SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY

IT3021 – DATA WAREHOUSE AND BUSINESS INTELLIGENCE

ASSIGNMENT – 02

2022

SUBMITTED BY:

IT19991986 – RAEESUL ISLAM S.Z.

Y3S2.WE.DS.03

Table of Contents

1. Data Source

Data Warehouse implemented in the previous assignment was used as the source to complete **Assignment 1**. As described in the Assignment I, the selected data set consisted of transactional data. Customer specific details involved in transactions, Items, customers are keen to purchase, customer participation to promotion campaigns are some of the key details included in the data set.

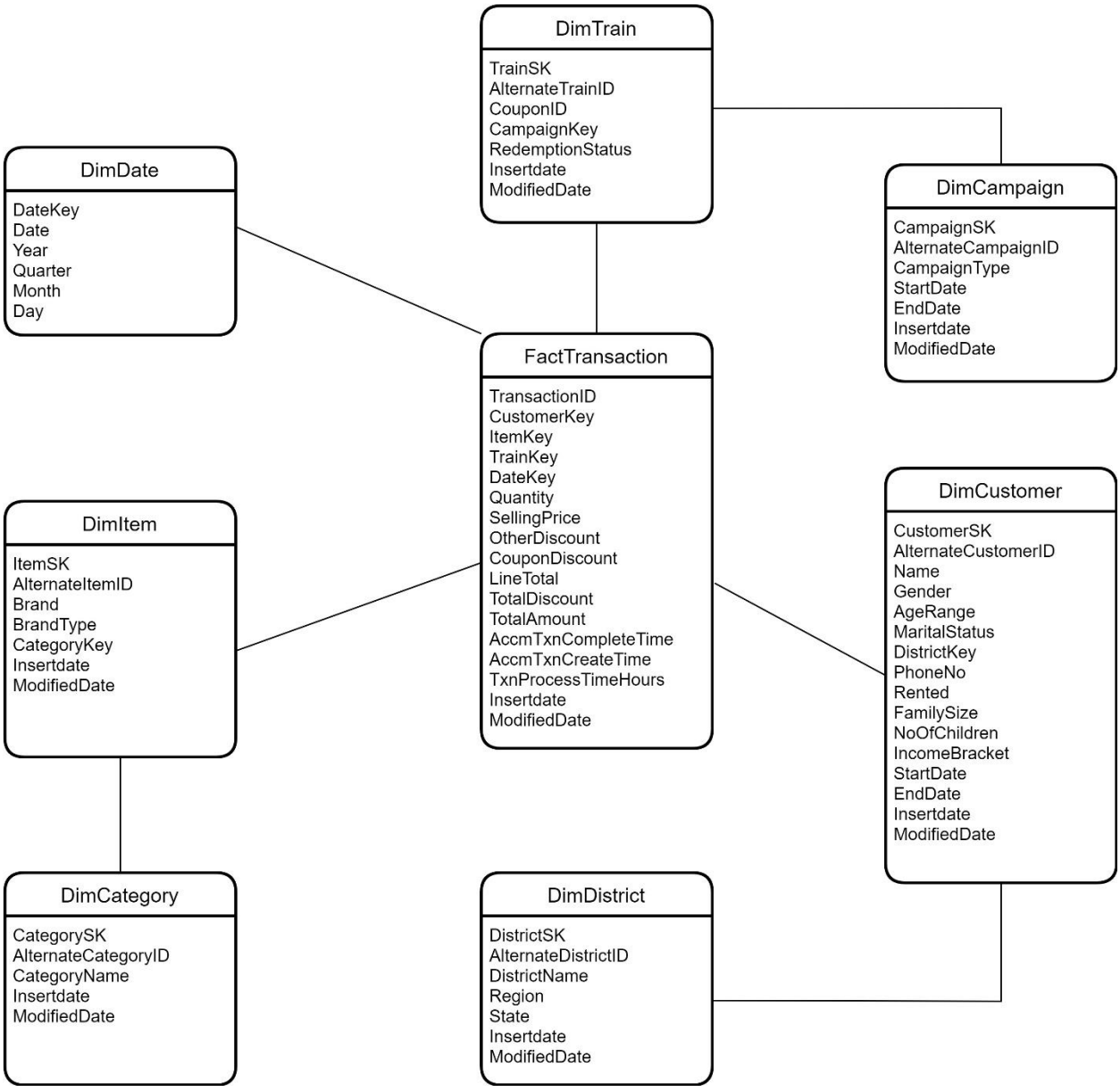


Figure 1. Snowflake Schema for Data Warehouse

The Data Warehouse design was implemented using the **Snowflake Schema**. Snowflake Schema is an extension of **Star Schema** and consists of some dimensions that are normalized. According to the schema above, there are **7 Dimensions** and **1 Fact table**.

Assumption:

Customer Dimension is considered as a **Slowly Changing Dimension (SCD)**

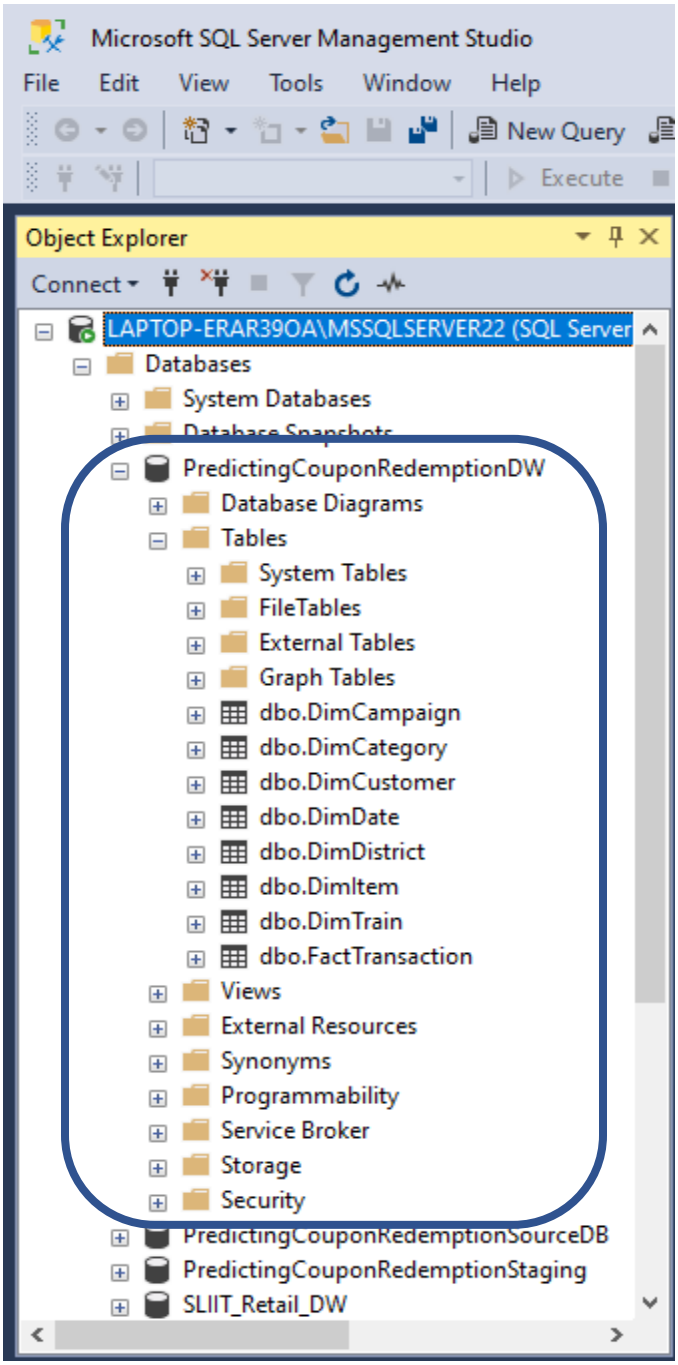


Figure 2. Data Warehouse Snapshot

Dimension Tables:

01. DimCategory
02. DimCampaign
03. DimCustomer
04. DimDate
05. DimDistrict
06. DimItem
07. DimTrain

Fact Table:

01. FactTransaction

2. SSAS Cube Implementation

OLAP Cube is a method for storing data in Multidimensional Forms. It will allow to analyze a multidimensional data from multiple perspectives. The advantage of using a cube is that it pre-calculates most of the queries, that is time consuming to execute over relational tables that contains joins and aggregates. The main components of the cube are:

01. Dimensions: Define the structure of the cube that is used for OLAP operations.

02. Measures: Provide aggregated numeric values of interest to the end user.

2.1. Cube Creation

- ✓ As the first step an analysis service project in the name '**PredictingCouponRedemption_SSAS**' was created was the data source was configured in order to extract data to the cube.
- ✓ A data source view '**DSV_PredictingCouponRedemption**' was created and all necessary table links were created.
- ✓ A cube named '**PredictingCouponRedemption**' was created, by selecting the necessary measures. Then necessary attributes and hierarchies were added before the deployment of the cube.

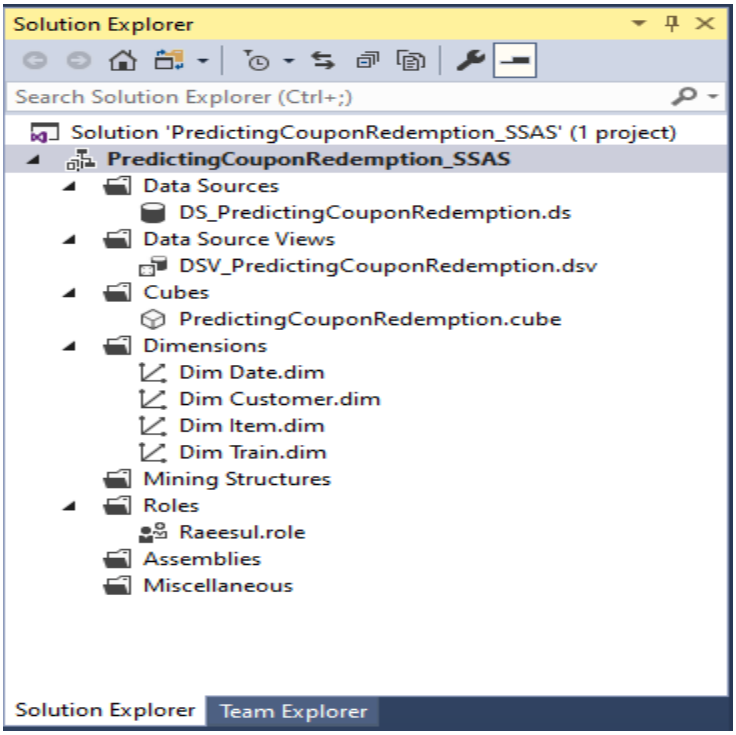


Figure 3. Cube Creation Solution Explorer

2.2. Data Source View

It represents the cube structure, measures, and dimensions.

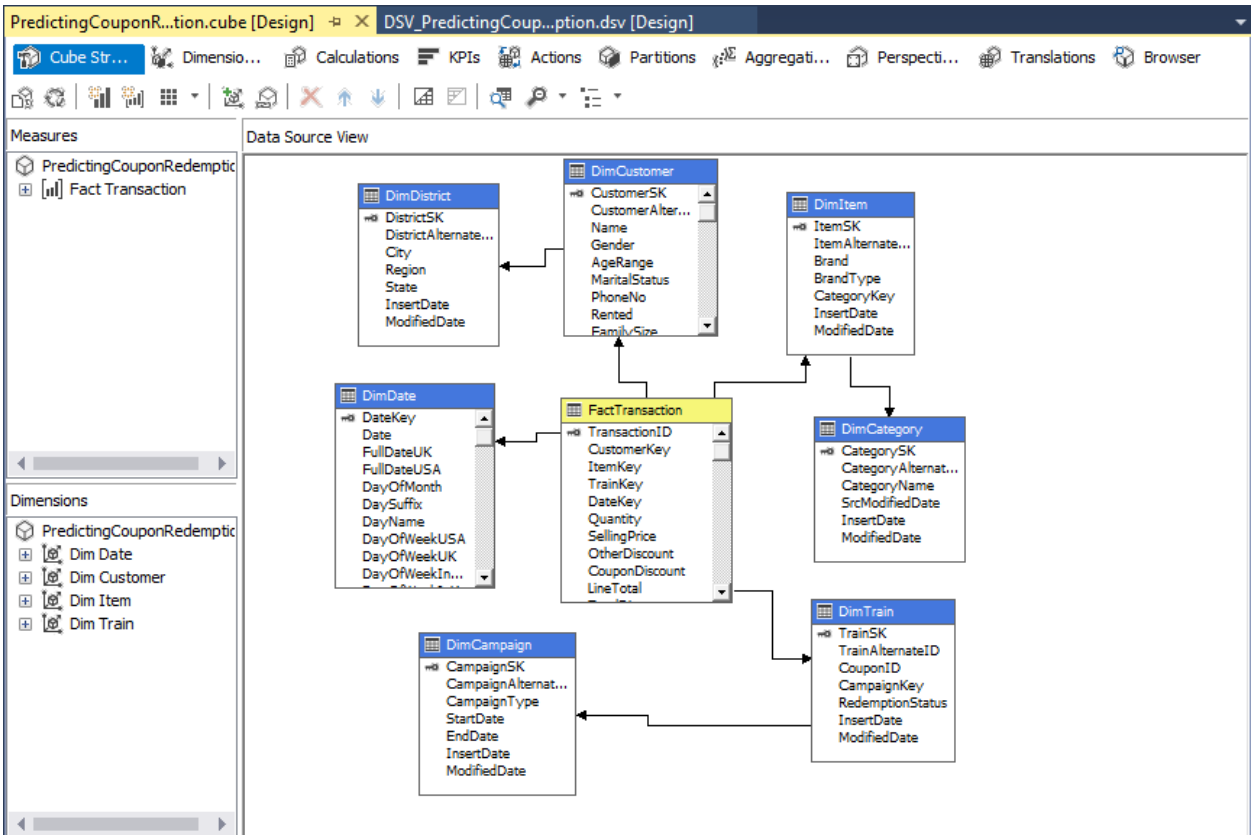


Figure 4. Cube Structure

2.3. Hierarchies

Hierarchies are a useful tool in SSAS to reduce complexity between attributes and guide users into a certain drill-down behavior.

1. **Date Hierarchy**

   - The higher level is the **Year**, which then is followed by the lower levels **Quarter**, **Month**, and **Date**.

Figure 5. Date Hierarchy

2. **Location Hierarchy**

   - The higher level is the **Regions**, which contains multiple **States**, and the States contain multiple **Cities.**

Figure 6. Location Hierarchy

3. **Brand Hierarchy**

   - The higher level is **Brand Type**, which contains multiple **Brands**

Figure 7. Brand Hierarchy

2.4. KPI Values

In **SQL Server Analysis Services**(SSAS), add **Key Performance Indicators** (KPIs) can be added to our database cube in order to evaluate business performance, as reflected in the cube data. A KPI is associated with a measure group and is made up of a set of calculations. Typically, the calculations are a combination of calculated members and Multidimensional Expressions (MDX) statements.

KPI Total Transaction Amount:

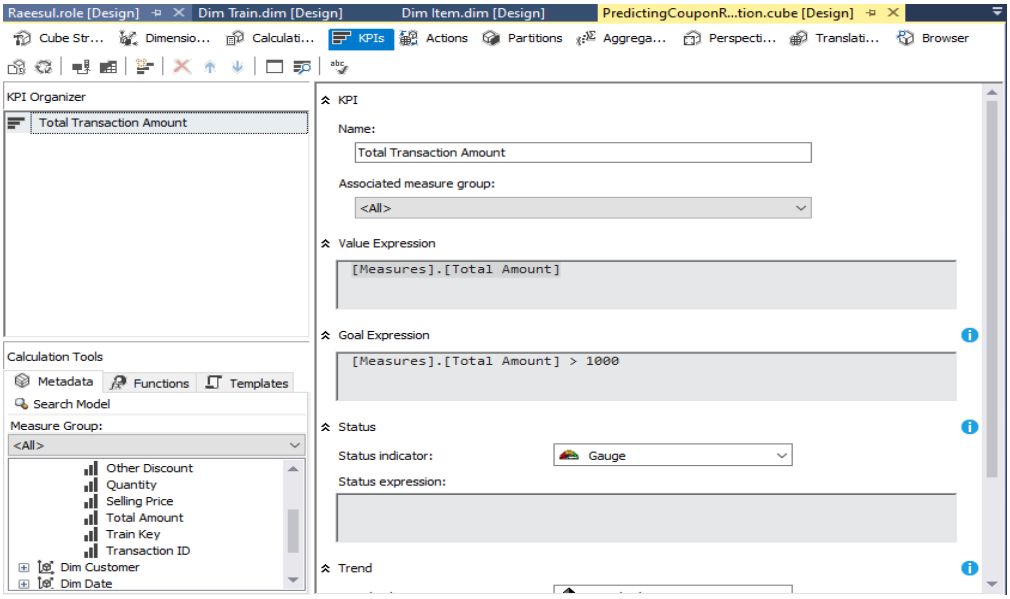


Figure 8. KPI – Total Amount

3.5.Creating a Role

a role was created by defining permissions assigned to the role. The particular role has full control.

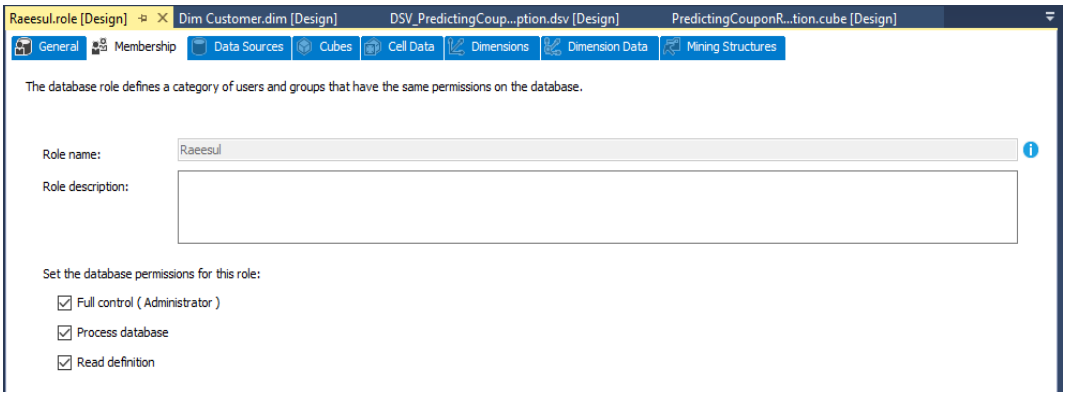


Figure 9. Role Creation

2.6. Cube Deployment

After setting all attributes, hierarchies and KPI's, finally the cube was deployed.

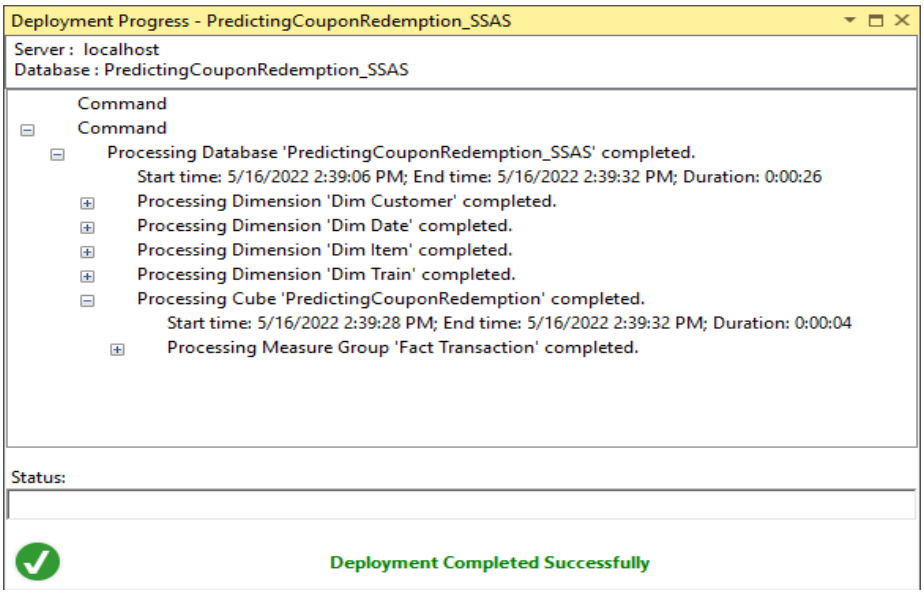


Figure 10. Cube Deployment Completed Successfully

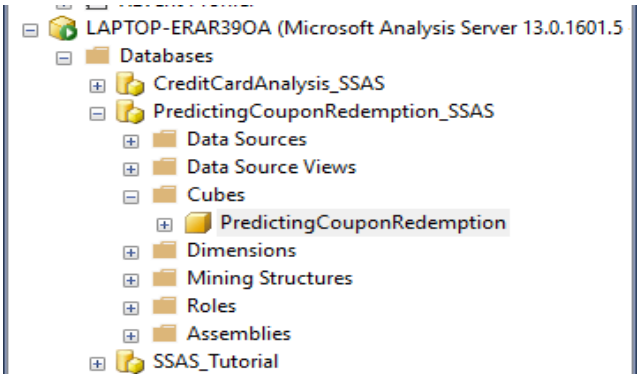


Figure 11. SSAS in SSMS

3. Demonstration of OLAP Operations

- ✓ **OLAP** operation is an important part of **Business Intelligence**, that provides powerful capabilities for data mining and trend analysis. They are capable of solving problems in both business and IT departments. OLAP helps to analyze big data amounts from different perspectives rapidly.

- ✓ **MDX** query can be used to connect to the excel workbooks to get data to the semantic layer for respective demonstrations. This method needs to build up MDX query through SSAS project by browsing data. The same can be done using 'data' tab in excel. This will enable the connection with the whole set of facts and dimension tables. For the assignment purpose, the second approach of connecting to the entire data set was used. Five analytical operations can be performed using OLAP:
  1. Roll-up
  2. Drill-down
  3. Slice
  4. Dice
  5. Pivot

3.1. Roll – Up

- **Roll-up** is also known as 'consolidation' and 'aggregation,' which can be performed in two ways:
  1. Reducing dimensions
  2. Climbing up a dimension hierarchy

3.2. Drill Down

- **Drill-down** is the opposite of roll-up. It means to step down a hierarchy, which will enable navigation through details of a dimension. This operation fragments data into smaller parts. It can be done via:
  1. Moving down a hierarchy
  2. Increasing the dimension.

Report :

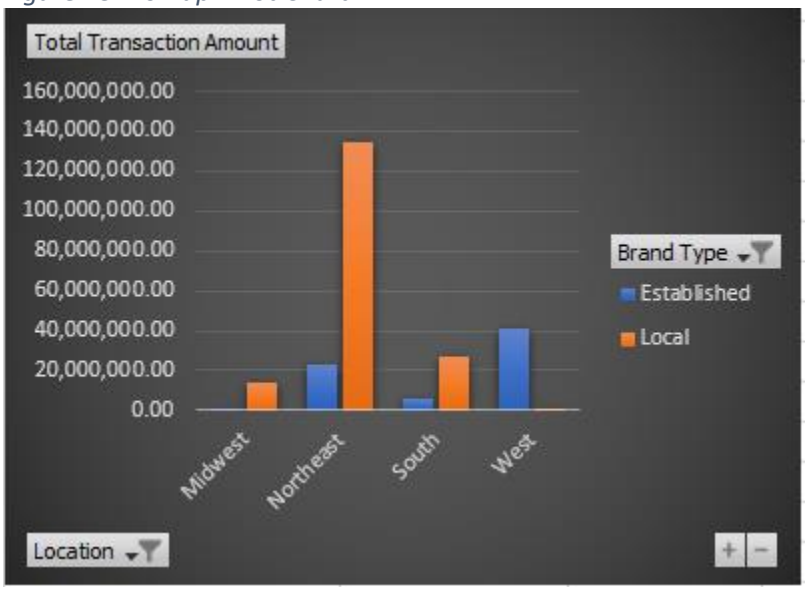
- ✓ **Total Transaction Amounts** based on **Roll-ups** and **Drill-down** of **Region** and **State**. The report displays the **Total Transaction Amount** for each **Brand Type**, based on the **Roll-ups** and **Drill-downs** of **Regions, States,** and **Cities**.

Roll-up (Regions rolled-up to States)

Figure 12. Roll-up Pivot Table

Total Transaction Amount	Column Labels		
Row Labels	Established	Local	Grand Total
⊞ Midwest	17,167.67	13,924,288.24	13,941,455.91
⊞ Northeast	22,654,945.68	133,984,893.05	156,639,838.73
⊞ South	5,920,272.94	27,425,167.37	33,345,440.31
⊞ West	40,821,193.42	33,601.36	40,854,794.78
Grand Total	69,413,579.71	175,367,950.02	244,781,529.73

Figure 13. Roll-up Pivot Chart



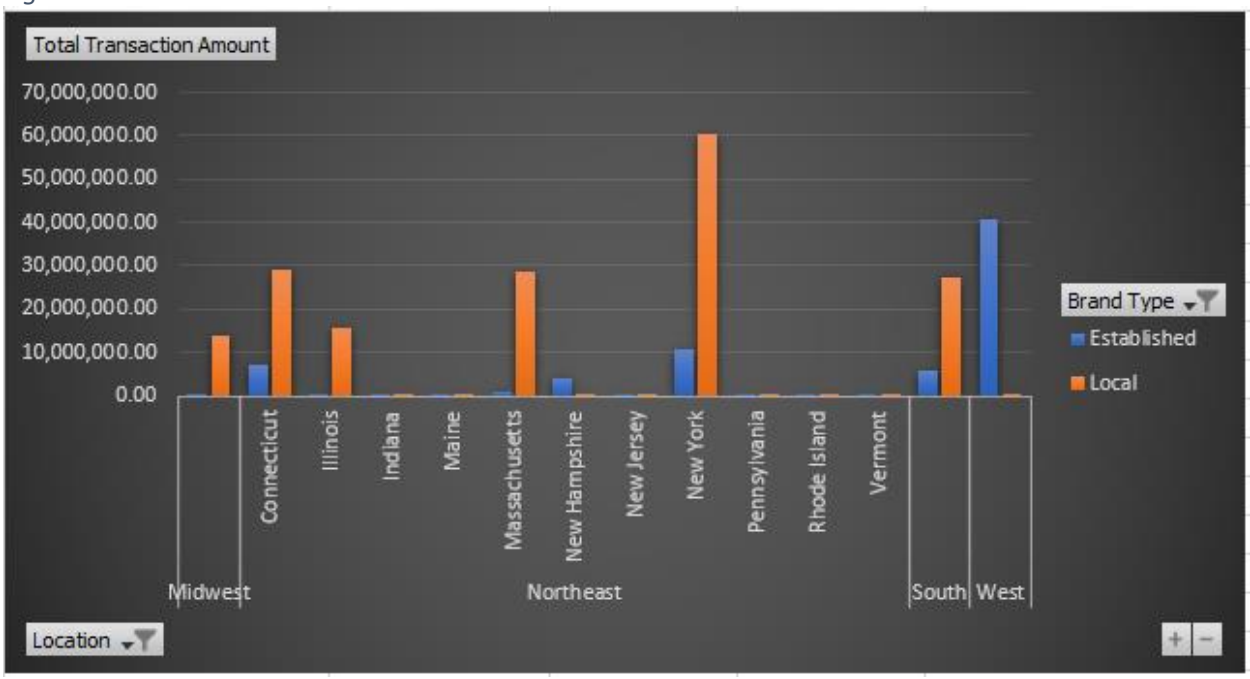
- ❖ The **Pivot table** and the **Pivot chart** show the **Total Transaction Amounts** for all the rolled-up **States**, according to each **Brand Type**.

Drill-down(States have been drilled down to Regions)

Figure 14. Drill-down Pivot Table

Total Transaction Amount	Column Labels		
Row Labels	Established	Local	Grand Total
Midwest	**17,167.67**	**13,924,288.24**	**13,941,455.91**
Northeast			
Connecticut	7,002,775.13	29,276,466.94	36,279,242.07
Illinois	14,526.53	15,460,676.30	15,475,202.83
Indiana	974.57	1,192.90	2,167.47
Maine	1,143.41	2,218.40	3,361.81
Massachusetts	973,165.42	28,694,564.66	29,667,730.08
New Hampshire	4,011,361.75	5,879.05	4,017,240.80
New Jersey	2,536.13	2,696.39	5,232.52
New York	10,635,250.61	60,519,758.03	71,155,008.64
Pennsylvania	4,415.42	4,226.31	8,641.73
Rhode Island	7,817.18	13,672.74	21,489.92
Vermont	979.53	3,541.33	4,520.86
South	**5,920,272.94**	**27,425,167.37**	**33,345,440.31**
West	**40,821,193.42**	**33,601.36**	**40,854,794.78**
Grand Total	**69,413,579.71**	**175,367,950.02**	**244,781,529.73**

Figure 15. Drill-down Pivot Chart



- The **Pivot table** and **Pivot chart** represent the **Total Transaction Amounts** for the **drilled downed** States of the Region '**Northeast**,' for each **Brand Type**.

3.3. Slice

- **Slice** create a rectangular subset of the cube, by selecting a single value for one of its dimensions. A slice function is much like a report or a query that it returns data based on a request for what to see.

Report:

✓ **Line Total,** and the **Total Amount** based on **Regions** and the **Item Category**. The report represents the **Line Total,** and the **Total Amount** based on a selected **Regions** and the **Item Category** belonging to the **Region**.

Figure 16. Pivot Table for Slicer

Row Labels	Line Total	Total Amount
Northeast		
Bakery	3,726,720.37	3,724,589.96
Grocery	136,764,078.40	136,740,221.82
Juices & Snacks	2,389.01	2,157.13
Miscellaneous	1,580.11	1,534.16
Natural Products	1,785.99	1,500.68
Packaged Meat	5,233.62	4,509.48
Pharmaceutical	16,162,989.92	16,157,998.50
Prepared Food	1,880.02	1,681.62
Seafood	5,334.79	4,760.23
Skin & Hair Care	1,005.54	885.15
Grand Total	156,672,997.77	156,639,838.73

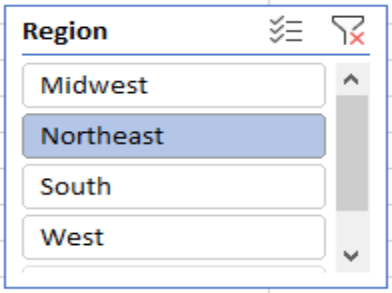
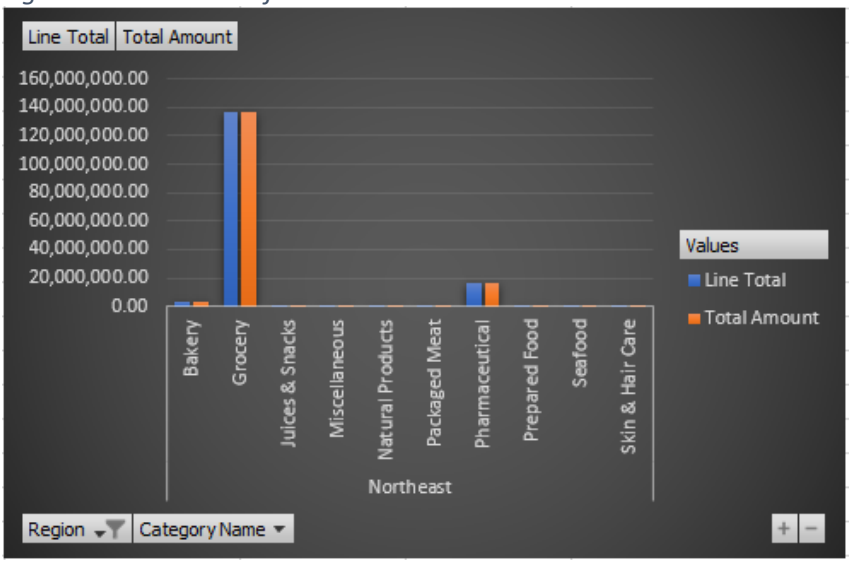
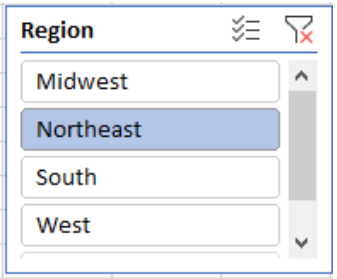


Figure 17. Pivot Chart for Slicer





❖ **Pivot Table** and **Pivot Chart** show the **Line Total** and **Total Amount** based on slicing of Region '**Northeast.**

3.4. Dice

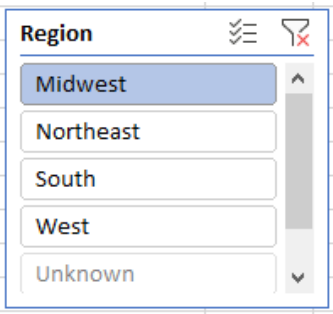
- **Dice** operation selects two or more dimensions from a cube, and results in a sub cube by selecting specific values on those selected dimensions. Dicing on the other hand, is more of a zoom feature that selects a subset over all the dimensions, but for specific values of the dimension.

Report:

✓ **Quantity** based on **Region** and **Brand Type**. The report represents the **Quantity**, that could be gathered for a **selected Region, and a selected Brand Type.**

Figure 18. Pivot Table for Dice

Row Labels	Quantity
Midwest	
Established	
Grocery	124
Juices & Snacks	8
Packaged Meat	8
Pharmaceutical	32
Skin & Hair Care	3
Grand Total	175



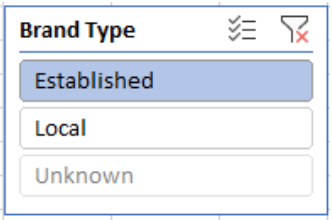
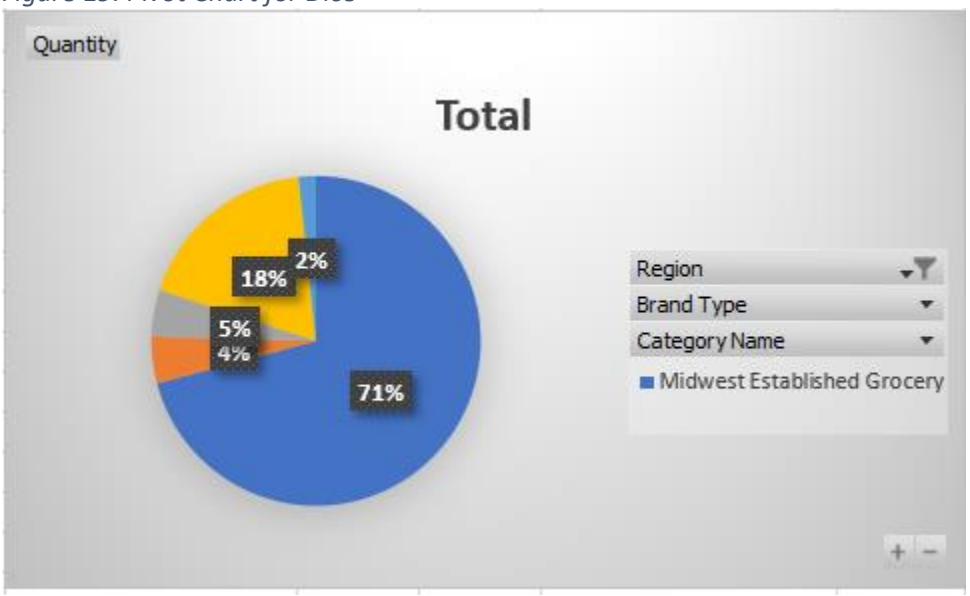


Figure 19. Pivot Chart for Dice



❖ **Pivot Table** and **Pivot Chart** show the **Quantity** for selected Region **'Midwest'**, and selected Brand Type **'Established'**.

3.5. Pivot

- **Pivot** operation provides a new perspective to the cube by rotating the data axes of the cube. It may contain swapping the rows and columns or moving one of the Row dimensions into the column dimensions.

Report:

✓ **Total Amounts, Quantity** and **Number of Transactions** based on **Campaign Type**

Figure 20. Pivot Table

Row Labels	Quantity	Total Amount	Fact Transaction Count
X	264,119	265,300,680.31	2,621
Y	81,012	87,048,764.36	979
Grand Total	345,131	352,349,444.67	3,600

❖ The report with **Campaign Type** as the rows and **Total Amounts, Quantity** and **Number of Transactions** as columns.

Figure 21. Change the Angle of the Pivot Table

	Column Labels		
Values	X	Y	Grand Total
Quantity	264,119	81,012	345,131
Total Amount	265,300,680.31	87,048,764.36	352,349,444.67
Fact Transaction Count	2,621	979	3,600

❖ The report has now changed the perspective, as **Total Amounts, Quantity** and **Number of Transactions** are transposed to rows and **Campaign Type** to Columns.

Report 1:

✓ This report analyses data between **Categories, Brand hierarchy, KPI Goal** and **Total Transaction Amount**. To find out the Categories and Items which makes more profit during the business process. **Power pivot**, **Slice**, and **Dice** have been used.

	Column Labels			
	Established		Total Total Transaction Amount	Total Total Transaction Amount Goal
Row Labels	Total Transaction Amount	Total Transaction Amount Goal		
⊞Grocery	39,800,180.43	TRUE	39,800,180.43	TRUE
⊞Juices & Snacks	6,227.44	TRUE	6,227.44	TRUE
⊞Miscellaneous	1,600.77	TRUE	1,600.77	TRUE
⊞Packaged Meat	6,790.18	TRUE	6,790.18	TRUE
⊞Pharmaceutical	60,330,486.98	TRUE	60,330,486.98	TRUE
⊞Skin & Hair Care	1,527.74	TRUE	1,527.74	TRUE
Grand Total	100,146,813.54	TRUE	100,146,813.54	TRUE

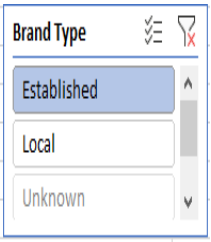


Figure 22. Pivot Table for Report 1

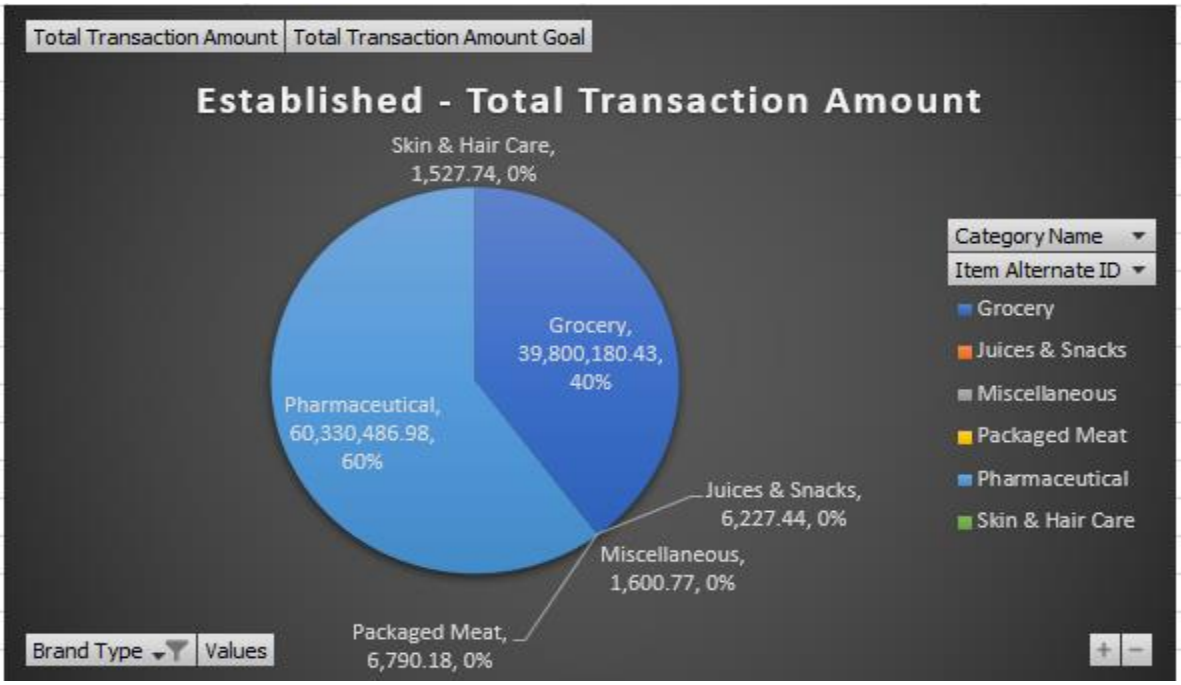


Figure 23. Pivot Chart for Report 1

3.6.Power BI Reports

Report 1:

- ✓ This report shows **Line Total and Total Income** and **Campaign types of Categories and Regions**. To analyse the different between the **Line Total vs Total Income**. And also, to take decision on promoting Items using Campaign Type. Power pivot and roll-up and drill-down have been used.

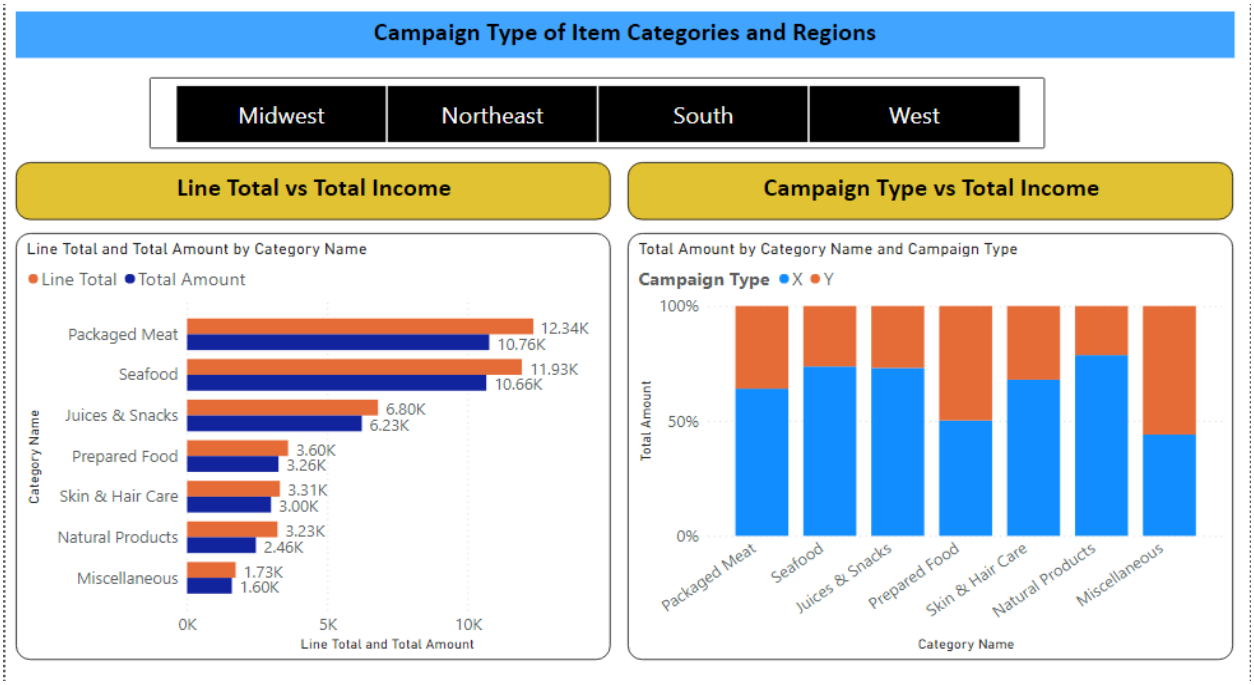


Figure 24. Power BI Visualization for Report 1

Report 2:

- ✓ This report analyses data between **States and Total income**. This is to focus on **States** which helps to reach the **KPI goals**. And also, to motivate other **States** in order to reach goal and increase the performance of the business.

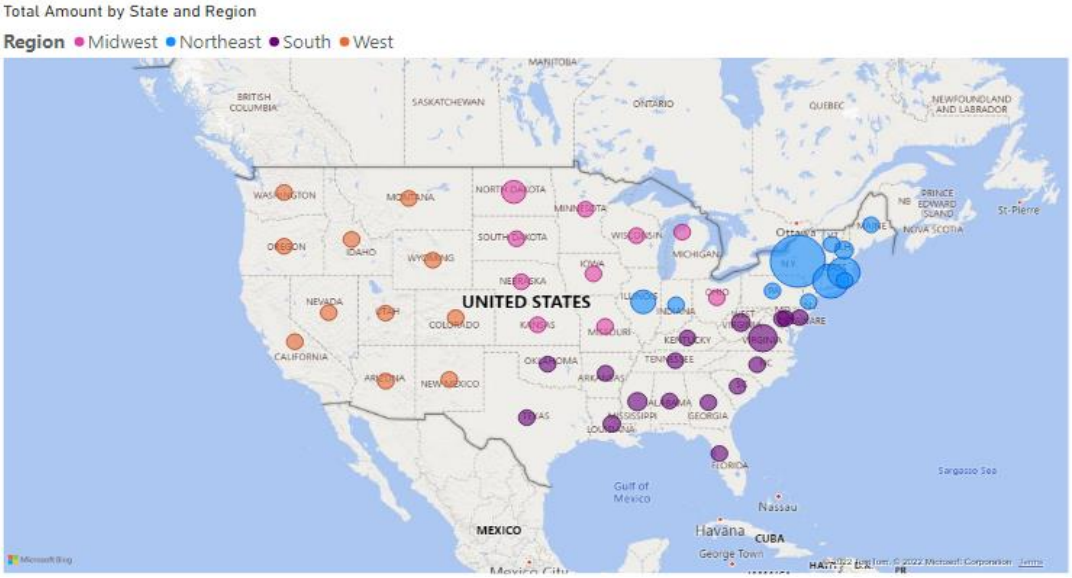


Figure 25. Power BI Visualization for Report 2

Report 3:

- ✓ This report analyses overall Total Income. This is to focus to increase the performance of business.

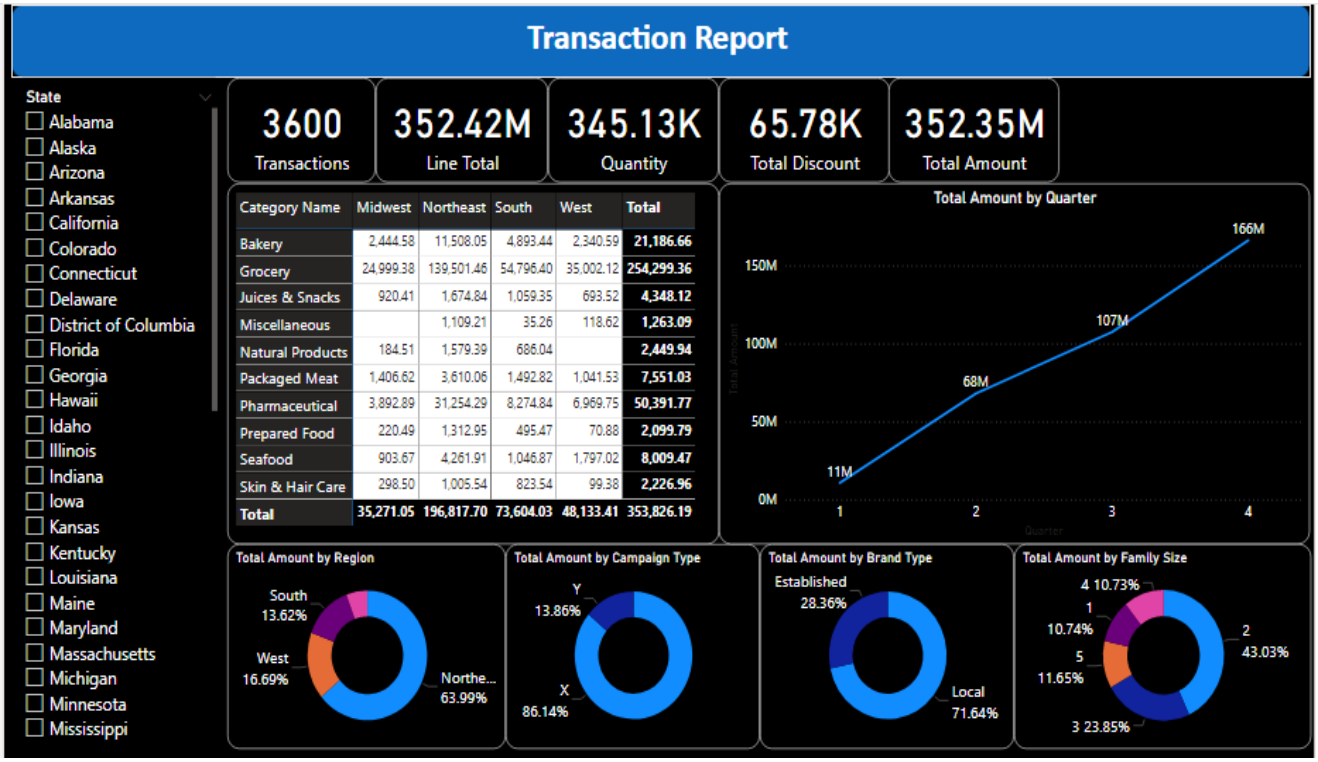


Figure 26. Power BI Visualization for Report 3

4. SSRS Reports

SQL Server Reporting Services (SSRS) is a reporting software that allows you to produce formatted reports with tables in the form of data, graph, images, and charts. These reports are hosted on a server that can be executed any time using parameters defined by the users. SSRS reports can be developed using tools like report builder and deployed in SSRS Web portal for viewing.

Report Builder:

- It is a standalone application for creating paginated reports. Once designed the report can be deployed to the report server and displayed on the SSRS web portal.

SSRS Web Portal:

- The SQL Server Report Service Web Portal is a web-based experience. In this portal users can view reports, KPIs and navigate through the elements in the report server instance. User can also use the web portal to administer a single report server instance.

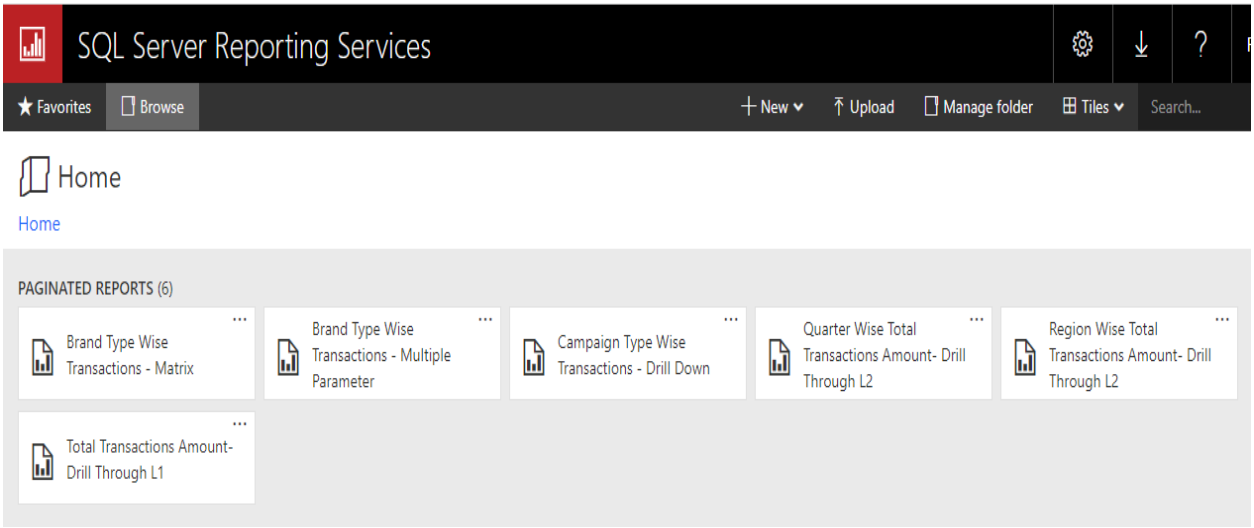


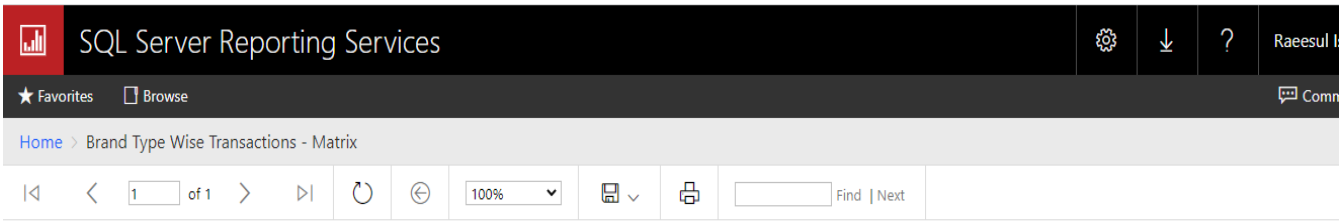
Figure 27. SSRS Web Portal Home Page

4.1. Report with Matrix

- In SSRS, Matrix is very similar to a table, but it is configured to show data grouped by columns and rows, with aggregate data at the intersection.

Report:

✓ **Line Total, Quantity, Total Discount and Total Transaction Amount**. The report contains the data of **Line Total, Quantity, Total Discount** and **Total Transaction Amount** for each **Category** for each **Brand Type**.



Brand Type wise Transactions

	Established				Local				Total			
Category Name	Line Total	Quantity	Total Discount	Total Amount	Line Total	Quantity	Total Discount	Total Amount	Line Total	Quantity	Total Discount	Total Amount
Bakery					3,743,612.39	7,227	4,369.12	3,739,243.27	**3,743,612.39**	**7,227**	**4,369.12**	**3,739,243.27**
Grocery	39,819,161.42	59,444	18,980.99	39,800,180.43	233,288,721.28	223,504	27,896.65	233,260,824.63	**273,107,882.70**	**282,948**	**46,877.64**	**273,061,005.06**
Juices & Snacks	6,804.82	54	577.38	6,227.44					**6,804.82**	**54**	**577.38**	**6,227.44**
Miscellaneous	1,733.99	18	133.22	1,600.77					**1,733.99**	**18**	**133.22**	**1,600.77**
Natural Products					3,226.10	35	769.05	2,457.05	**3,226.10**	**35**	**769.05**	**2,457.05**
Packaged Meat	7,676.06	64	885.88	6,790.18	4,659.45	40	685.33	3,974.12	**12,335.51**	**104**	**1,571.21**	**10,764.30**
Pharmaceutical	60,337,484.16	40,541	6,997.18	60,330,486.98	15,183,299.60	14,064	2,555.34	15,180,744.26	**75,520,783.76**	**54,605**	**9,552.52**	**75,511,231.24**
Prepared Food					3,601.53	28	340.53	3,261.00	**3,601.53**	**28**	**340.53**	**3,261.00**
Seafood					11,927.68	79	1,269.51	10,658.17	**11,927.68**	**79**	**1,269.51**	**10,658.17**
Skin & Hair Care	1,675.91	18	148.17	1,527.74	1,635.33	15	166.70	1,468.63	**3,311.24**	**33**	**314.87**	**2,996.37**

Figure 28. SSRS Matrix Report

SQL Query:

```
select i.BrandType, c.CategoryName, sum(ft.LineTotal) as 'Line Total',
sum(ft.Quantity) as 'Quantity', sum(ft.TotalDiscount) as 'Total Discount',
sum(ft.TotalAmount) as 'Total Amount'
from FactTransaction ft
inner join DimItem i
    on ft.ItemKey = i.ItemSK
inner join DimCategory c
    on i.CategoryKey = c.CategorySK

group by i.BrandType, c.CategoryName
```

4.2. Report with Multiple Parameters

- In SSRS, Multiparameter-values allows us to pass either one or more input parameter values to the report. Also, it offers a "Select All" option that helps to select all parameter values.

Report:

✓ **Region and State wise Quantity and Total Amount**. The report allows to select the Region and State through a drop down. When the Regions are selected, the State belonging to the particular Region will be filtered and allowed for selection. On selection of view report, the report displays the Quantity and the Total Amount for each Brand Type, grouped according to Item Category, and the selected Regions and States accepted as parameters.

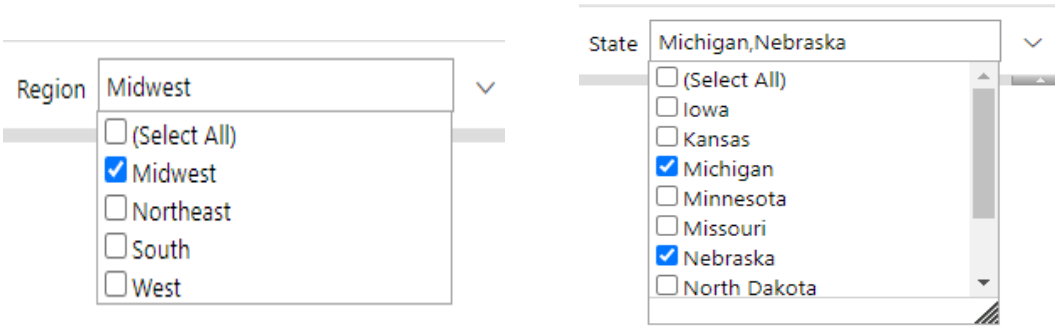
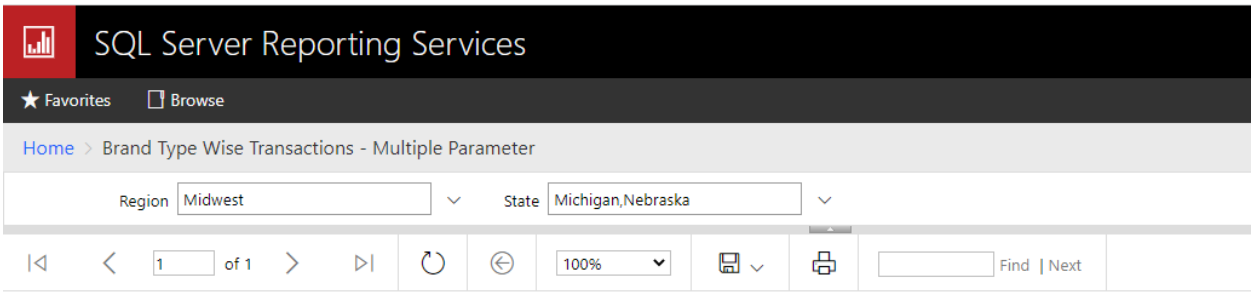


Figure 29. State and Region Selection



Brand Type wise Transactions

				Established		Local		Total	
Region	**State**	**City**	**Category Name**	**Quantity**	**Total Amount**	**Quantity**	**Total Amount**	**Quantity**	**Total Amount**
⊟Midwest	⊟Michigan	⊟Detroit	Bakery			2	180.94	**2**	**180.94**
			Grocery	3	883.01	8	499.03	**11**	**1,382.04**
			Packaged Meat	1	88.69			**1**	**88.69**
			Prepared Food			1	43.10	**1**	**43.10**
			Seafood			2	36.33	**2**	**36.33**
			Total	4	971.70	13	759.40	17	1,731.10
		Total		4	971.70	13	759.40	17	1,731.10
	⊞Nebraska	Total		23	1,359.99	44	4,870.28	67	6,230.27
	Total			27	2,331.69	57	5,629.68	84	7,961.37
Total				**27**	**2,331.69**	**57**	**5,629.68**	**84**	**7,961.37**

Figure 30. SSRS Multiple Parameter Report

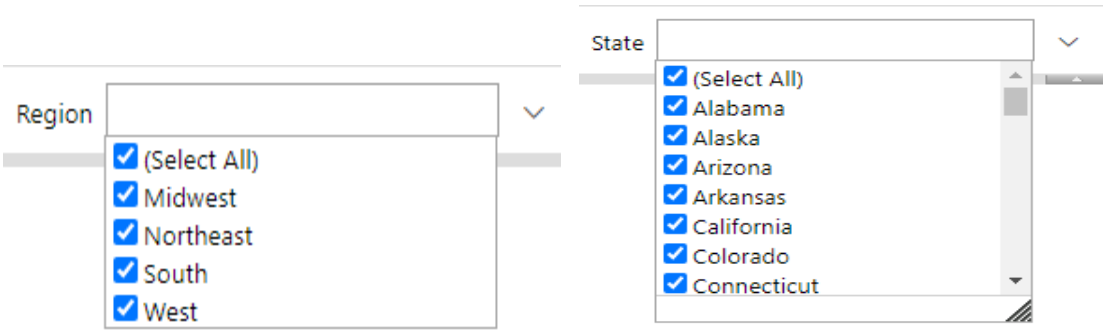
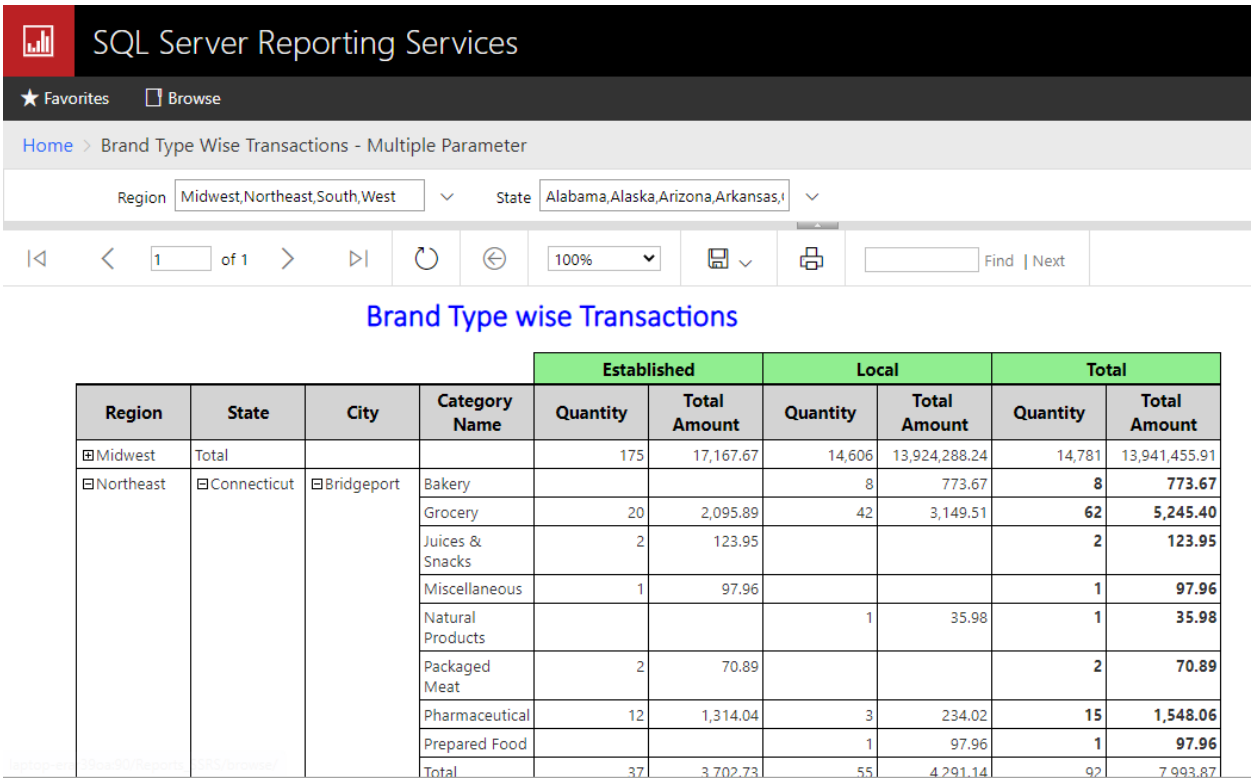


Figure 31. State and Region All Selected



				Established		Local		Total	
Region	State	City	Category Name	Quantity	Total Amount	Quantity	Total Amount	Quantity	Total Amount
⊞Midwest	Total			175	17,167.67	14,606	13,924,288.24	14,781	13,941,455.91
⊟Northeast	⊟Connecticut	⊟Bridgeport	Bakery			8	773.67	**8**	**773.67**
			Grocery	20	2,095.89	42	3,149.51	**62**	**5,245.40**
			Juices & Snacks	2	123.95			**2**	**123.95**
			Miscellaneous	1	97.96			**1**	**97.96**
			Natural Products			1	35.98	**1**	**35.98**
			Packaged Meat	2	70.89			**2**	**70.89**
			Pharmaceutical	12	1,314.04	3	234.02	**15**	**1,548.06**
			Prepared Food			1	97.96	**1**	**97.96**
			Total	37	3,702.73	55	4,291.14	92	7,993.87

Figure 32. SSRS Multiple Parameter All Selected Report

SQL Query:

```
---- Transaction Dataset ---
select d.Region, d.State, d.City, i.BrandType, c.CategoryName, sum(ft.Quantity)
as 'Quantity',
sum(ft.TotalAmount) as 'Total Amount'
from FactTransaction ft
inner join DimCustomer ct
   on ft.CustomerKey = ct.CustomerSK
inner join DimDistrict d
   on ct.DistrictKey = d.DistrictSK
inner join DimItem i
```

```
    on ft.ItemKey = i.ItemSK
inner join DimCategory c
    on i.CategoryKey = c.CategorySK

where State in (@State)
group by d.Region, d.State, d.City, i.BrandType, c.CategoryName
```

---- Getting Region ---

```
select d.State, d.Region, d.City, ct.DistrictKey, ct.CustomerSK,
ft.CustomerKey
from DimDistrict d
inner join DimCustomer ct
    on d.DistrictSK = ct.DistrictKey
inner join FactTransaction ft
    on ct.CustomerSK = ft.CustomerKey

where d.Region in (@Region)
```

---- Transaction Dataset ---

```
select distinct(State)
from DimDistrict
where d.Region in (@Region)
```

---- Transaction Dataset ---

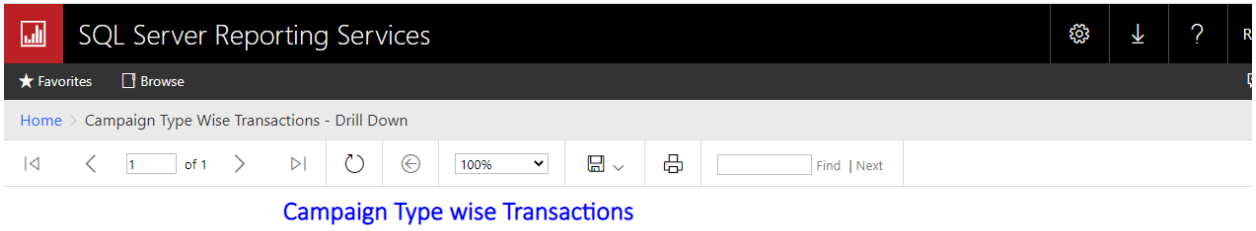
```
select distinct(Region)
from DimDistrict
```

4.3. Drill – Down Report

- In SSRS reports, Drill-down allows expand or collapse a section of a report to show or hide detail data. We can expand the data using the plus button and collapse data using the minus button.

Report:

✓ **Drill down through Regions, States, and Cities**. The feature allows a user to **view Line Total, Total Discount and Total Amount** for each **Campaign Type** based on Region, State, and Cities. Initially the Totals for a Region will be displayed, which must then be expanded to view the same for a State and a City.



Campaign Type wise Transactions

			X			Y			Total		
Region	State	City	Line Total	Total Discount	Total Amount	Line Total	Total Discount	Total Amount	Line Total	Total Discount	Total Amount
⊞Midwest	Total		13,932,057.30	3,843.38	13,928,213.92	14,721.99	1,480.00	13,241.99	13,946,779.29	5,323.38	13,941,455.91
⊞Northeast	Total		126,796,879.25	25,149.67	126,771,729.58	29,876,118.52	8,009.37	29,868,109.15	156,672,997.77	33,159.04	156,639,838.73
⊞South	Total		29,326,281.15	10,334.73	29,315,946.42	4,033,204.05	3,710.16	4,029,493.89	33,359,485.20	14,044.89	33,345,440.31
⊞West	Total		40,845,341.94	5,720.18	40,839,621.76	17,566.32	2,393.30	15,173.02	40,862,908.26	8,113.48	40,854,794.78
Total			**210,900,559.64**	**45,047.96**	**210,855,511.68**	**33,941,610.88**	**15,592.83**	**33,926,018.05**	**244,842,170.52**	**60,640.79**	**244,781,529.73**

Figure 33. SSRS Report Before Drill-down

Campaign Type wise Transactions

			X			Y			Total		
Region	State	City	Line Total	Total Discount	Total Amount	Line Total	Total Discount	Total Amount	Line Total	Total Discount	Total Amount
⊟Midwest	⊟Iowa	Des Moines	6,449.32	787.20	5,662.12	2,446.04	149.24	2,296.80	**8,895.36**	**936.44**	**7,958.92**
		Total	6,449.32	787.20	5,662.12	2,446.04	149.24	2,296.80	8,895.36	936.44	7,958.92
	⊞Kansas	Total	1,702.62	242.56	1,460.06	1,095.67	86.92	1,008.75	2,798.29	329.48	2,468.81
	⊞Michigan	Total	1,625.32	97.61	1,527.71	292.08	88.69	203.39	1,917.40	186.30	1,731.10
	⊞Minnesota	Total	92.97	0.00	92.97	141.06	0.00	141.06	234.03	0.00	234.03
	⊞Missouri	Total	1,428.02	398.96	1,029.06	1,559.79	191.28	1,368.51	2,987.81	590.24	2,397.57
	⊞Nebraska	Total	4,354.88	653.97	3,700.91	2,805.75	276.39	2,529.36	7,160.63	930.36	6,230.27
	⊞North Dakota	Total	13,903,476.25	369.39	13,903,106.86				13,903,476.25	369.39	13,903,106.86
	⊞Ohio	Total	1,906.02	196.60	1,709.42	914.35	29.92	884.43	2,820.37	226.52	2,593.85
	⊞South Dakota	Total	2,844.24	376.87	2,467.37	2,867.03	390.41	2,476.62	5,711.27	767.28	4,943.99
	⊞Wisconsin	Total	8,177.66	720.22	7,457.44	2,600.22	267.15	2,333.07	10,777.88	987.37	9,790.51
	Total		13,932,057.30	3,843.38	13,928,213.92	14,721.99	1,480.00	13,241.99	13,946,779.29	5,323.38	13,941,455.91

Figure 34. SSRS Report After Drilled down

SQL Query:

```
select d.Region, d.State, d.City, c.CampaignType,
sum(ft.LineTotal) as 'Line Total', sum(ft.TotalDiscount) as 'Total Discount',
sum(ft.TotalAmount) as 'Total Amount'
from FactTransaction ft
inner join DimCustomer ct
   on ft.CustomerKey = ct.CustomerSK
inner join DimDistrict d
   on ct.DistrictKey = d.DistrictSK
inner join DimTrain t
   on ft.TrainKey = t.TrainSK
inner join DimCampaign c
   on t.CampaignKey = c.CampaignSK

group by d.Region, d.State, d.City, c.CampaignType
```

4.4. Drill – Through Report

- In SSRS, a drill through allows a user click on a link or an area in a chart with summarized data, which then opens a separate, related report to show detailed data. Drill through reports commonly contain details about an item that is contained in an original summary report. The data in the drill through report is not retrieved until the user clicks the link in the main report.

Report:

✓ **Level 1** – Region and Quarter wise Line Total and Total Amounts.
✓ **Level 2** – Region wise Total Amount Report & Quarter wise drill through.

- This report displays two column charts. 1st chart represents the data of Total Amounts based on Brand Type for each **Region**. 2nd chart represents the Total Amounts based on Brand Type for each **Quarter**.

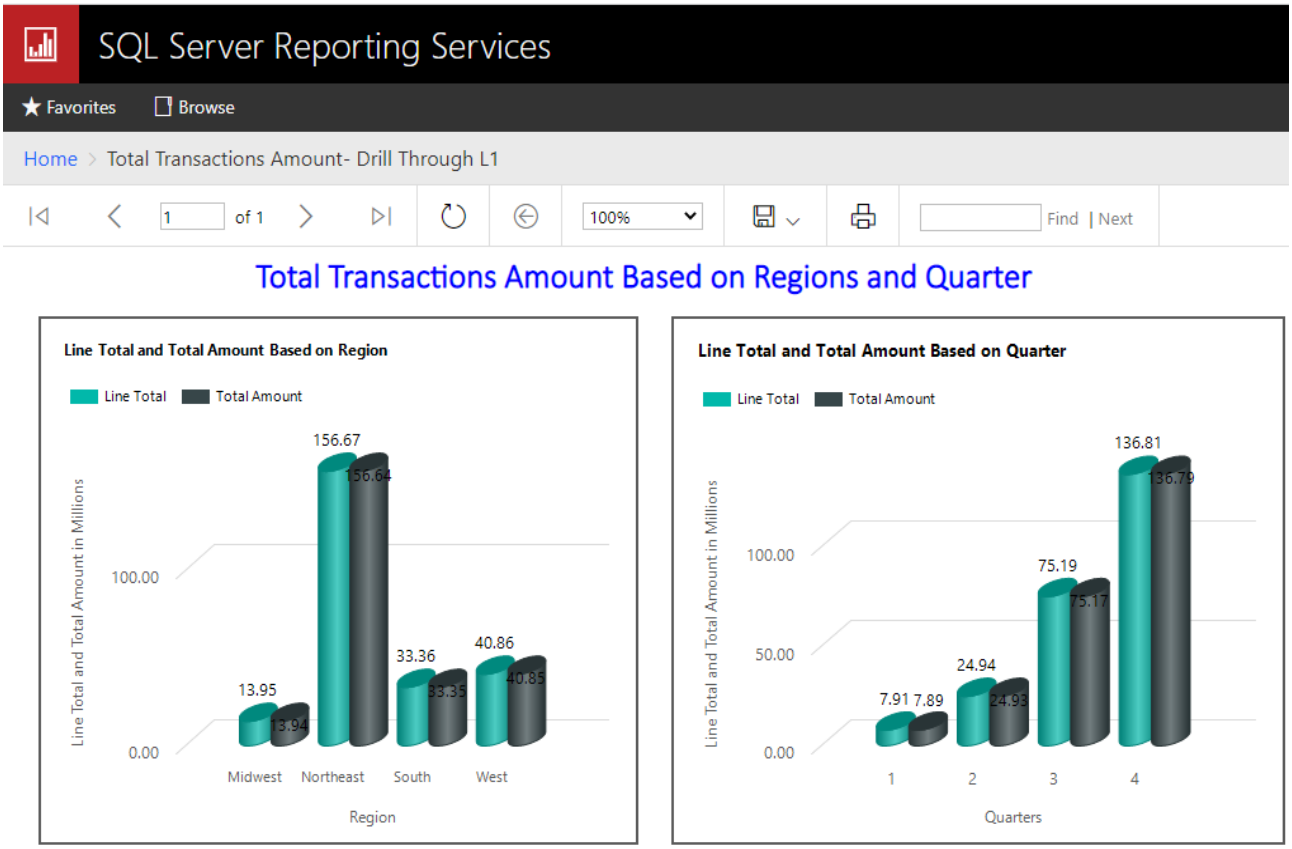


Figure 35. The Level 1 SSRS Report for Drill – Through

- When the user clicks on a bar in the report contain states, it will display a detailed report that provides the same information based on each region belonging to the state. The below diagram shows details when the Region **'Northeast'** is clicked.

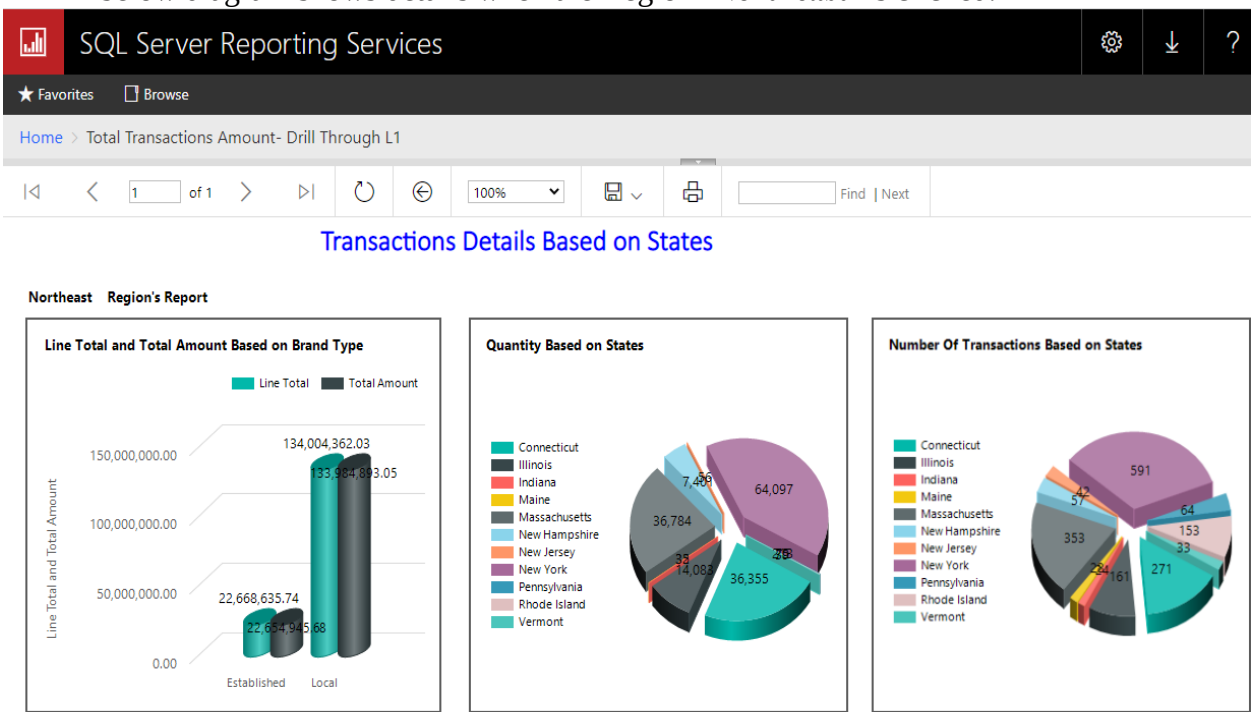


Figure 36. Drilled - Through SSRS Report for Region Northeast

- When the user clicks on a bar in the report containing Quarters, it will display a detailed report that provides the information about particular year. The below diagram shows details when the Quarter **3** is clicked.

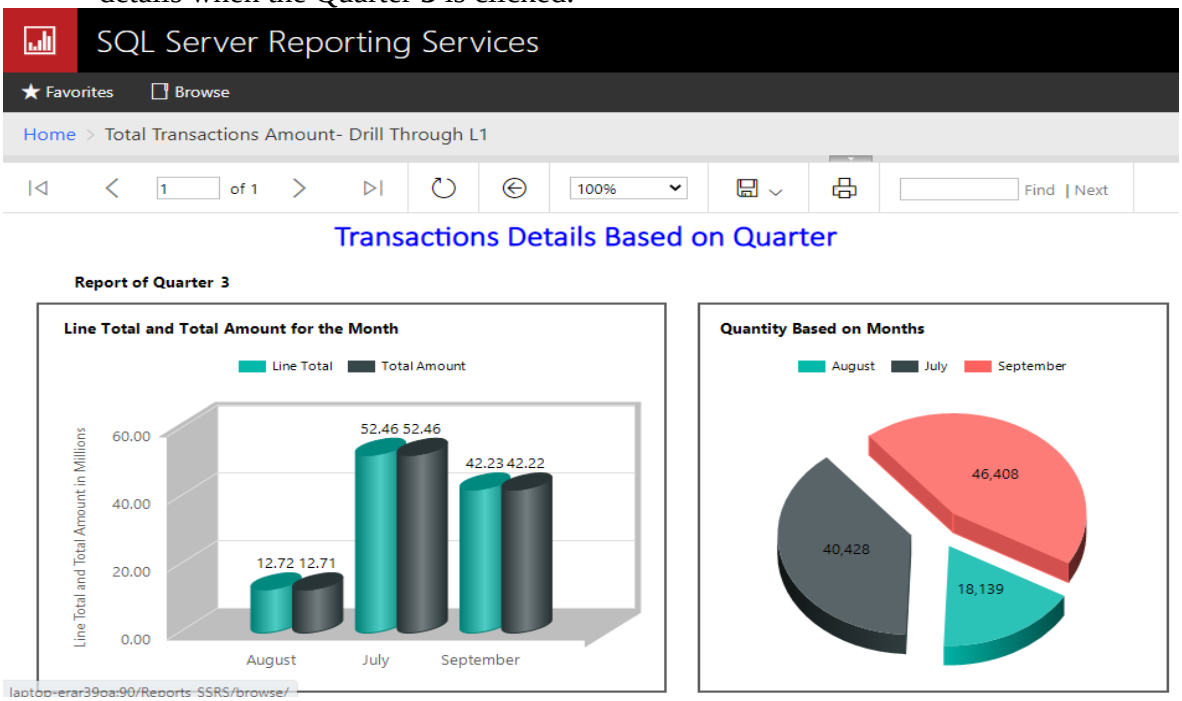


Figure 37. Drilled – Through SSRS Report for Quarter 3

SQL Query:

```
---- L1 ------
select d.Region, d.State, d.City, dt.Quarter, i.BrandType,
sum(ft.LineTotal) as 'Line Total',sum(ft.TotalAmount) as 'Total Amount'
from FactTransaction ft
inner join DimCustomer ct
   on ft.CustomerKey = ct.CustomerSK
inner join DimDistrict d
   on ct.DistrictKey = d.DistrictSK
inner join DimDate dt
   on ft.DateKey = dt.DateKey
inner join DimItem i
   on ft.ItemKey = i.ItemSK

group by d.Region, d.State,  d.City, dt.Quarter, i.BrandType
```

```
---- State Wise – L2 ------
select d.State, d.Region, d.City, c.CategoryName, i.BrandType,
sum(ft.Quantity) as 'Quantity',sum(ft.LineTotal) as 'Line Total',
sum(ft.TotalAmount) as 'Total Amount',
count(ft.TransactionID) as 'No of Transaction'
from FactTransaction ft
inner join DimCustomer ct
   on ft.CustomerKey = ct.CustomerSK
inner join DimDistrict d
   on ct.DistrictKey = d.DistrictSK
inner join DimItem i
   on ft.ItemKey = i.ItemSK
inner join DimCategory c
   on i.CategoryKey = c.CategorySK

where d.Region in (@Region)
group by d.State, d.Region, d.City, c.CategoryName,i.BrandType
```

```
---- Quarter Wise – L2 ------
select d.Quarter, d.MonthName, c.CategoryName, i.BrandType,
sum(ft.Quantity) as 'Quantity',
sum(ft.LineTotal) as 'Line Total',
sum(ft.TotalAmount) as 'Total Amount'
from FactTransaction ft
inner join DimDate d
   on ft.DateKey = d.DateKey
inner join DimItem i
   on ft.ItemKey = i.ItemSK
inner join DimCategory c
   on i.CategoryKey = c.CategorySK

where d.Quarter in (@Quarter)
group by d.Quarter, d.MonthName, c.CategoryName, i.BrandType
```

5. References

[1]. https://www.sqlshack.com/analysis-services-ssas-multidimensional-design-tips-relations-hierarchies/#:~:text=sample%20OLAP%20cube.,Hierarchies,a%20certain%20drill%2Ddown%20behavior

[2]. https://www.guru99.com/online-analytical-processing.html

[3]. Working with Multi-Valued Parameters in SSRS, https://www.interfacett.com/blogs/using-parameters-with-multiple-values-in-sql-server-reporting-services/

[4]. "SSRS - How to build Drill-Through Report in SSRS," https://www.youtube.com/watch?v=sPl-Zp0X5Pk&feature=youtu.be