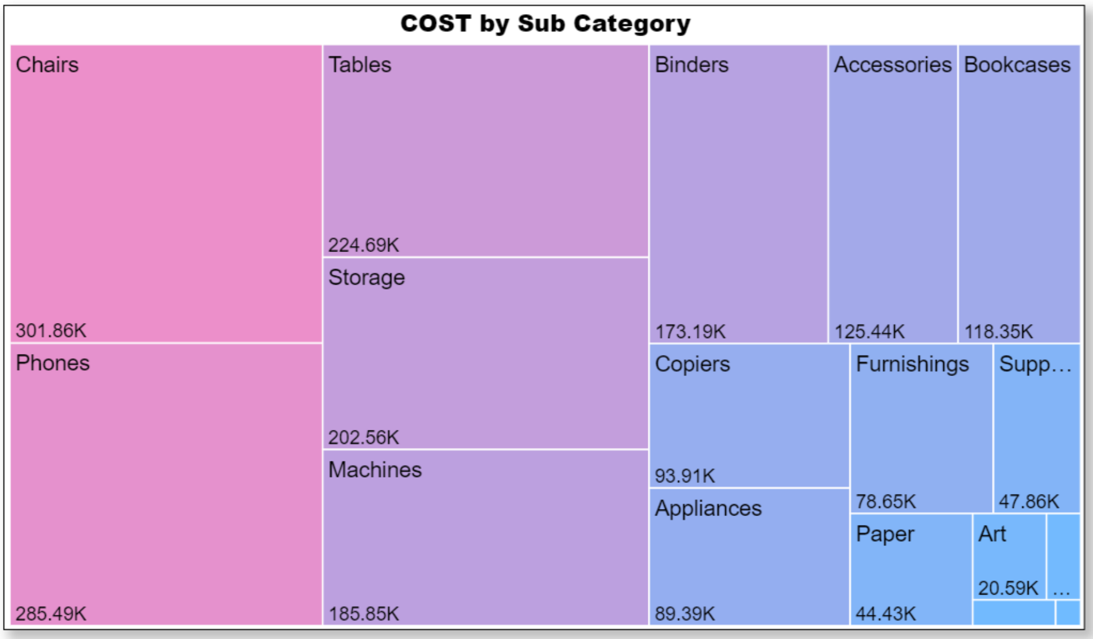




Although office supplies are the most selling category (60,3%) the profit is highest for the technology sector under which the profit has come more from the Consumers segment.



Copiers is the least selling sub-category as we have already seen (only 17.88% of sales). However, taking into consideration this diagram above, copiers has given the most profit out of all the sub categories. Moreover, we could conclude that there is a huge loss from Tables.



It seems that chairs and phones are the products with the highest cost.

References:

In order to create our visualizations, we had to import some new visuals in Power Bi which were found in the above site:

https://appsource.microsoft.com/el/marketplace/apps?product=power-platform%3Bpower-bi&page=1