Sri Lanka Institute of Information Technology

Assignment II

Data Warehouse & Business Intelligence

2022

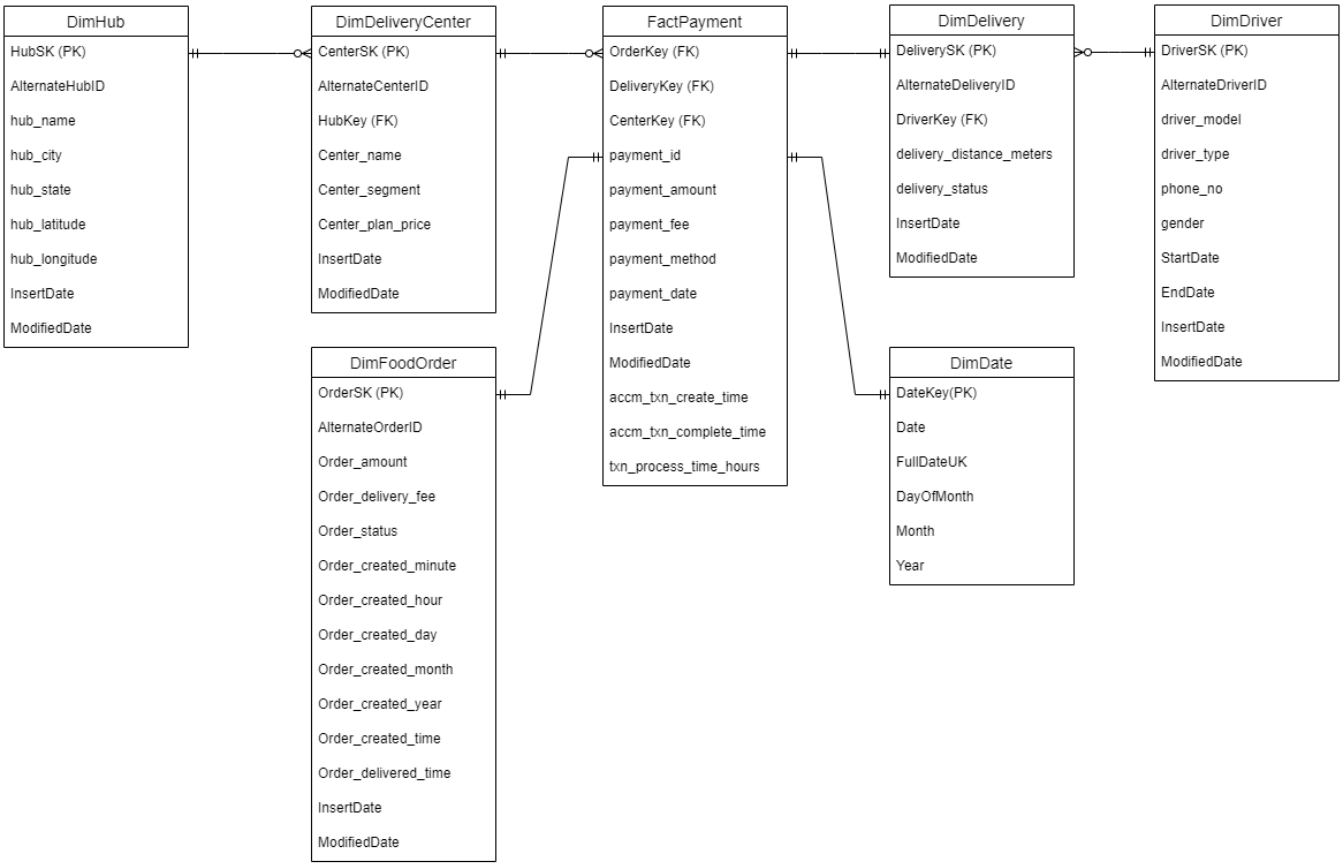
Submitted By
Ariyasinghe P.A.D.N.I
IT20033828

Table of Contents

1. Data Source for Assignment-2

1.1. Dimensional Model Schema:

The data warehouse database solution that created during the first assignment was selected as the data source for the second assignment. The data warehouse solution was implemented using a snowflake schema with one fact table and 6-dimension tables in total. Given below is a Dimensional model designed to showcase the structure of the data warehouse database solution through explaining the relationships among the fact and dimension tables.



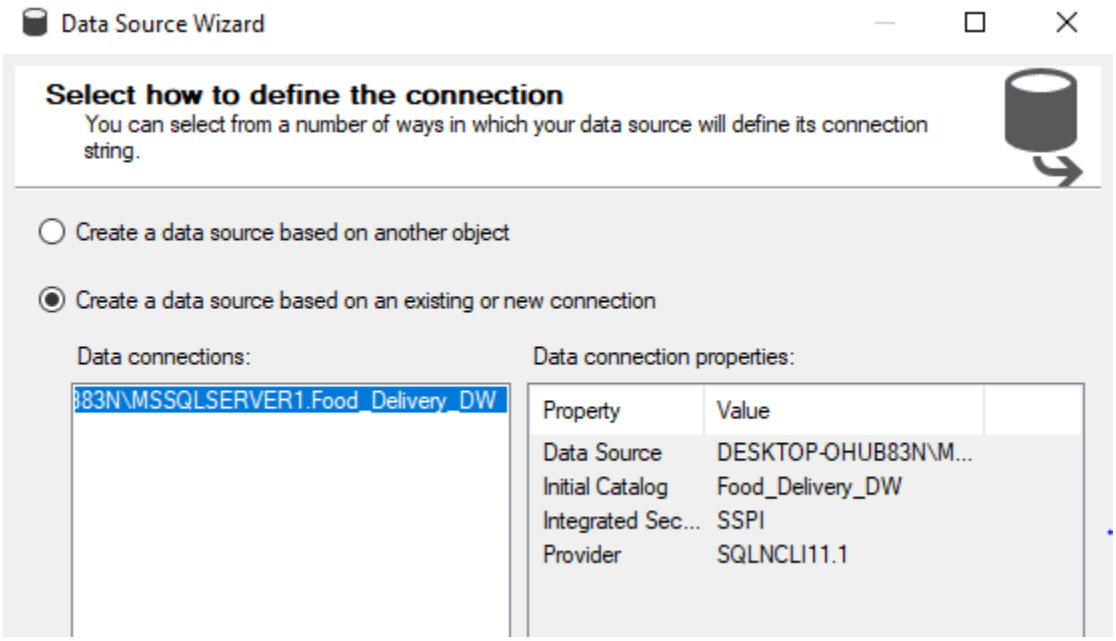
1.2. Details of fact and dimension tables:

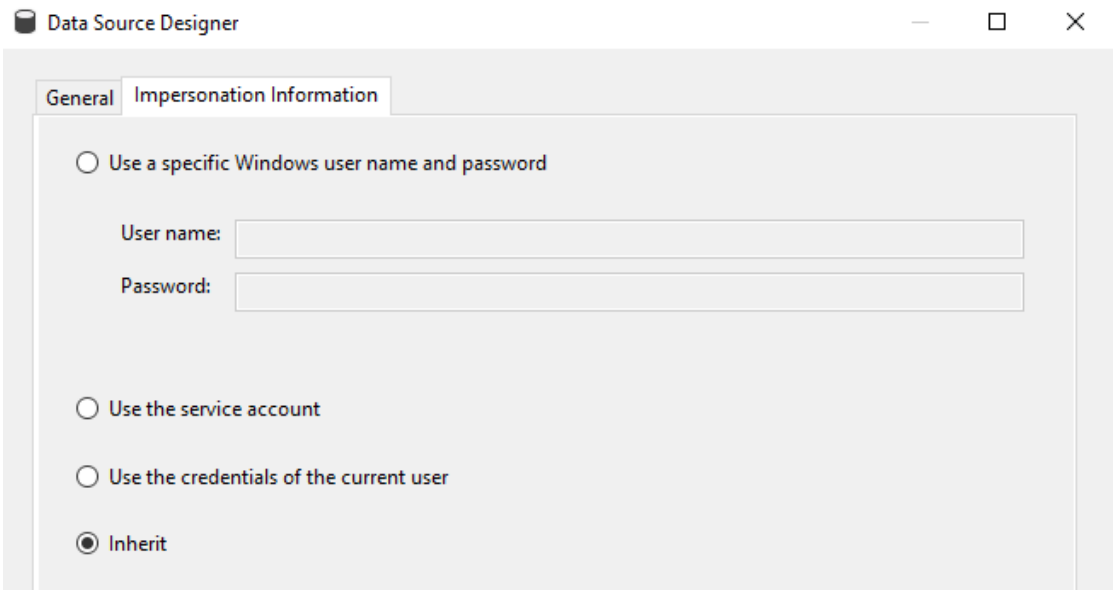
Dimension Name	Truncate Before Update	Dimension Attributes	Derived Attribu	Data Type	Key Column	Derived Logic
FactPayment	No	OrderKey	N	int	FK	
	No	CenterKey	N	smallint	FK	
	No	DeliveryKey	N	int	FK	
	No	Payment_id	N	int		
	No	Payment_amount	N	Float		
	No	Payment_fee	N	Float		
	No	Payment_method	N	nvarchar(50)		
	No	Payment_date	N	datetime		
	No	InsertDate	N	datetime		
	No	ModifiedDate	N	datetime		
	No	accm_txn_created_time	N	datetime		
	No	accm_txn_complete_time	N	datetime		
	No	txn_process_time_hours	N	int		[accm_txn_complete_time] - [accm_txn_create_time]
DimDeliveries	No	DeliverySK	N	int	Y	
	No	AlternateDeliveryID	N	int		
	No	Driver_id	N	int	FK	
	No	Delivery_distance	N	int		
	No	Delivery_status	N	nvarchar(50)		
	No	InsertDate	N	datetime		
	No	ModifiedDate	N	datetime		
DimFoodOrders	No	OrderSK	N	int	Y	
	No	AlternateOrderID	N	int		
	No	Order_status	N	nvarchar(50)		
	No	Order_amount	N	float		
	No	Order_delivery_fee	N	float		
	No	Order_created_hour	N	tinyint		
	No	Order_created_minute	N	tinyint		
	No	Order_created_day	N	tinyint		
	No	Order_created_year	N	smallint		
	No	Order_created_time	N	Datetime2		
	No	Order_delivered_time	N	Datetime2		
	No	InsertDate	N	datetime		
	No	ModifiedDate	N	datetime		
DimHubs	No	HubSK	N	tinyint	Y	
	No	AlternateHubID	N	tinyint		
	No	Hub_name	N	nvarchar(50)		
	No	Hub_city	N	nvarchar(50)		
	No	Hub_state	N	nvarchar(50)		
	No	Hub_latitude	N	float		
	No	Hub_longitude	N	float		
	No	InsertDate	N	datetime		
	No	ModifiedDate	N	datetime		
DimDeliveryCenters	No	CenterSK	N	smallint	Y	
	No	AlternateCenterID	N	smallint		
	No	HubKey	N	tinyint	FK	
	No	Center_name	N	nvarchar(50)		
	No	Center_segment	N	nvarchar(50)		
	No	Center_plan_price	N	float		
	No	InsertDate	N	datetime		
	No	ModifiedDate	N	datetime		
DimDrivers	No	DriverSK	N	int	Y	
	No	AlternateDriverID	N	int		
	No	driver_modal	N	nvarchar(50)		
	No	driver_type	N	nvarchar(50)		
	No	phone_no	N	nvarchar(50)		
	No	gender	N	nvarchar(50)		
	No	StartDate	N	datetime		
	No	EndDate	N	datetime		
	No	InsertDate	N	datetime		
	No	ModifiedDate	N	datetime		
DimDate	No	DateKey	N	int	FK	
		More Attributes...	...	...		

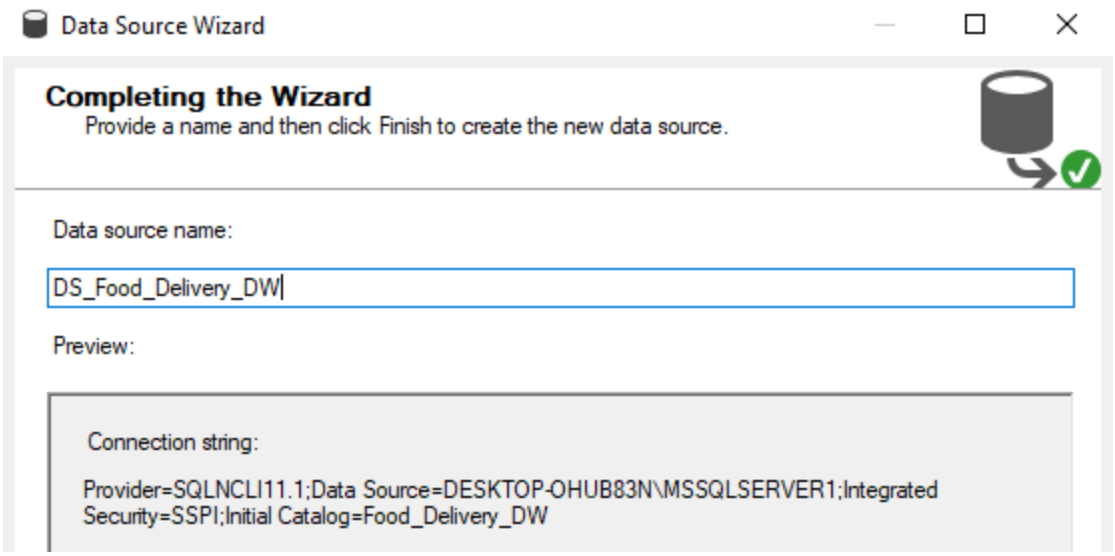
2. SSAS Cube Implementation

To implement the SSAS Cube, first, an “Analysis Services Multidimensional and Data Mining Project” was created under the name of “FoodDelivery_SSAS”.

After project was created successfully, new Data Source was added using Data Source Wizard through a connection that was established to the existing Data Warehouse database in the localhost of the machine.

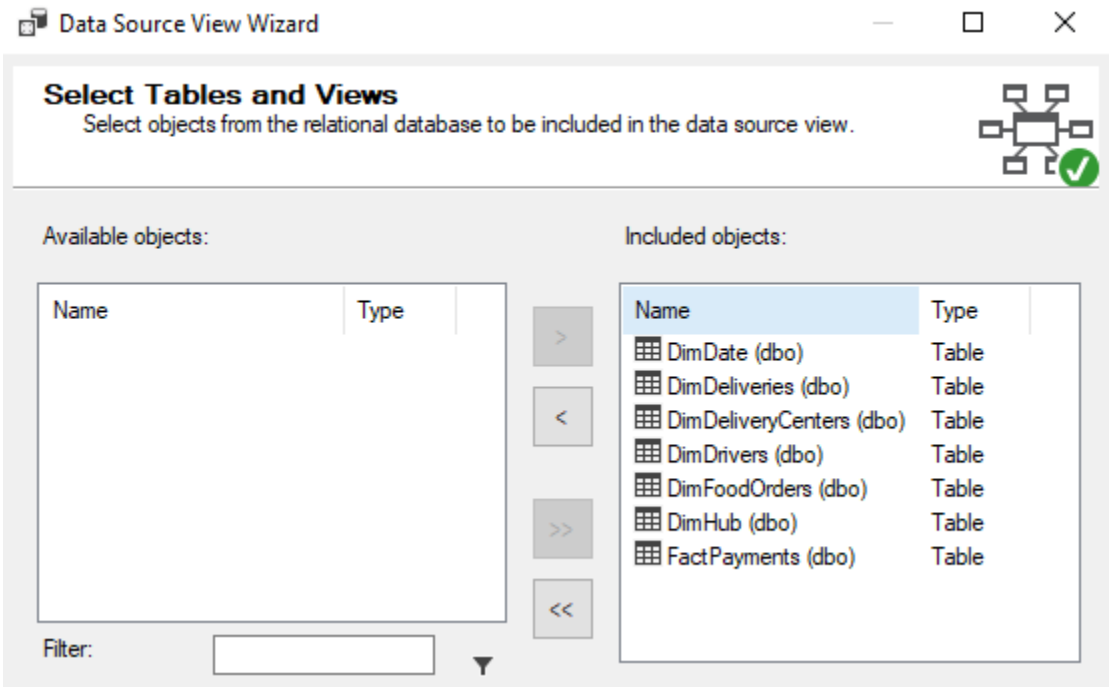


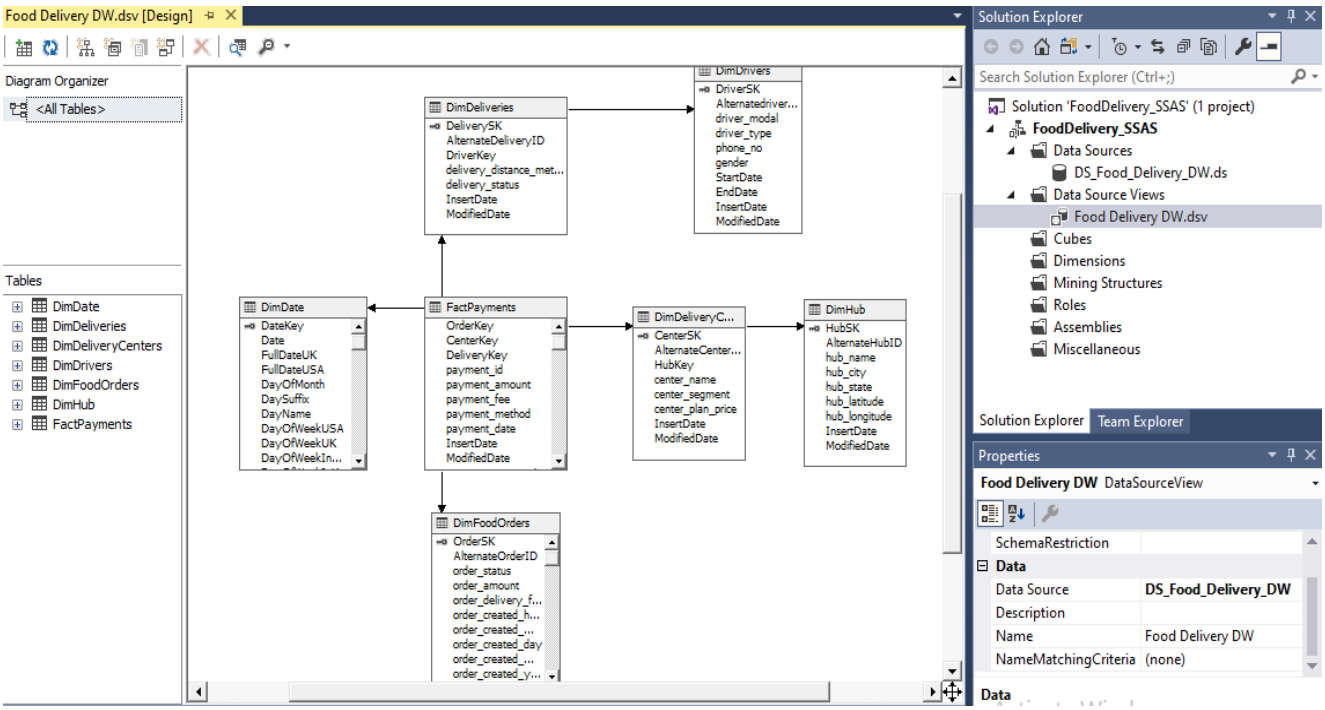




2.1. Creating the Data Source View:

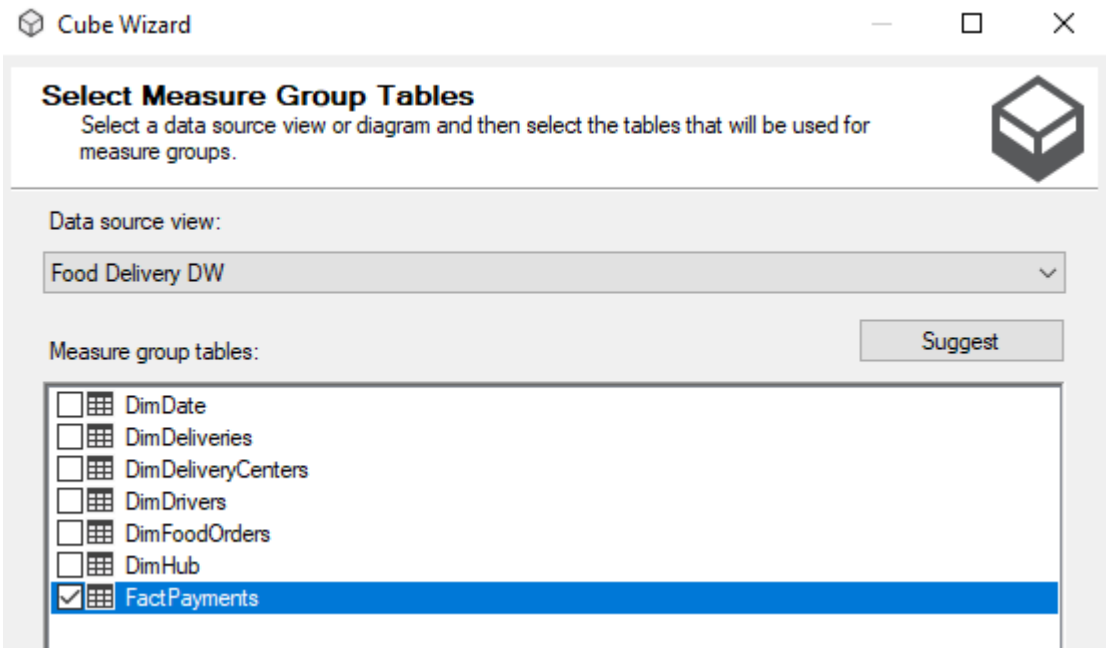
Based on the Data Source created above, a data source view was created using Data Source View Wizard. All Dimensions and Fact tables were added to Included Objects of the Data Source View as all of them are necessary for the following steps.

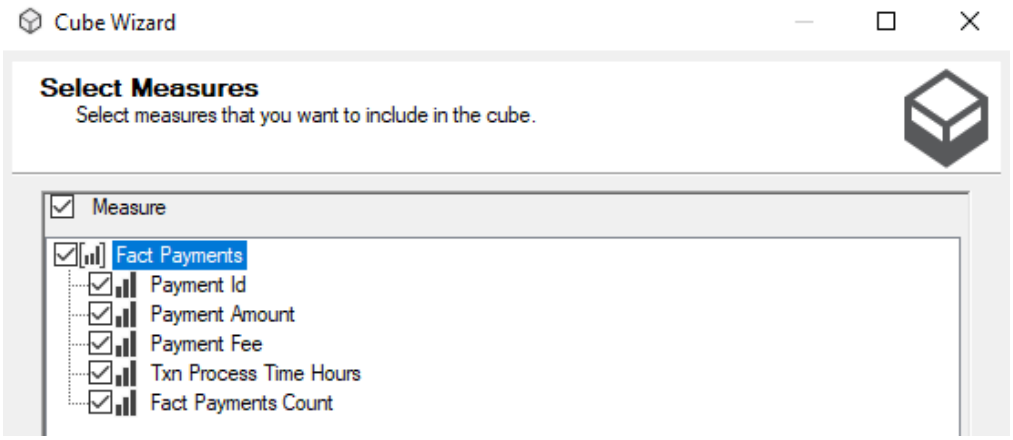


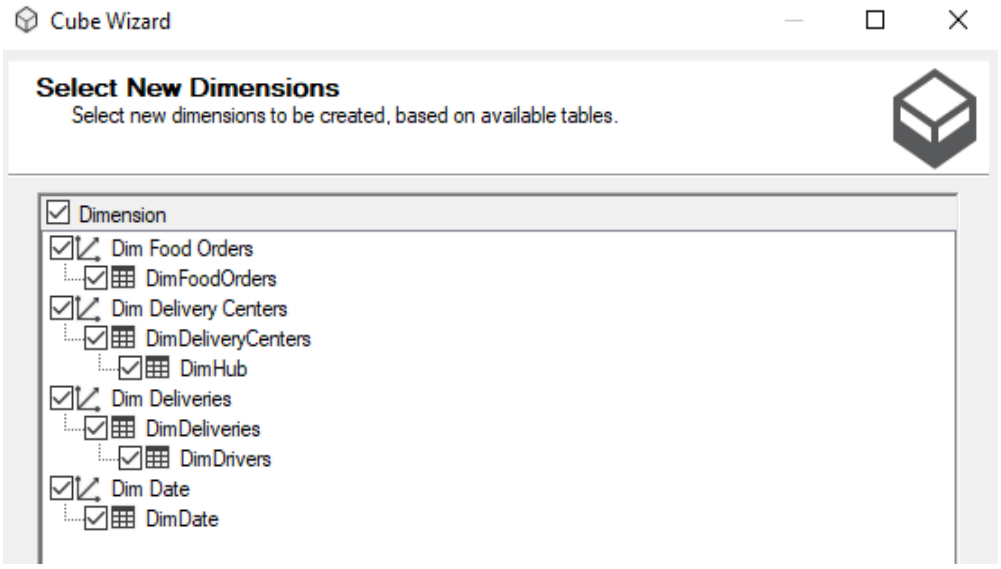


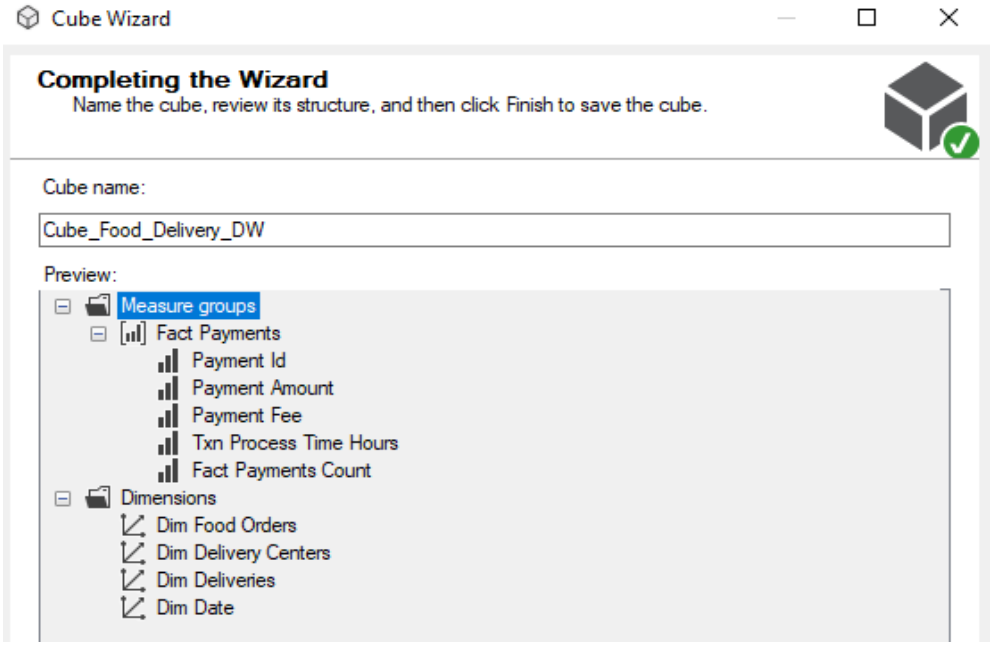
2.2. Implementing the SSAS Cube:

Cube Wizard was used to create the cube with existing tables which are imported and linked in the previous data source view creation step. FactPayments table was selected as the Measure group table.

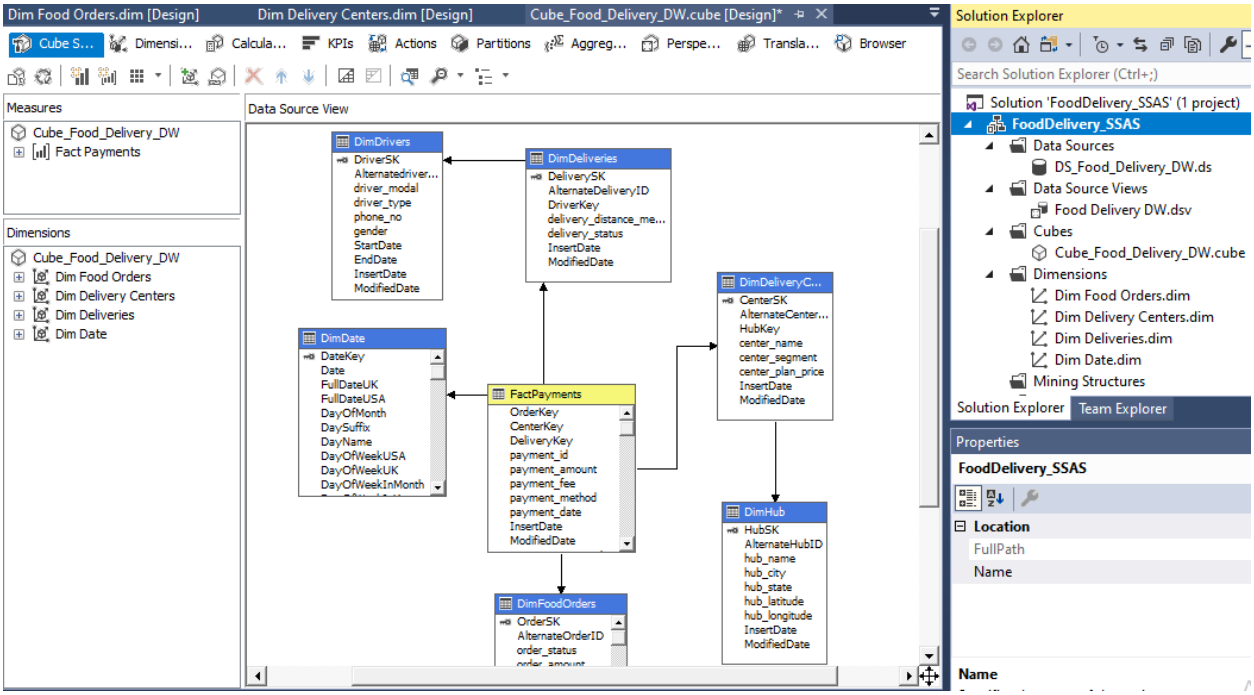



Cube Wizard
Select Measures
Select measures that you want to include in the cube.
Measure
Fact Payments
Payment Id
Payment Amount
Payment Fee
Txn Process Time Hours
Fact Payments Count

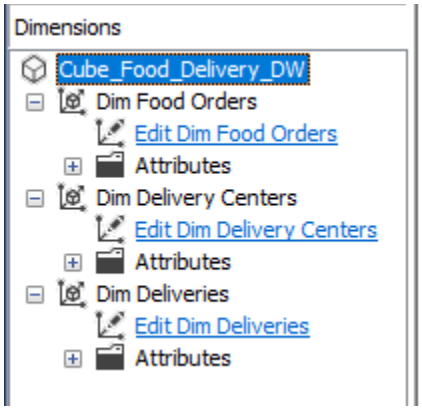

Cube Wizard
Select New Dimensions
Select new dimensions to be created, based on available tables.
Dimension
Dim Food Orders
DimFoodOrders
Dim Delivery Centers
DimDeliveryCenters
DimHub
Dim Deliveries
DimDeliveries
DimDrivers
Dim Date
DimDate


Cube Wizard
Completing the Wizard
Name the cube, review its structure, and then click Finish to save the cube.
Cube name:
Cube_Food_Delivery_DW
Preview:
Measure groups
Fact Payments
Payment Id
Payment Amount
Payment Fee
Txn Process Time Hours
Fact Payments Count
Dimensions
Dim Food Orders
Dim Delivery Centers
Dim Deliveries
Dim Date

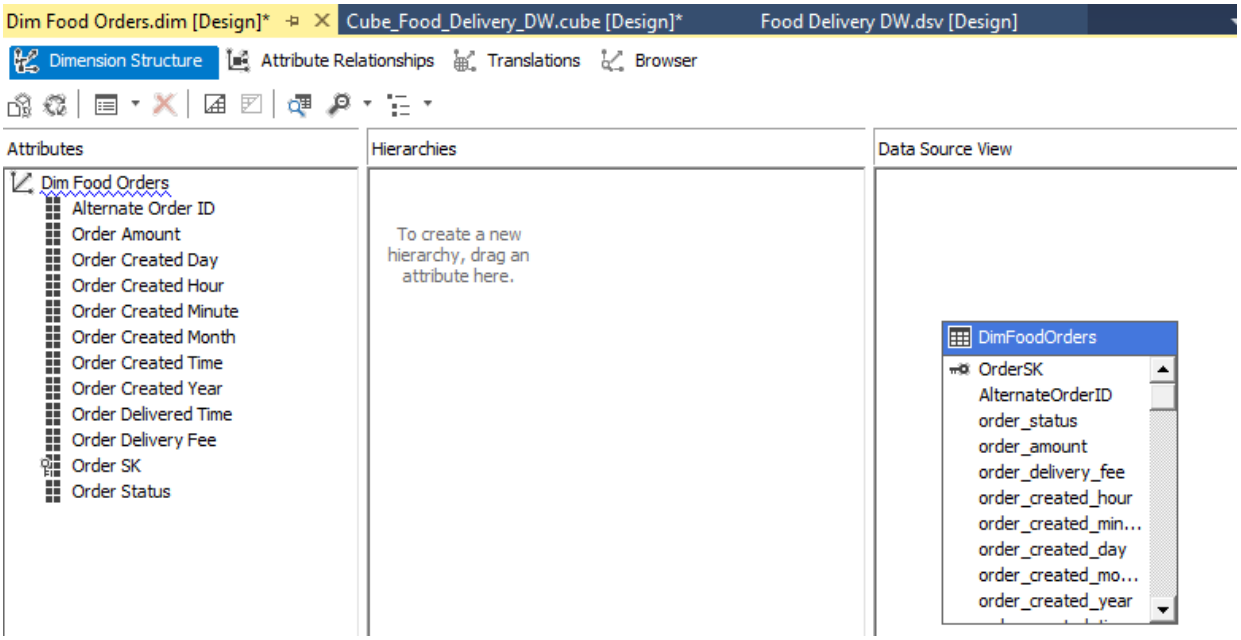
As it is indicated in the above images, Measure groups, Measures and dimensions were selected appropriately based on the existing data warehouse tables and their columns, and finally, the cube was created.



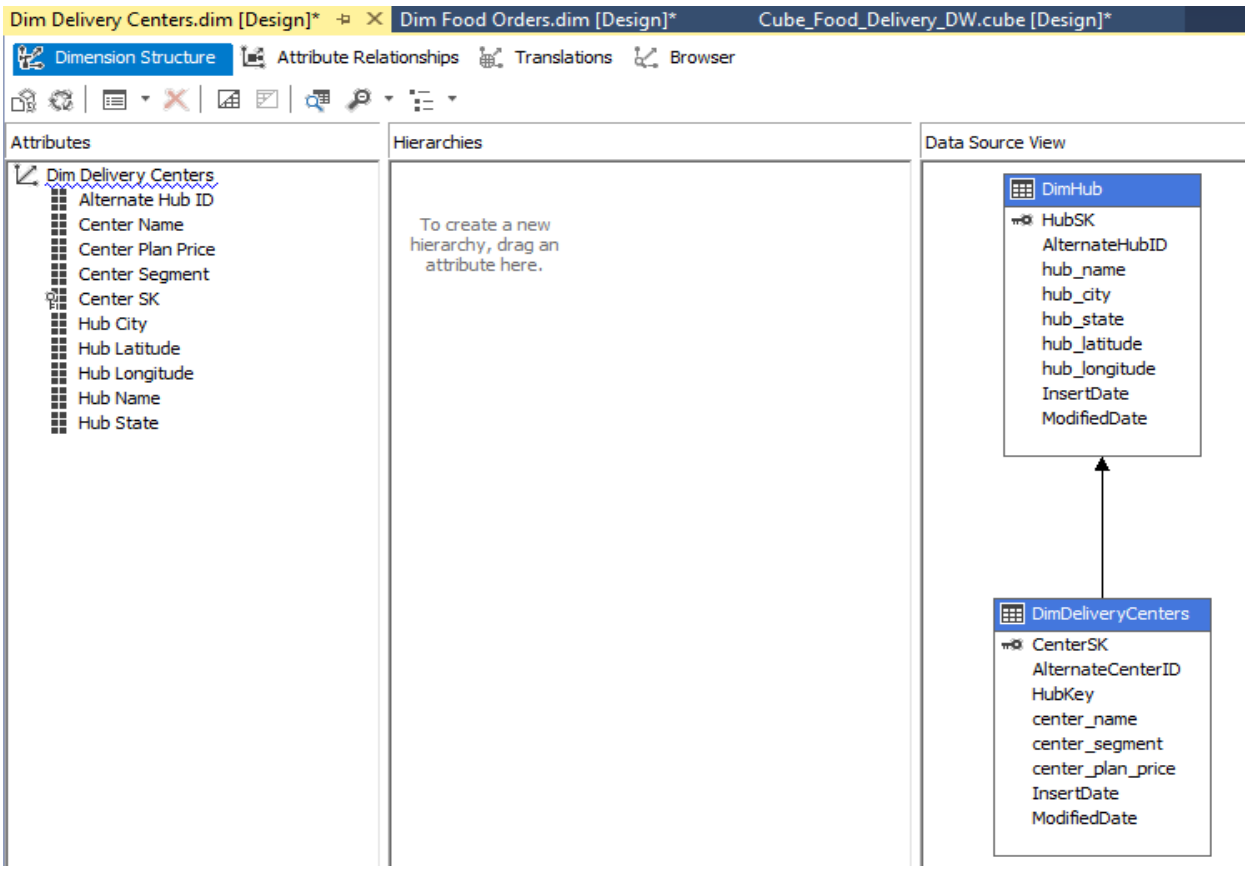
After the cube was created initially, missing attributes in the dimension tables were added using Edit dimension links in the Dimensions tab of the 'cube structure' main tab in the cube designer.



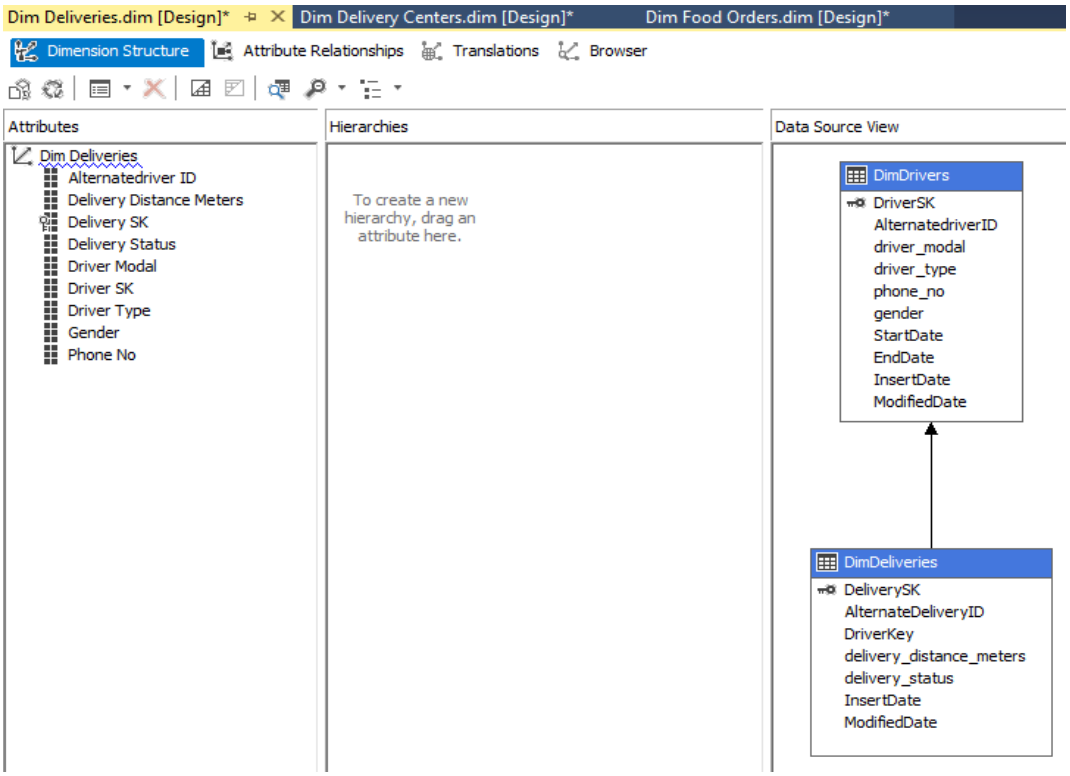
2.2.1. DimFoodOrders dimension after adding the necessary attributes for the cube:



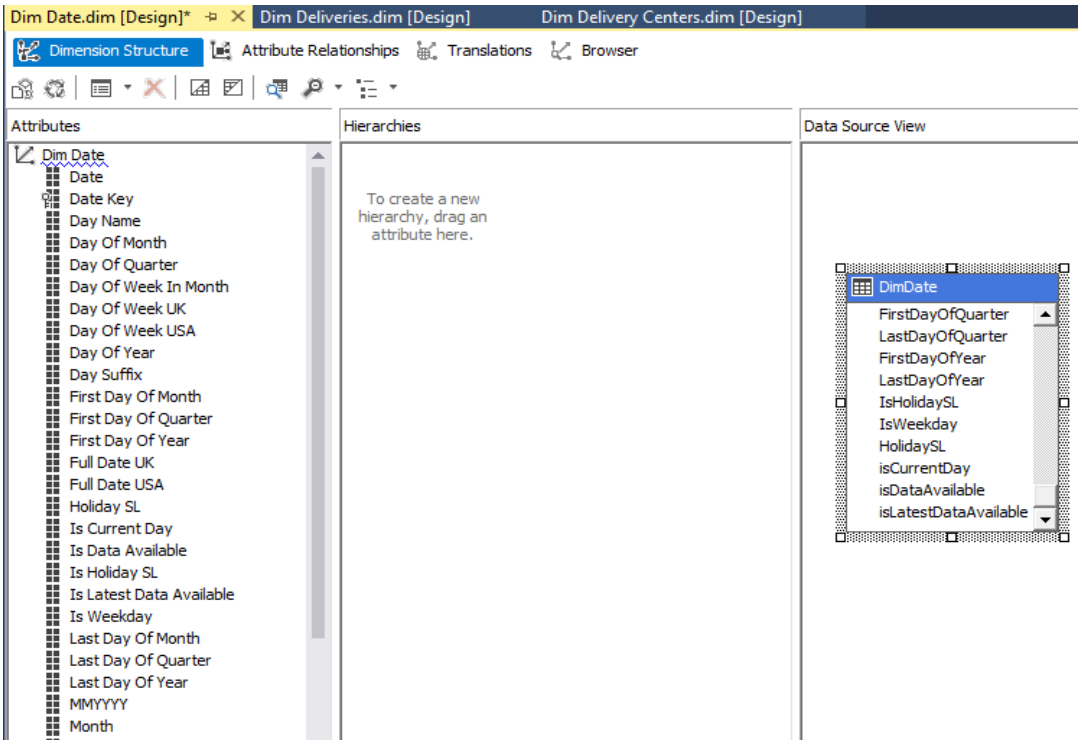
2.2.2. DimDeliveryCenters dimension after adding the necessary attributes for the cube:



2.2.3. DimDeliveries dimension after adding the necessary attributes for the cube:



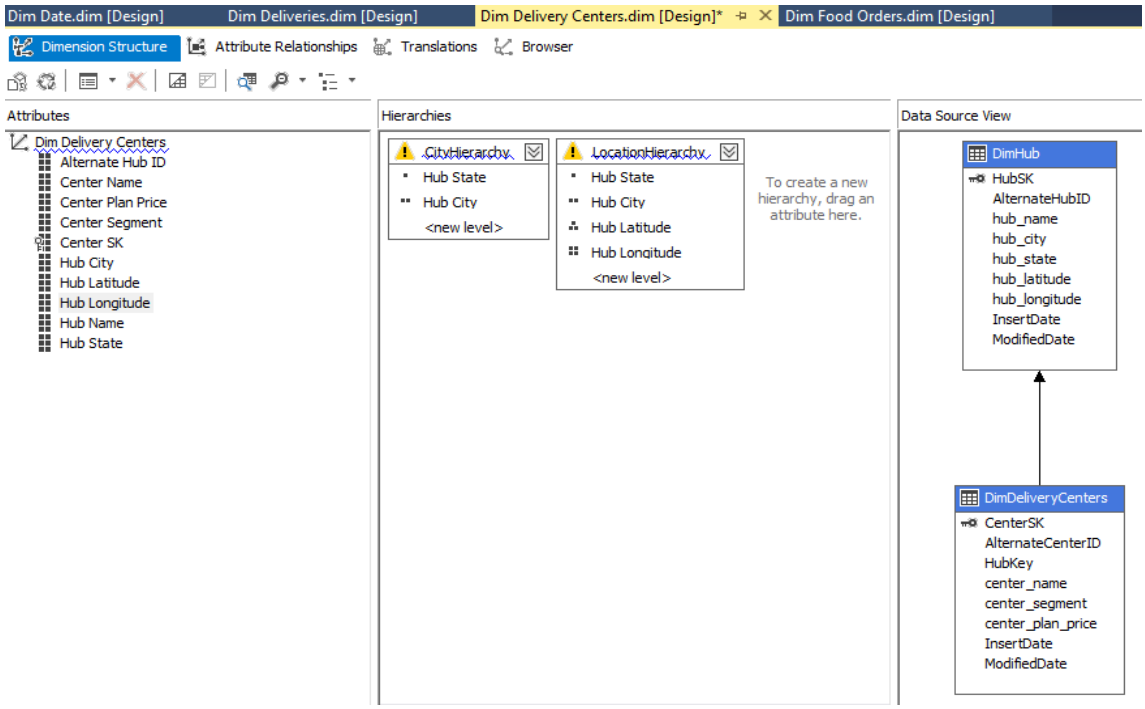
2.2.4. DimDate dimension after adding the necessary attributes for the cube:



2.3. Adding Hierarchies:

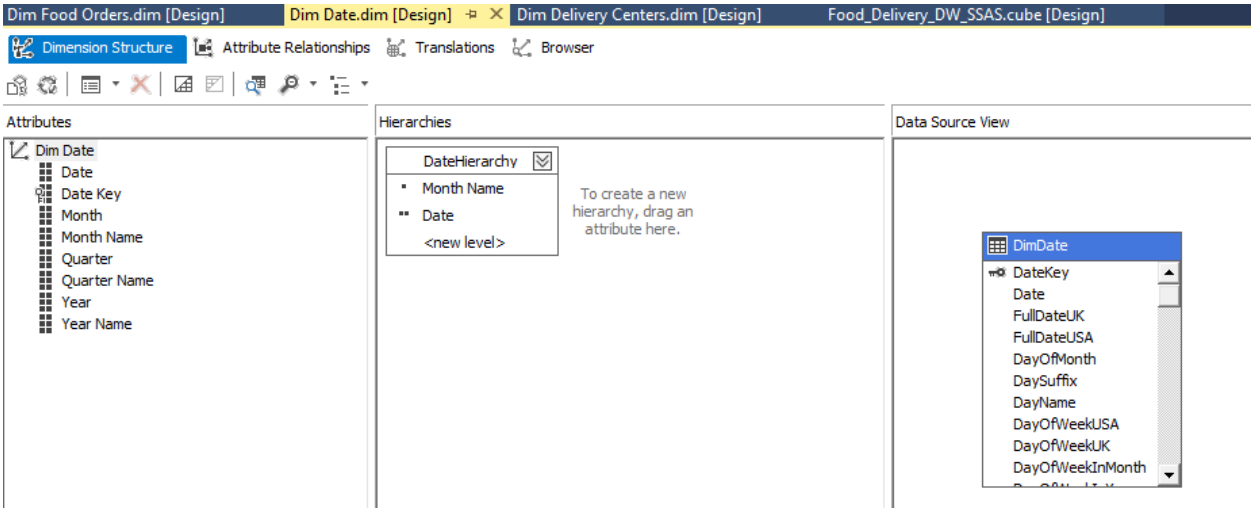
1). DimDeliveryCenters Dimension

- City Hierarchy
- Location Hierarchy



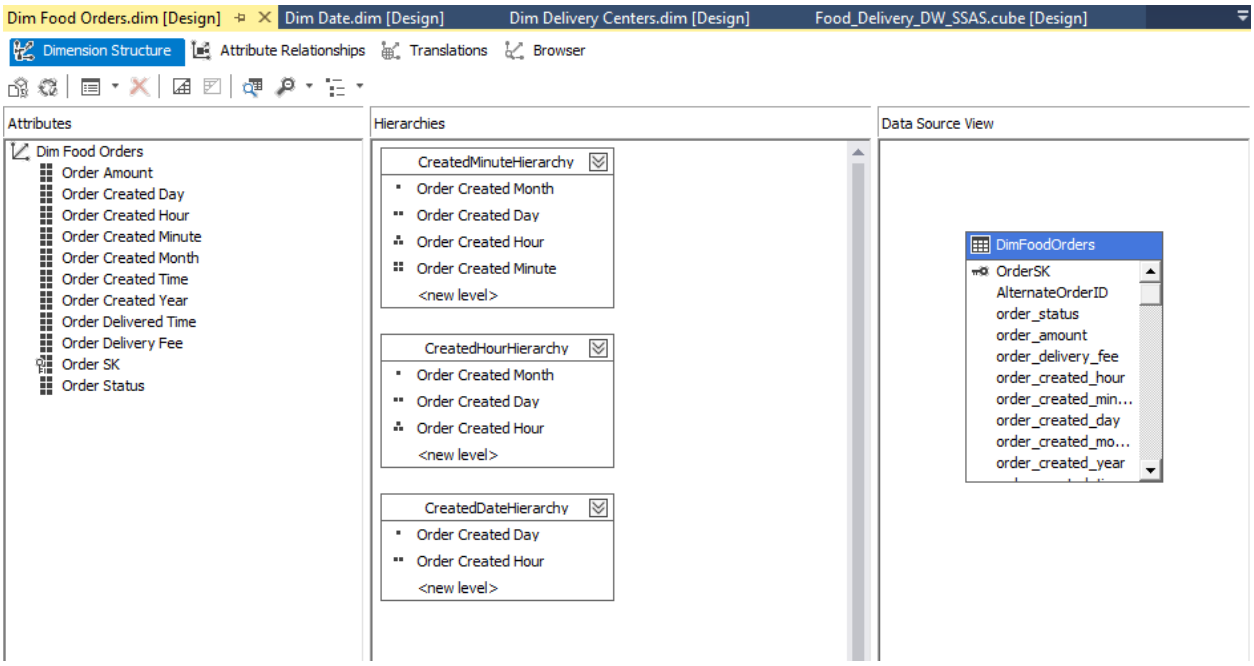
2). DimDate Dimension

- Date Hierarchy

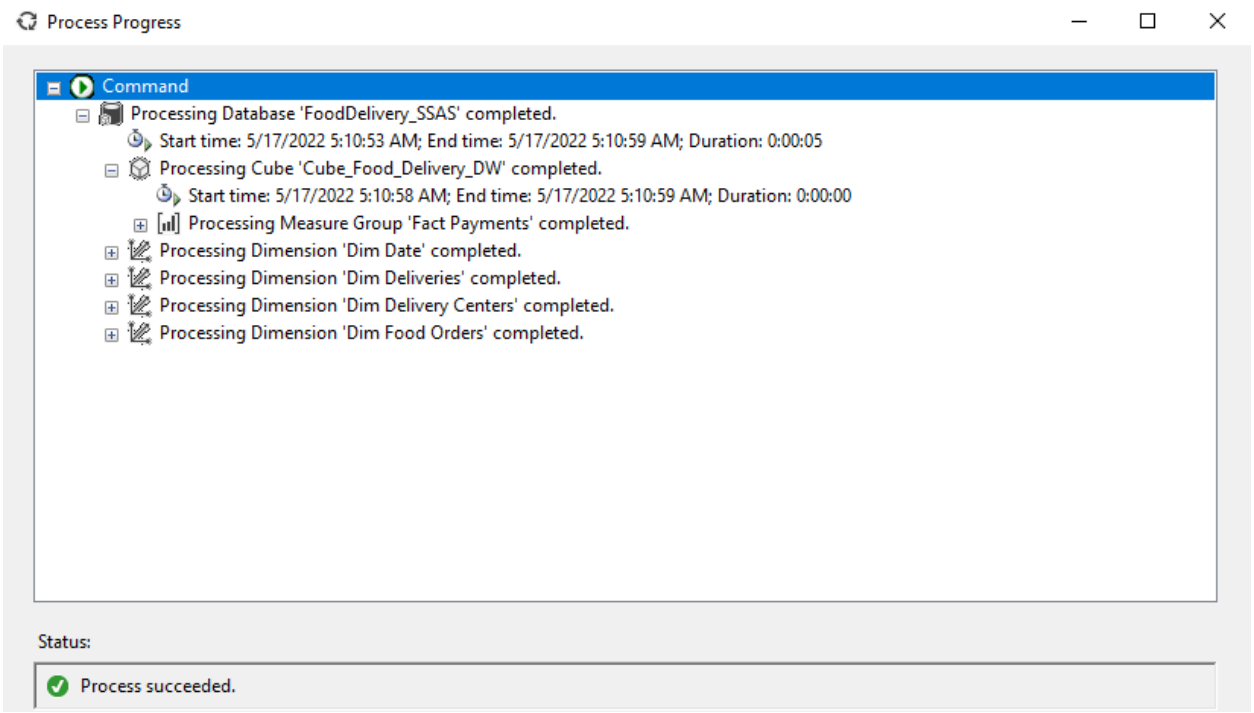


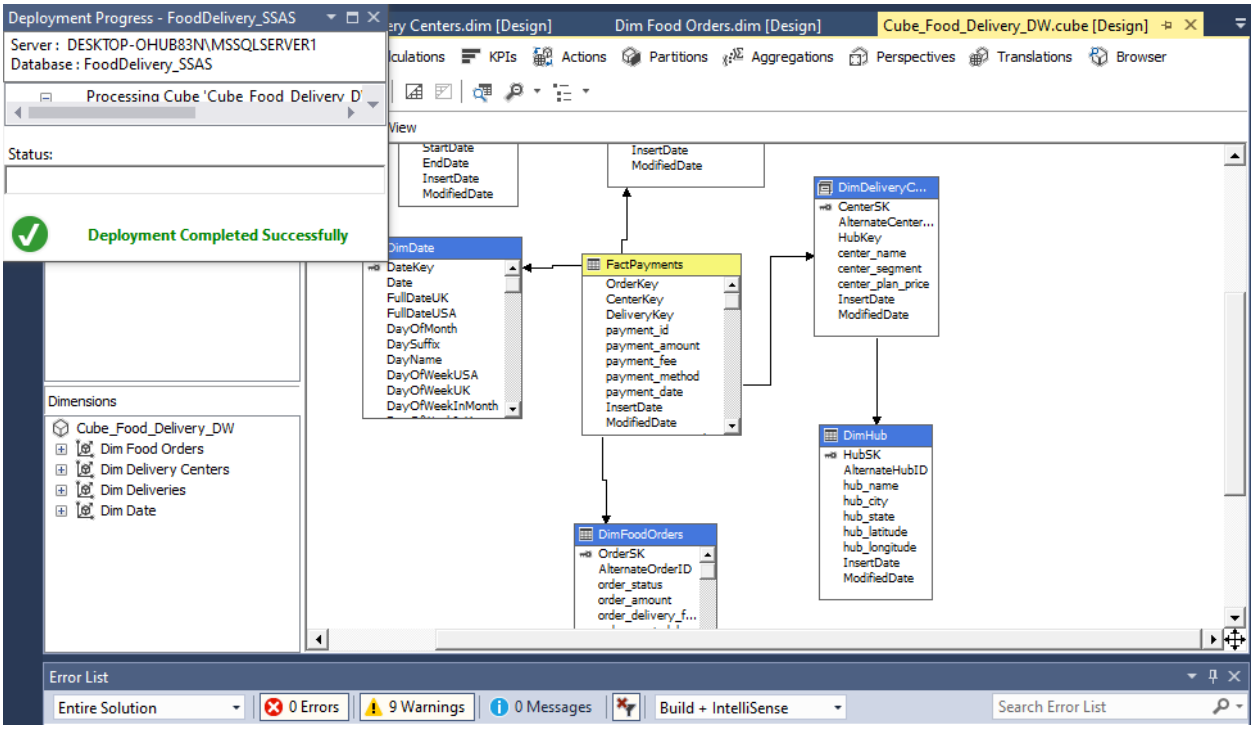
3). DimFoodOrders Dimension

- Created Minute Hierarchy
- Created Hour Hierarchy
- Created Date Hierarchy



Finally, the cube was deployed.





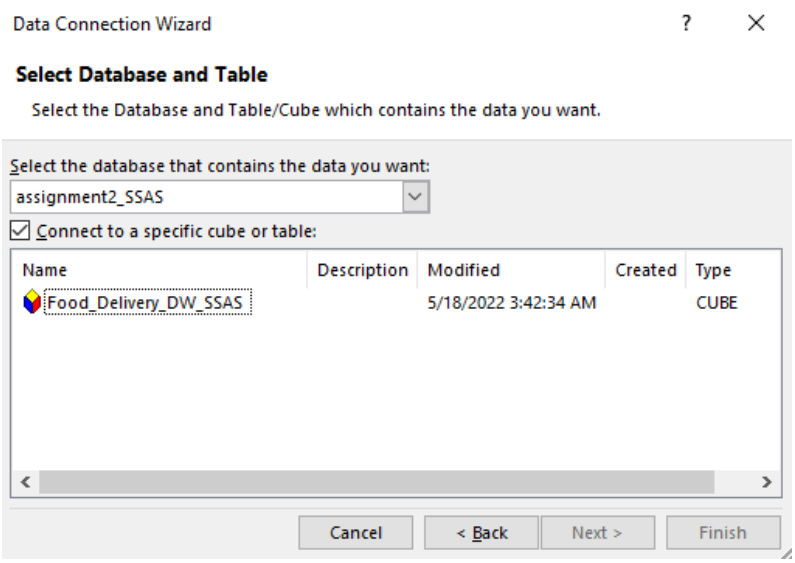
3. Demonstration of OLAP operations

To do the OLAP Operation demonstrations, Excel Data tab was used to directly connect to the cube to get all the fact and dimensional tables. When connecting to the deployed OLAP cube for the first time, it was done as follows.

1. Firstly, I Opened a new Excel sheet and go the Data tab and Select **From Other Sources ⟹ From Analysis Service**.

2. In the Data Connection Wizard, I provided the Server Name and selected Windows Authentication, and clicked Next.

3. Then I selected the relevant SSAS database, and the cube named “Food_Delivery_DW_SSAS” and then click Next.

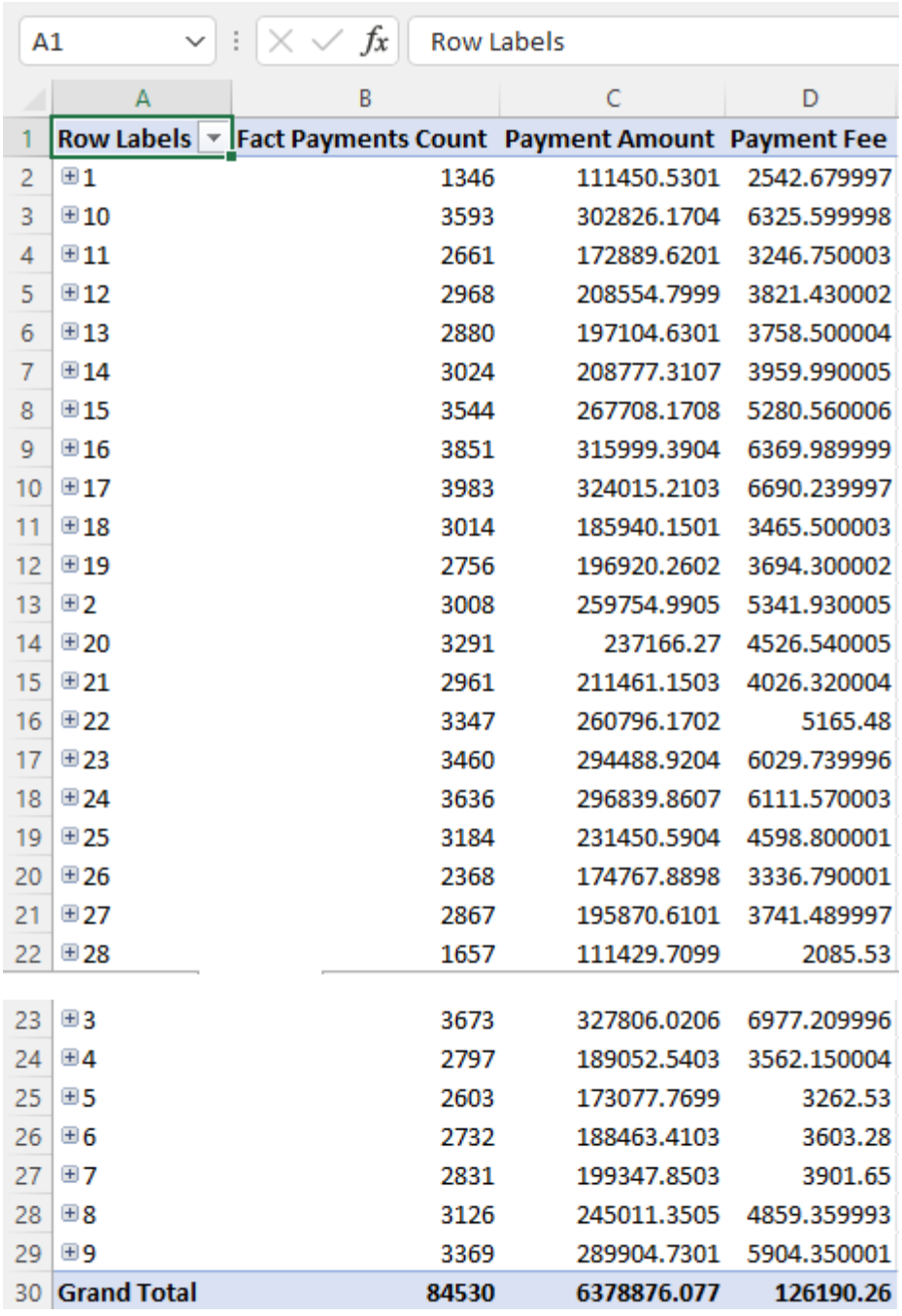


4. Left the default values in the next pop-up window and clicked OK.

After successfully connecting to the cube, Roll-up, Drill-down, Slice, Dice, and Pivot OLAP operations were demonstrated using different excel pivot tables using several worksheets of the same excel workbook.

(01). Roll-Up:

Roll-up operation was demonstrated in the "Roll-up & Drill-down" worksheet with the help of created hierarchies and other columns in the cube. As shown in the following figure, the number of payments made (Fact Payment Count), Total payments earned (Payment amount) and discounts given (Payment Fee) within a specified time (day/ hour) is presented in the pivot table. The time axis/column is in a roll-up view where the table shows payments made (Fact Payment Count), Total payments earned (Payment amount) and discounts given (Payment Fee) based on specific days only. This can be further drilled down to hours where it is possible to obtain payments made (Fact Payment Count), Total payments earned (Payment amount) and discounts given (Payment Fee) within a given hours in a day since the hierarchy used here allows it.



A1 | Row Labels

	A	B	C	D
1	Row Labels	Fact Payments Count	Payment Amount	Payment Fee
2	1	1346	111450.5301	2542.679997
3	10	3593	302826.1704	6325.599998
4	11	2661	172889.6201	3246.750003
5	12	2968	208554.7999	3821.430002
6	13	2880	197104.6301	3758.500004
7	14	3024	208777.3107	3959.990005
8	15	3544	267708.1708	5280.560006
9	16	3851	315999.3904	6369.989999
10	17	3983	324015.2103	6690.239997
11	18	3014	185940.1501	3465.500003
12	19	2756	196920.2602	3694.300002
13	2	3008	259754.9905	5341.930005
14	20	3291	237166.27	4526.540005
15	21	2961	211461.1503	4026.320004
16	22	3347	260796.1702	5165.48
17	23	3460	294488.9204	6029.739996
18	24	3636	296839.8607	6111.570003
19	25	3184	231450.5904	4598.800001
20	26	2368	174767.8898	3336.790001
21	27	2867	195870.6101	3741.489997
22	28	1657	111429.7099	2085.53
23	3	3673	327806.0206	6977.209996
24	4	2797	189052.5403	3562.150004
25	5	2603	173077.7699	3262.53
26	6	2732	188463.4103	3603.28
27	7	2831	199347.8503	3901.65
28	8	3126	245011.3505	4859.359993
29	9	3369	289904.7301	5904.350001
30	**Grand Total**	**84530**	**6378876.077**	**126190.26**

(02). Drill-Down:

Drill-down operation was also demonstrated using the same dimension attributes and measures using the same pivot table which is shown below in the same worksheet where roll-up was demonstrated. The rows of the pivot table are now drilled down further till hours where it allows to read the number of payments made (Fact Payment Count), Total payments earned (Payment amount) and discounts given (Payment Fee) based on hours of a given day. As it is shown here, low-level hierarchy attributes were used to demonstrate data in a drilled-down manner.

A92 | 9

	A	B	C	D
1	Row Labels	Fact Payments Count	Payment Amount	Payment Fee
2	**1**			
3	14	27	1929.34	45.90000024
4	15	183	13144.69997	296.7299998
5	16	148	13516.21003	324.4599998
6	17	167	14220.55	337.7599988
7	18	112	9806.220053	230.7300013
8	19	101	8933.060028	189.1599996
9	2	1	394.8099976	7.900000095
10	20	77	5515.549992	112.9299991
11	21	142	10553.49002	230.5999988
12	22	168	15531.85003	356.2999993
13	23	220	17904.75003	410.2100004
14	**10**			
15	0	335	28121.49006	586.4199976
16	1	40	2113.789989	39.12000018
17	12	2	211.7999954	3.530000091
18	13	17	1025.230003	18.73000011
19	14	212	21374.89999	487.3400002
20	15	476	46760.01	1032.36
21	16	433	41656.85008	889.9999993
22	17	284	25606.37007	531.7599994

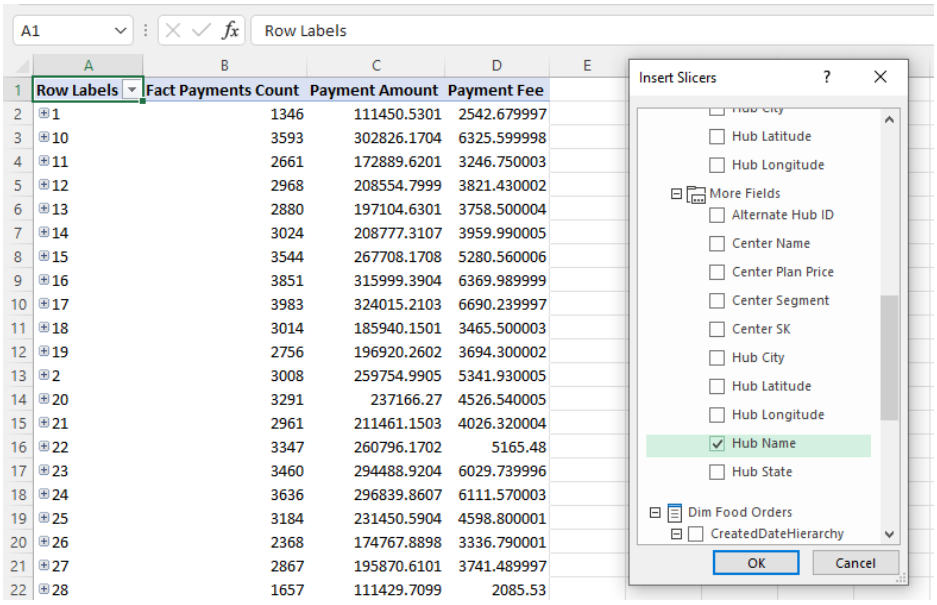
...

105	21	247	20324.81006	400.9900012
106	22	485	38233.33002	774.3500028
107	23	497	39298.26003	778.0499993
108	3	6	294.5	4.420000076
109	**Grand Total**	**84530**	**6378876.077**	**126190.26**

(03). Slice:

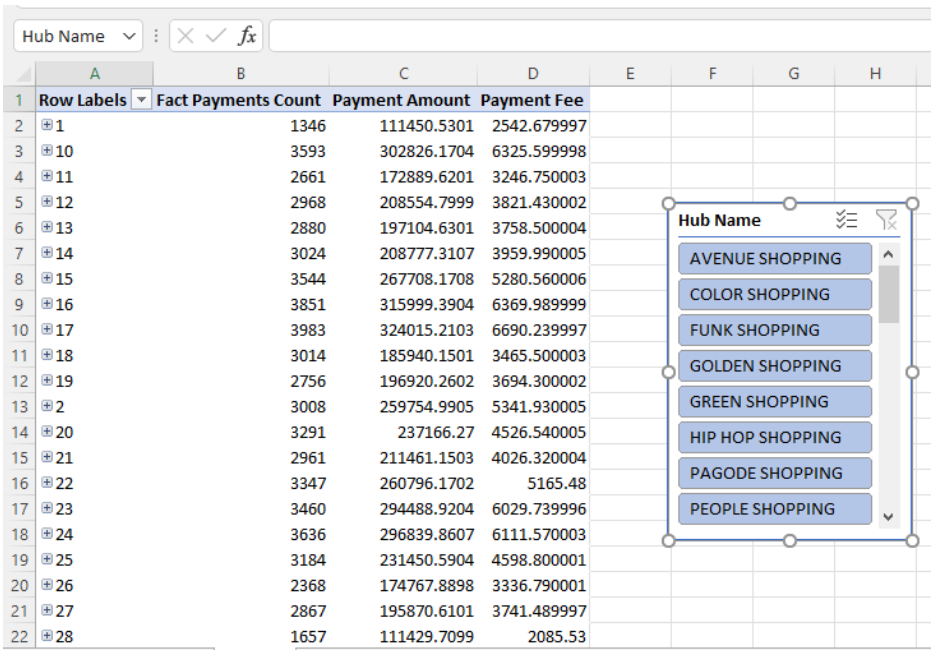
Below pivot table which was created in the “Slice excel sheet” helps to identify the number of payments made (Fact Payment Count), Total payments earned (Payment amount) and discounts given (Payment Fee) based on hour for each hub (Hub Name) separately, which has been used to demonstrate the slice operation. Here the data can be sliced based on “hub names” using the given slicer to see data related only for the selected “hub name/names”.

Slicer was implemented by clicking on the pivot table and then heading to “PivotTable Analyze” tab in the Excel Ribbon and then selecting the “Insert Slicer”. Then from the check list, the dimension attribute named “Hub Name” in “DimHub” was selected to perform the slicing.



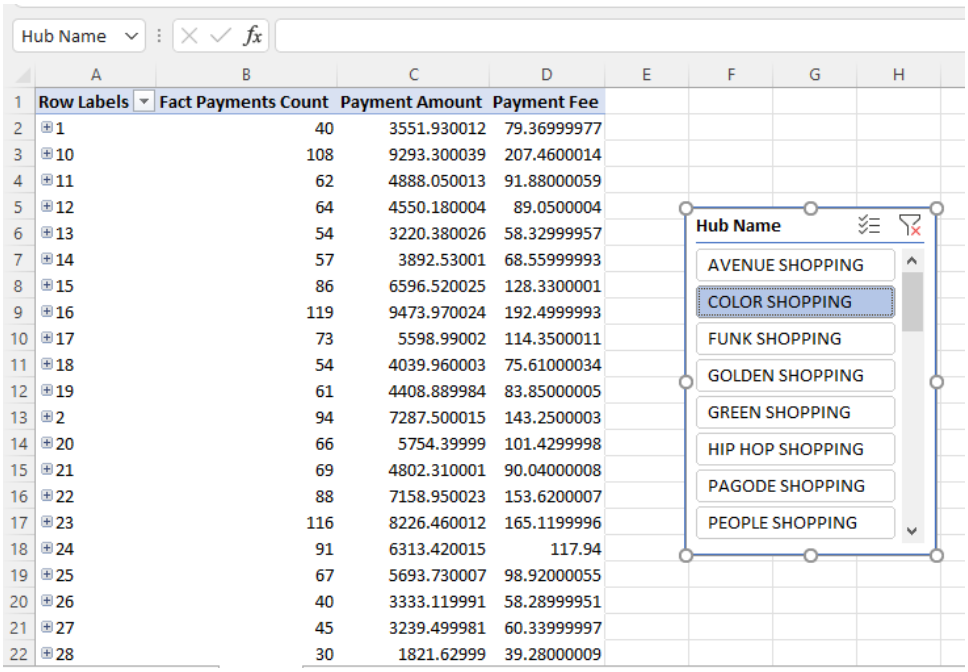
Row Labels	Fact Payments Count	Payment Amount	Payment Fee
1	1346	111450.5301	2542.679997
10	3593	302826.1704	6325.599998
11	2661	172889.6201	3246.750003
12	2968	208554.7999	3821.430002
13	2880	197104.6301	3758.500004
14	3024	208777.3107	3959.990005
15	3544	267708.1708	5280.560006
16	3851	315999.3904	6369.989999
17	3983	324015.2103	6690.239997
18	3014	185940.1501	3465.500003
19	2756	196920.2602	3694.300002
2	3008	259754.9905	5341.930005
20	3291	237166.27	4526.540005
21	2961	211461.1503	4026.320004
22	3347	260796.1702	5165.48
23	3460	294488.9204	6029.739996
24	3636	296839.8607	6111.570003
25	3184	231450.5904	4598.800001
26	2368	174767.8898	3336.790001
27	2867	195870.6101	3741.489997
28	1657	111429.7099	2085.53

Insert slicer from the toolbar was selected to insert hub name



Row Labels	Fact Payments Count	Payment Amount	Payment Fee
1	1346	111450.5301	2542.679997
10	3593	302826.1704	6325.599998
11	2661	172889.6201	3246.750003
12	2968	208554.7999	3821.430002
13	2880	197104.6301	3758.500004
14	3024	208777.3107	3959.990005
15	3544	267708.1708	5280.560006
16	3851	315999.3904	6369.989999
17	3983	324015.2103	6690.239997
18	3014	185940.1501	3465.500003
19	2756	196920.2602	3694.300002
2	3008	259754.9905	5341.930005
20	3291	237166.27	4526.540005
21	2961	211461.1503	4026.320004
22	3347	260796.1702	5165.48
23	3460	294488.9204	6029.739996
24	3636	296839.8607	6111.570003
25	3184	231450.5904	4598.800001
26	2368	174767.8898	3336.790001
27	2867	195870.6101	3741.489997
28	1657	111429.7099	2085.53

Before selecting a hub name/ names (before slicing was performed)

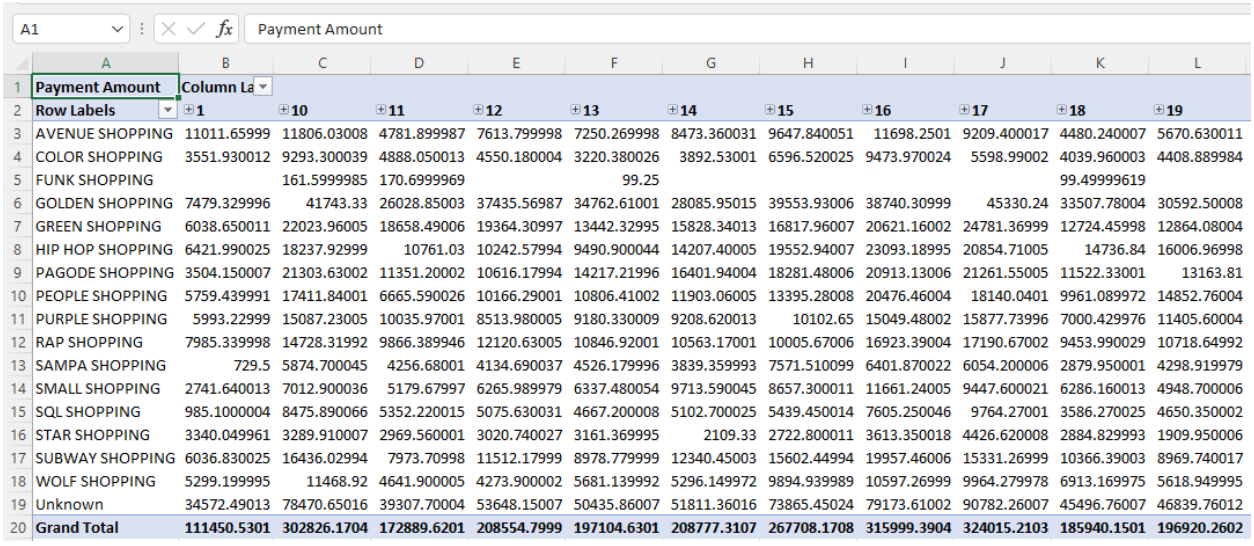


Row Labels	Fact Payments Count	Payment Amount	Payment Fee
1	40	3551.930012	79.36999977
10	108	9293.300039	207.4600014
11	62	4888.050013	91.88000059
12	64	4550.180004	89.0500004
13	54	3220.380026	58.32999957
14	57	3892.53001	68.55999993
15	86	6596.520025	128.3300001
16	119	9473.970024	192.4999993
17	73	5598.99002	114.3500011
18	54	4039.960003	75.61000034
19	61	4408.889984	83.85000005
2	94	7287.500015	143.2500003
20	66	5754.39999	101.4299998
21	69	4802.310001	90.04000008
22	88	7158.950023	153.6200007
23	116	8226.460012	165.1199996
24	91	6313.420015	117.94
25	67	5693.730007	98.92000055
26	40	3333.119991	58.28999951
27	45	3239.499981	60.33999997
28	30	1821.62999	39.28000009

After doing the slicing

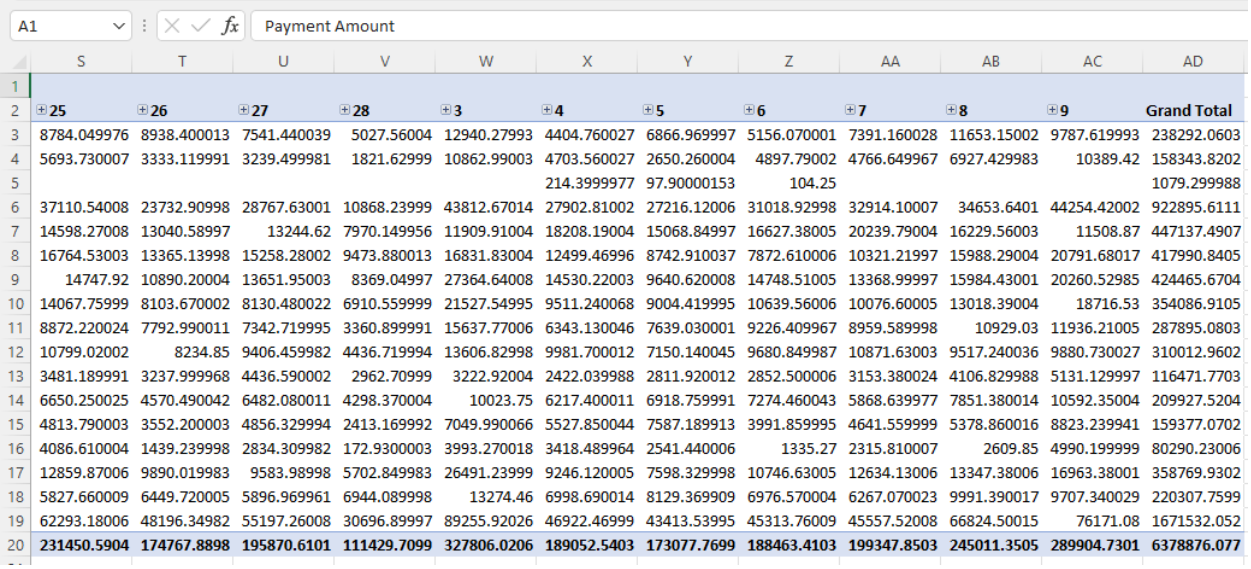
(04). Dice:

Dice is performed by slicing two dimensions in the pivot table. The Payment Amount, Order Created Day and Hub Name was used to implement dice operation in a new worksheet named "Dice" in the excel workbook. The pivot table attributes were diced by "Hub Name" from "DimHubs" and "Order Created Day" from "DimFoodOrders" to demonstrate the Dice OLAP operation as follows.



Payment Amount	Column La											
Row Labels	1	10	11	12	13	14	15	16	17	18	19
AVENUE SHOPPING	11011.65999	11806.03008	4781.899987	7613.799998	7250.269998	8473.360031	9647.840051	11698.2501	9209.400017	4480.240007	5670.630011
COLOR SHOPPING	3551.930012	9293.300039	4888.050013	4550.180004	3220.380026	3892.53001	6596.520025	9473.970024	5598.99002	4039.960003	4408.889984
FUNK SHOPPING		161.5999985	170.6999969		99.25					99.49999619	
GOLDEN SHOPPING	7479.329996	41743.33	26028.85003	37435.56987	34762.61001	28085.95015	39553.93006	38740.30999	45330.24	33507.78004	30592.50008
GREEN SHOPPING	6038.650011	22023.96005	18658.49006	19364.30997	13442.32995	15828.34013	16817.96007	20621.16002	24781.36999	12724.45998	12864.08004
HIP HOP SHOPPING	6421.990025	18237.92999	10761.03	10242.57994	9490.900044	14207.40005	19552.94007	23093.18995	20854.71005	14736.84	16006.96998
PAGODE SHOPPING	3504.150007	21303.63002	11351.20002	10616.17994	14217.21996	16401.94004	18281.48006	20913.13006	21261.55005	11522.33001	13163.81
PEOPLE SHOPPING	5759.439991	17411.84001	6665.590026	10166.29001	10806.41002	11903.06005	13395.28008	20476.46004	18140.0401	9961.089972	14852.76004
PURPLE SHOPPING	5993.22999	15087.23005	10035.97001	8513.980005	9180.330009	9208.620013	10102.65	15049.48002	15877.73996	7000.429976	11405.60004
RAP SHOPPING	7985.339998	14728.31992	9866.389946	12120.63005	10846.92001	10563.17001	10005.67006	16923.39004	17190.67002	9453.990029	10718.64992
SAMPA SHOPPING	729.5	5874.700045	4256.68001	4134.690037	4526.179996	3839.359993	7571.510099	6401.870022	6054.200006	2879.950001	4298.919979
SMALL SHOPPING	2741.640013	7012.900036	5179.67997	6265.989979	6337.480054	9713.590045	8657.300011	11661.24005	9447.600021	6286.160013	4948.700006
SQL SHOPPING	985.1000004	8475.890066	5352.220015	5075.630031	4667.200008	5102.700025	5439.450014	7605.250046	9764.27001	3586.270025	4650.350002
STAR SHOPPING	3340.049961	3289.910007	2969.560001	3020.740027	3161.369995	2109.33	2722.800011	3613.350018	4426.620008	2884.829993	1909.950006
SUBWAY SHOPPING	6036.830025	16436.02994	7973.70998	11512.17999	8978.779999	12340.45003	15602.44994	19957.46006	15331.26999	10366.39003	8969.740017
WOLF SHOPPING	5299.199995	11468.92	4641.900005	4273.900002	5681.139992	5296.149972	9894.939989	10597.26999	9964.279978	6913.169975	5618.949995
Unknown	34572.49013	78470.65016	39307.70004	53648.15007	50435.86007	51811.36016	73865.45024	79173.61002	90782.26007	45496.76007	46839.76012
Grand Total	**111450.5301**	**302826.1704**	**172889.6201**	**208554.7999**	**197104.6301**	**208777.3107**	**267708.1708**	**315999.3904**	**324015.2103**	**185940.1501**	**196920.2602**

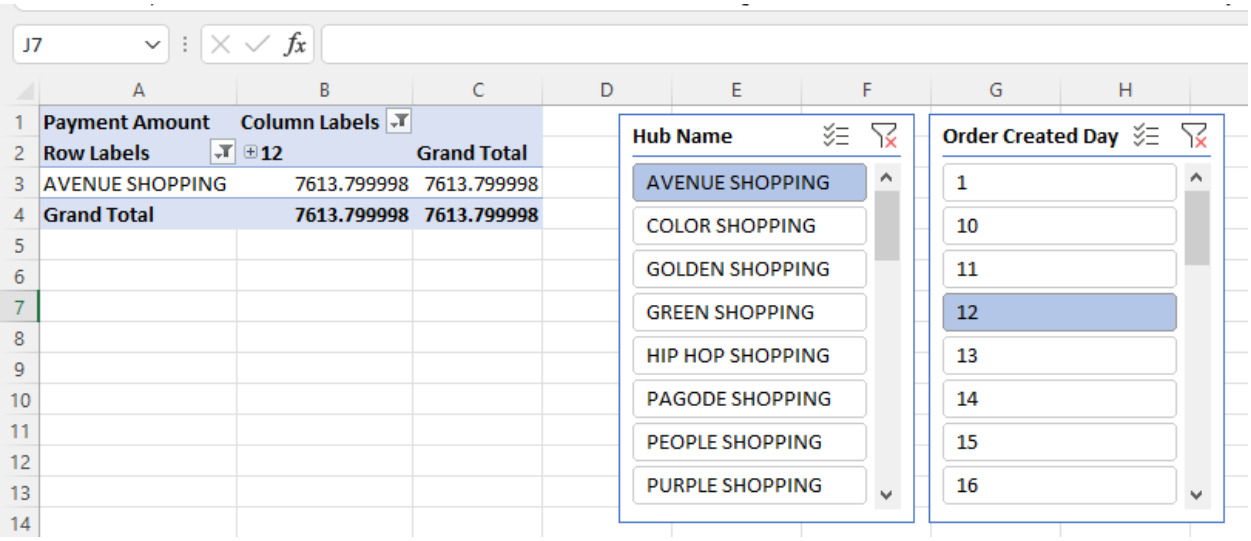
(Before the Dice operation was performed – table first half/ left side)



A1 | Payment Amount

	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
1												
2	25	26	27	28	3	4	5	6	7	8	9	Grand Total
3	8784.049976	8938.400013	7541.440039	5027.56004	12940.27993	4404.760027	6866.969997	5156.070001	7391.160028	11653.15002	9787.619993	238292.0603
4	5693.730007	3333.119991	3239.499981	1821.62999	10862.99003	4703.560027	2650.260004	4897.79002	4766.649967	6927.429983	10389.42	158343.8202
5						214.3999977	97.90000153	104.25				1079.299988
6	37110.54008	23732.90998	28767.63001	10868.23999	43812.67014	27902.81002	27216.12006	31018.92998	32914.10007	34653.6401	44254.42002	922895.6111
7	14598.27008	13040.58997	13244.62	7970.149956	11909.91004	18208.19004	15068.84997	16627.38005	20239.79004	16229.56003	11508.87	447137.4907
8	16764.53003	13365.13998	15258.28002	9473.880013	16831.83004	12499.46996	8742.910037	7872.610006	10321.21997	15988.29004	20791.68017	417990.8405
9	14747.92	10890.20004	13651.95003	8369.04997	27364.64008	14530.22003	9640.620008	14748.51005	13368.99997	15984.43001	20260.52985	424465.6704
10	14067.75999	8103.670002	8130.480022	6910.559999	21527.54995	9511.240068	9004.419995	10639.56006	10076.60005	13018.39004	18716.53	354086.9105
11	8872.220024	7792.990011	7342.719995	3360.899991	15637.77006	6343.130046	7639.030001	9226.409967	8959.589998	10929.03	11936.21005	287895.0803
12	10799.02002	8234.85	9406.459982	4436.719994	13606.82998	9981.700012	7150.140045	9680.849987	10871.63003	9517.240036	9880.730027	310012.9602
13	3481.189991	3237.999968	4436.590002	2962.70999	3222.92004	2422.039988	2811.920012	2852.500006	3153.380024	4106.829988	5131.129997	116471.7703
14	6650.250025	4570.490042	6482.080011	4298.370004	10023.75	6217.400011	6918.759991	7274.460043	5868.639977	7851.380014	10592.35004	209927.5204
15	4813.790003	3552.200003	4856.329994	2413.169992	7049.990066	5527.850044	7587.189913	3991.859995	4641.559999	5378.860016	8823.239941	159377.0702
16	4086.610004	1439.239998	2834.309982	172.9300003	3993.270018	3418.489964	2541.440006	1335.27	2315.810007	2609.85	4990.199999	80290.23006
17	12859.87006	9890.019983	9583.98998	5702.849983	26491.23999	9246.120005	7598.329998	10746.63005	12634.13006	13347.38006	16963.38001	358769.9302
18	5827.660009	6449.720005	5896.969961	6944.089998	13274.46	6998.690014	8129.369909	6976.570004	6267.070023	9991.390017	9707.340029	220307.7599
19	62293.18006	48196.34982	55197.26008	30696.89997	89255.92026	46922.46999	43413.53995	45313.76009	45557.52008	66824.50015	76171.08	1671532.052
20	**231450.5904**	**174767.8898**	**195870.6101**	**111429.7099**	**327806.0206**	**189052.5403**	**173077.7699**	**188463.4103**	**199347.8503**	**245011.3505**	**289904.7301**	**6378876.077**

(Before the Dice operation was performed – table second half/ right side)



J7

	A	B	C
1	Payment Amount	Column Labels	
2	Row Labels	12	Grand Total
3	AVENUE SHOPPING	7613.799998	7613.799998
4	**Grand Total**	**7613.799998**	**7613.799998**

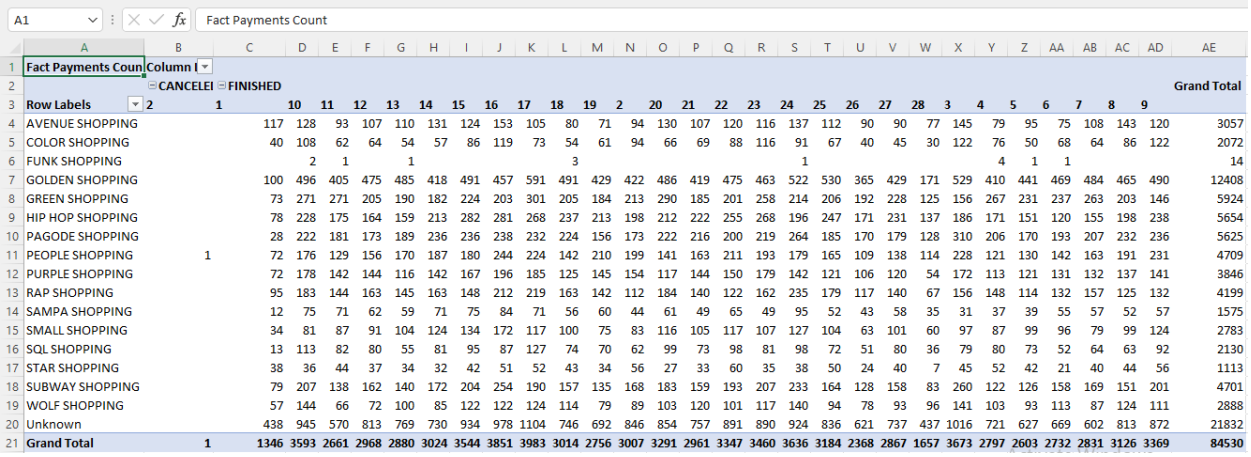
Hub Name: AVENUE SHOPPING, COLOR SHOPPING, GOLDEN SHOPPING, GREEN SHOPPING, HIP HOP SHOPPING, PAGODE SHOPPING, PEOPLE SHOPPING, PURPLE SHOPPING

Order Created Day: 1, 10, 11, 12, 13, 14, 15, 16

(After the Dice Operation – A hub name and a date was selected)

(05). Pivot:

Pivot was performed on a pivot table which was designed to analyze the total number of payments happened in a hub when the order status is either canceled or finished in a day. In here the pivot operation was demonstrated by inter-changing the order created day attribute from being a column to being a row in the pivot table. It can be observed how the measure values are changing when the pivot is done and how the grand total of all the total number of payments, remains unchanged.



Fact Payments Coun	Column L																													
	CANCELEI	FINISHED																												Grand Total
Row Labels	2	1	10	11	12	13	14	15	16	17	18	19	2	20	21	22	23	24	25	26	27	28	3	4	5	6	7	8	9	
AVENUE SHOPPING		117	128	93	107	110	131	124	153	105	80	71	94	130	107	120	116	137	112	90	90	77	145	79	95	75	108	143	120	3057
COLOR SHOPPING		40	108	62	64	54	57	86	119	73	54	61	94	66	69	88	116	91	67	40	45	30	122	76	50	68	64	86	122	2072
FUNK SHOPPING			2	1		1					3							1						4	1	1				14
GOLDEN SHOPPING		100	496	405	475	485	418	491	457	591	491	429	422	486	419	475	463	522	530	365	429	171	529	410	441	469	484	465	490	12408
GREEN SHOPPING		73	271	271	205	190	182	224	203	301	205	184	213	290	185	201	258	214	206	192	228	125	156	267	231	237	263	203	146	5924
HIP HOP SHOPPING		78	228	175	164	159	213	282	281	268	237	213	198	212	222	255	268	196	247	171	231	137	186	171	151	120	155	198	238	5654
PAGODE SHOPPING		28	222	181	173	189	236	236	238	232	224	156	173	222	216	200	219	264	185	170	179	128	310	206	170	193	207	232	236	5625
PEOPLE SHOPPING	1	72	176	129	156	170	187	180	244	224	142	210	199	141	163	211	193	179	165	109	138	114	228	121	130	142	163	191	231	4709
PURPLE SHOPPING		72	178	142	144	116	142	167	196	185	125	145	154	117	144	150	179	142	121	106	120	54	172	113	121	131	132	137	141	3846
RAP SHOPPING		95	183	144	163	145	163	148	212	219	163	142	112	184	140	122	162	235	179	117	140	67	156	148	114	132	157	125	132	4199
SAMPA SHOPPING		12	75	71	62	59	71	75	84	71	56	60	44	61	49	65	49	95	52	43	58	35	31	37	39	55	57	52	57	1575
SMALL SHOPPING		34	81	87	91	104	124	134	172	117	100	75	83	116	105	117	107	127	104	63	101	60	97	87	99	96	79	99	124	2783
SQL SHOPPING		13	113	82	80	55	81	95	87	127	74	70	62	99	73	98	81	98	72	51	80	36	79	80	73	52	64	63	92	2130
STAR SHOPPING		38	36	44	37	34	32	42	51	52	43	34	56	27	33	60	35	38	50	24	40	7	45	52	42	21	40	44	56	1113
SUBWAY SHOPPING		79	207	138	162	140	172	204	254	190	157	135	168	183	159	193	207	233	164	128	158	83	260	122	126	158	169	151	201	4701
WOLF SHOPPING		57	144	66	72	100	85	122	122	124	114	79	89	103	120	101	117	140	94	78	93	96	141	103	93	113	87	124	111	2888
Unknown		438	945	570	813	769	730	934	978	1104	746	692	846	854	757	891	890	924	836	621	737	437	1016	721	627	669	602	813	872	21832
Grand Total	**1**	**1346**	**3593**	**2661**	**2968**	**2880**	**3024**	**3544**	**3851**	**3983**	**3014**	**2756**	**3007**	**3291**	**2961**	**3347**	**3460**	**3636**	**3184**	**2368**	**2867**	**1657**	**3673**	**2797**	**2603**	**2732**	**2831**	**3126**	**3369**	**84530**

(Order created day is a column)

Fact Payments Count	Column Labels		
Row Labels	CANCELED	FINISHED	Grand Total
AVENUE SHOPPING			
1		117	117
10		128	128
11		93	93
12		107	107
13		110	110
14		131	131
15		124	124
16		153	153
17		105	105

...

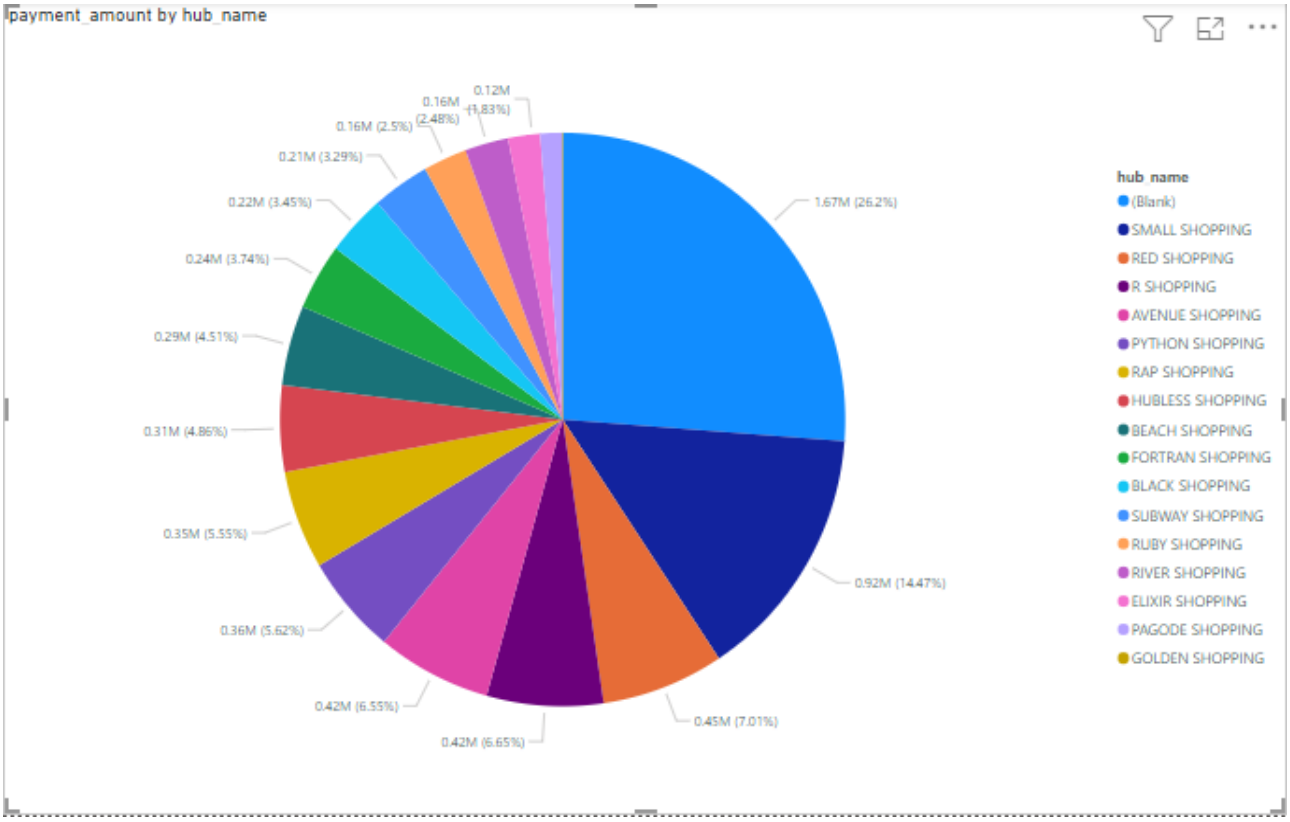
470	4		721	721
471	5		627	627
472	6		669	669
473	7		602	602
474	8		813	813
475	9		872	872
476	**Grand Total**	**1**	**84529**	**84530**

(Order created day is a row)

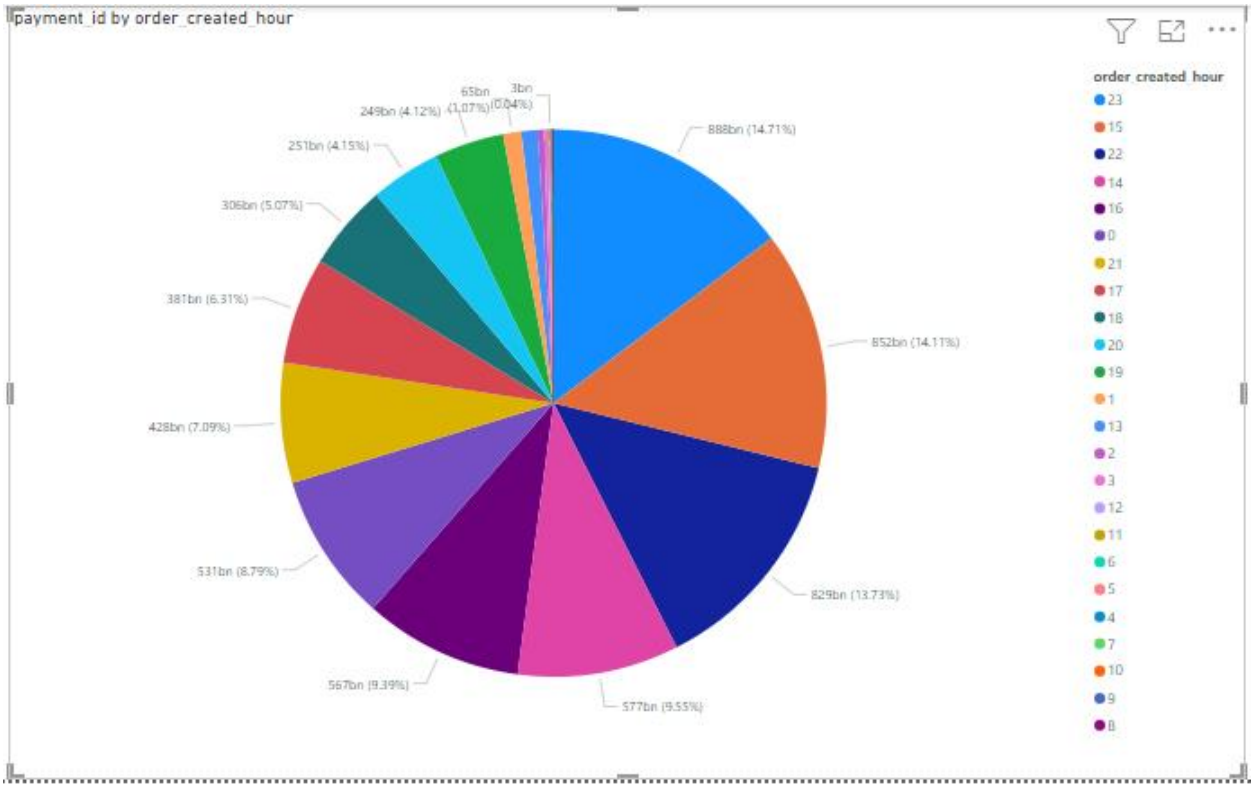
4. Power BI Visualization:

In addition to the above pivot table analysis, a dashboard like data visualization was implemented using power BI pivot charts and tables along with slices to filter-out the results on the dashboard.

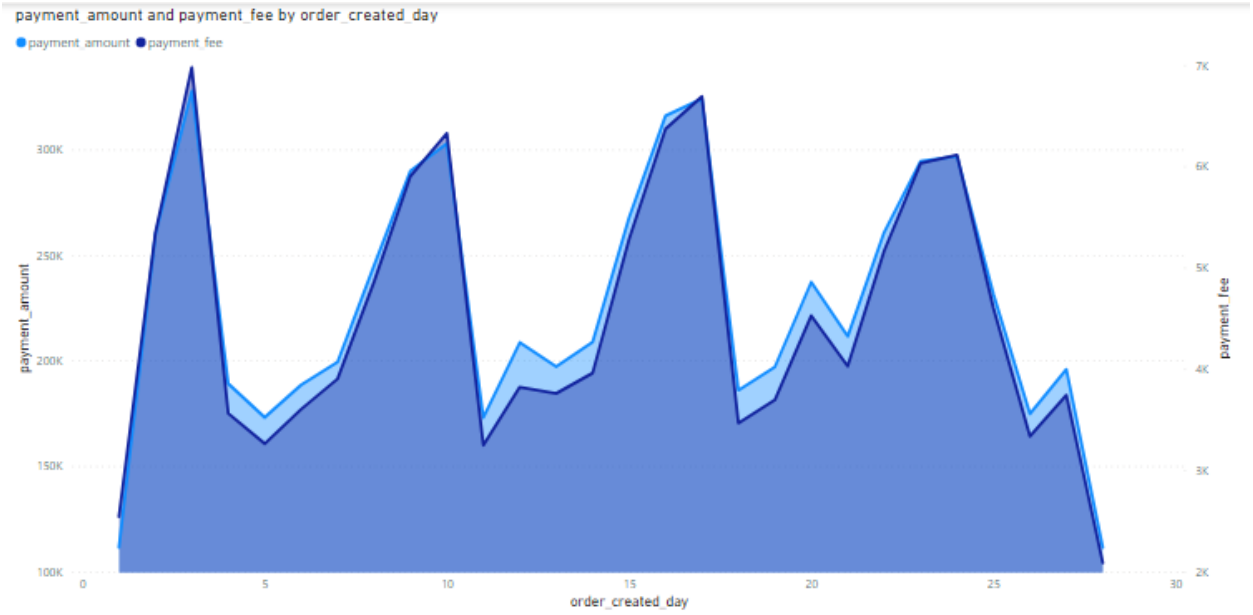
It was designed in a user-friendly manner and in a way that the analytics can be easily interpreted using visualizations rather than looking at complex pivot tables. Pivot charts such as Line, Bar charts were mainly used with slicers to filter the data faster than having to use drop down check lists.



(Profits in each hub – Slice operations)



(To find which days of the month has most food orders)

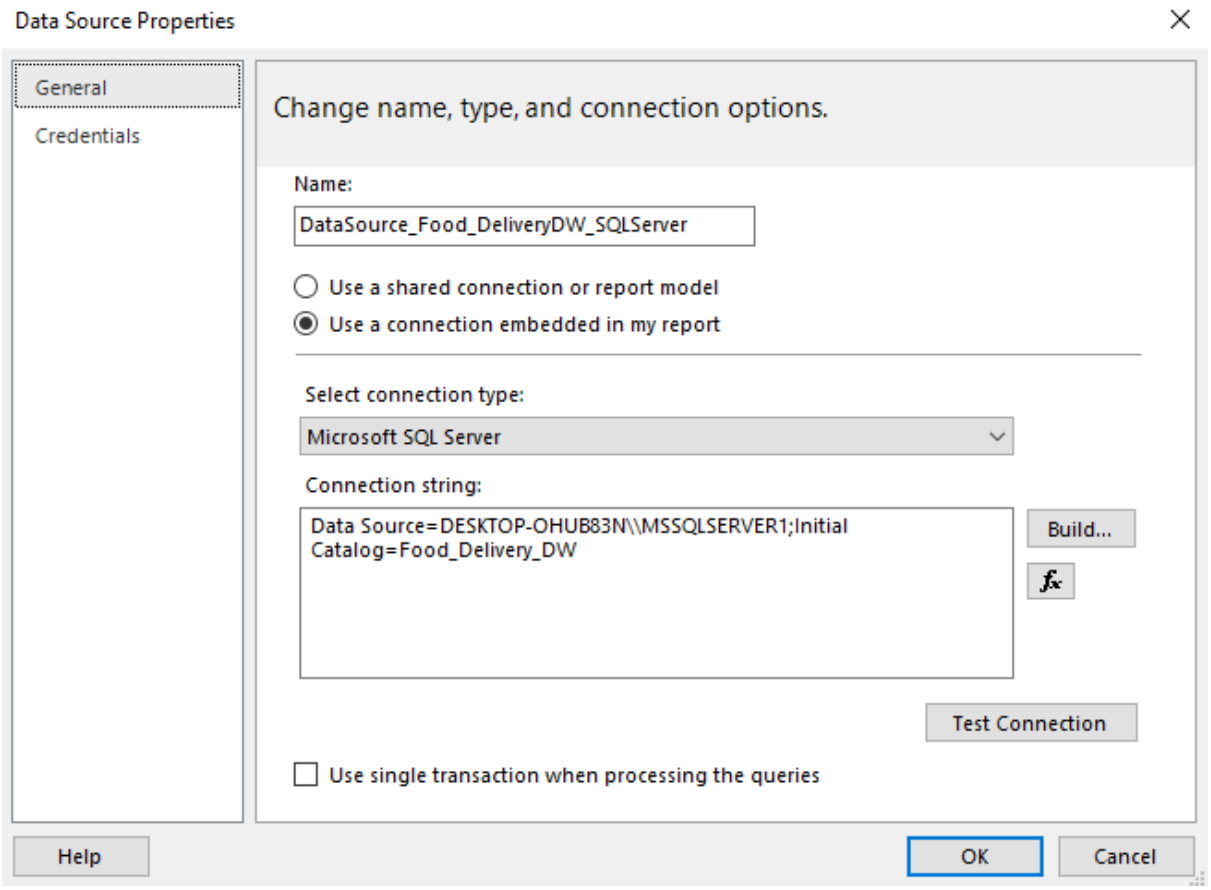


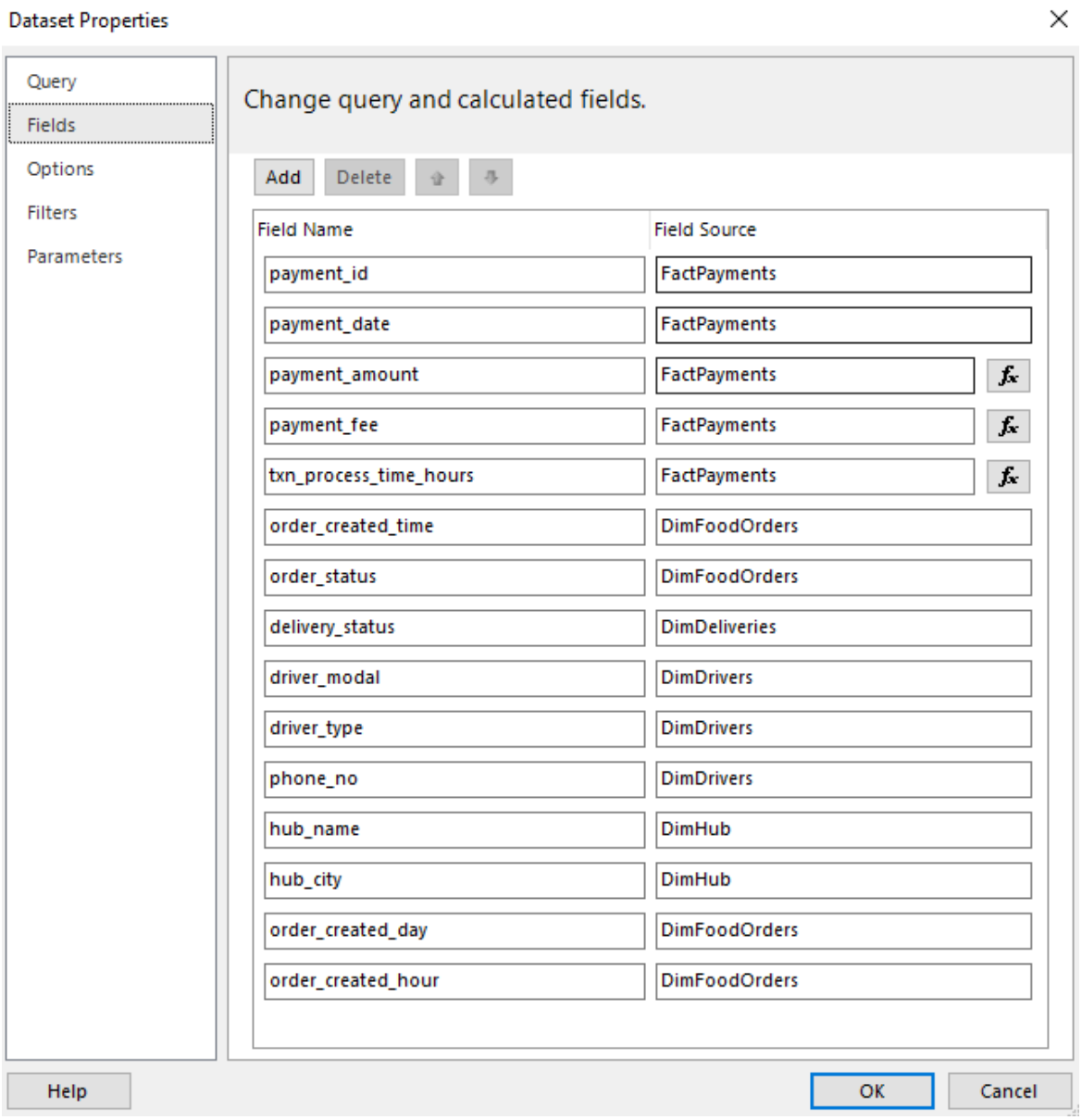
(Payment amount [earnings-light blue] vs Payment fee [discounts-dark blue] through the timeline)

5. SSRS Reports

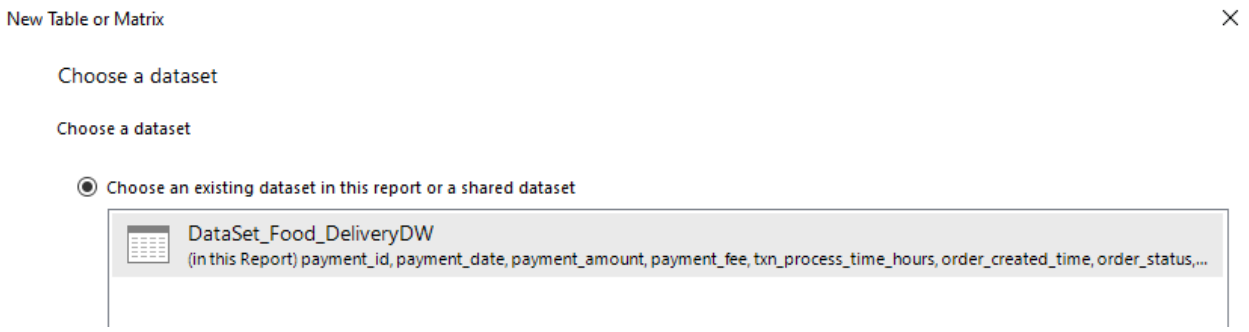
Report 1 – SSRS Report with a matrix

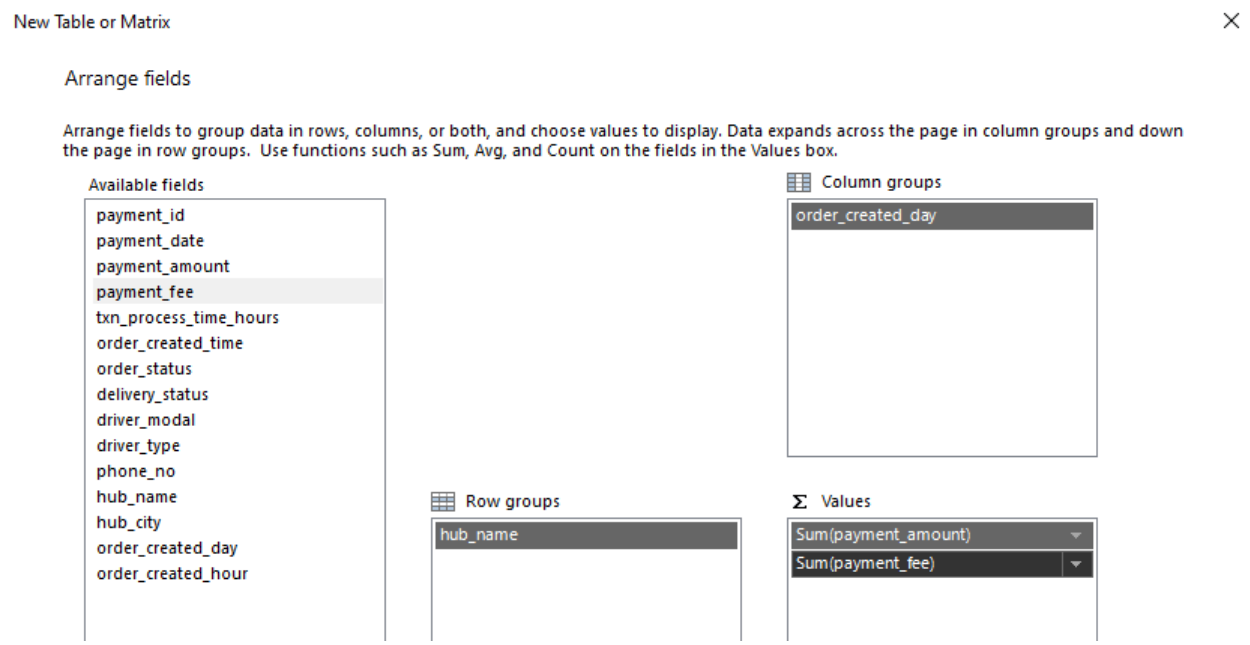
The first report was created as "Hubs Daily Transactions Report" which analyses the total payment amounts earned and total discounts given based on hubs in a daily manner. First, the Data warehouse was added as the Data Source to the Report Builder providing valid credentials. Then the necessary data were retrieved using the data source created, by adding the calculated and query fields in the fields section.

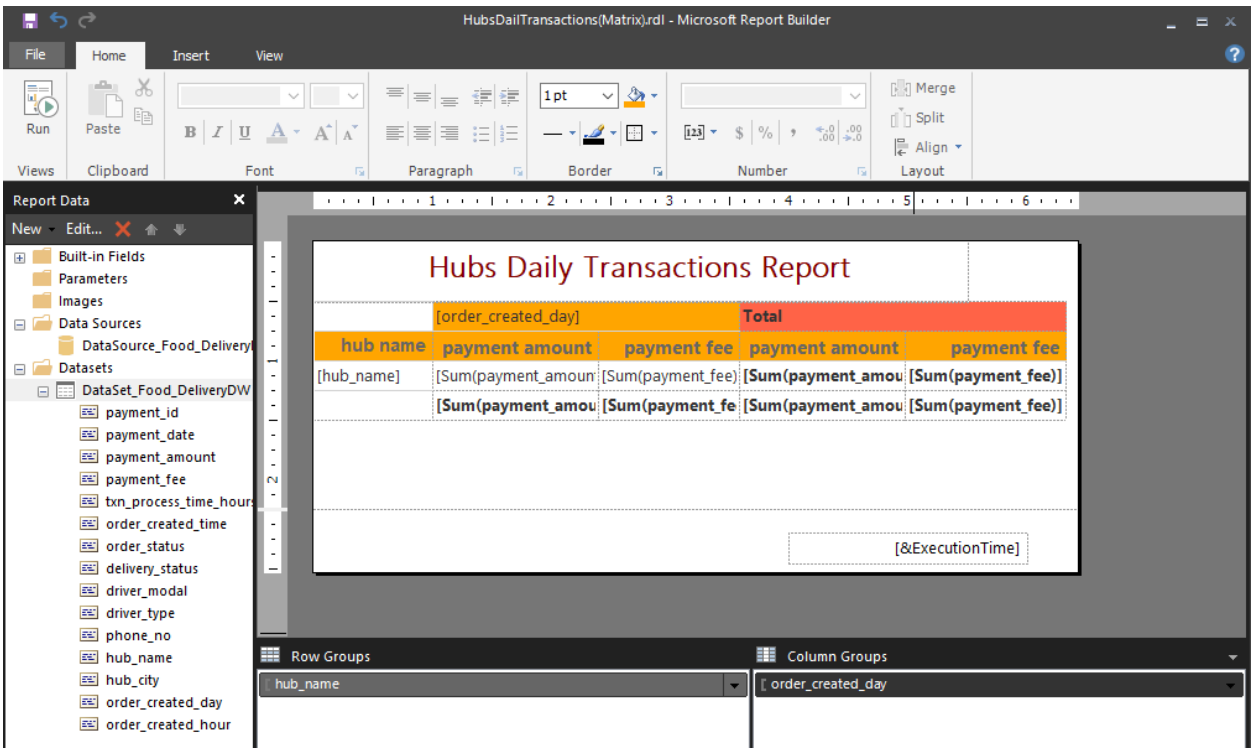




Then first the dataset was selected and next the following Fields were selected, and the layout settings were left as it is From the New Table or Matrix Window and the Matrix was created.

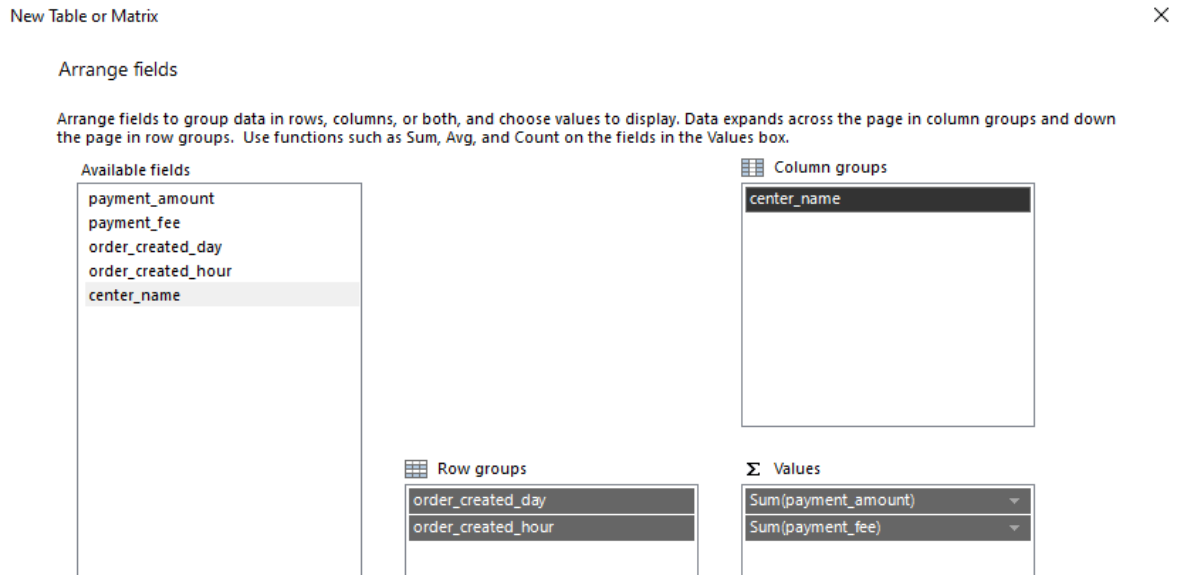


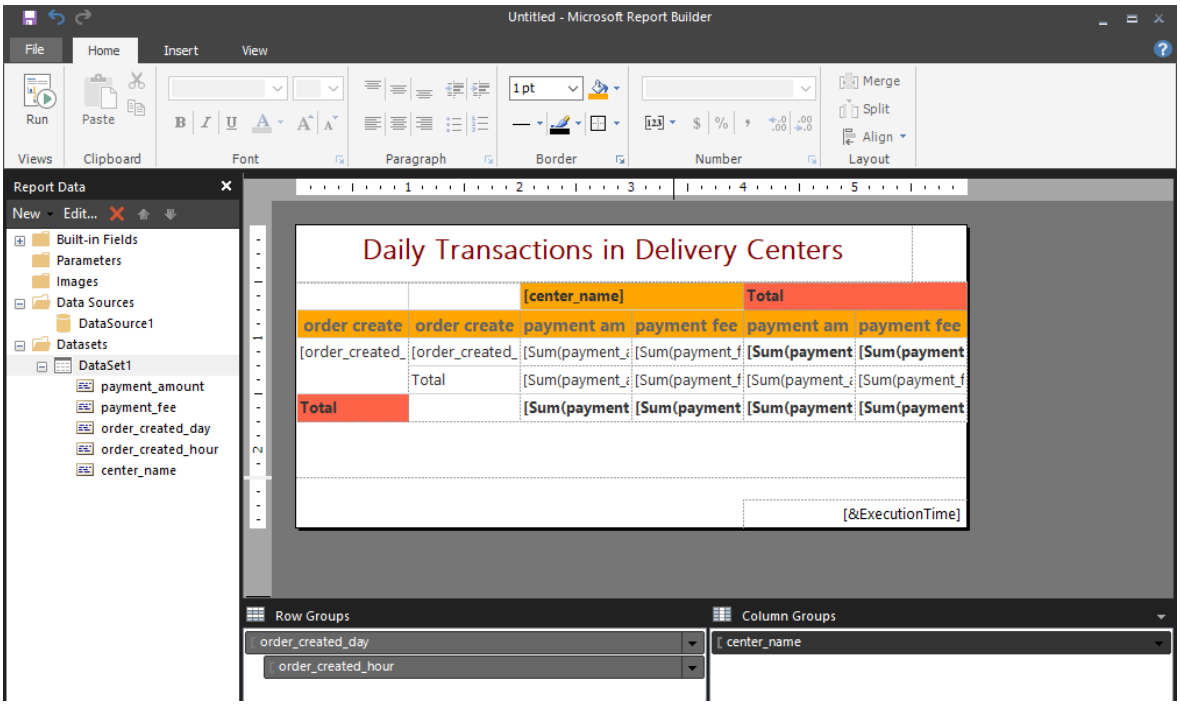

New Table or Matrix
Arrange fields
Arrange fields to group data in rows, columns, or both, and choose values to display. Data expands across the page in column groups and down the page in row groups. Use functions such as Sum, Avg, and Count on the fields in the Values box.
Available fields
payment_id
payment_date
payment_amount
payment_fee
txn_process_time_hours
order_created_time
order_status
delivery_status
driver_modal
driver_type
phone_no
hub_name
hub_city
order_created_day
order_created_hour
Column groups
order_created_day
Row groups
hub_name
Values
Sum(payment_amount)
Sum(payment_fee)


HubsDailTransactions(Matrix).rdl - Microsoft Report Builder
File
Home
Insert
View
Report Data
Built-in Fields
Parameters
Images
Data Sources
Datasets
DataSet_Food_DeliveryDW
payment_id
payment_date
payment_amount
payment_fee
txn_process_time_hours
order_created_time
order_status
delivery_status
driver_modal
driver_type
phone_no
hub_name
hub_city
order_created_day
order_created_hour
Hubs Daily Transactions Report
[order_created_day]
Total
hub name
payment amount
payment fee
payment amount
payment fee
[hub_name]
[&ExecutionTime]
Row Groups
Column Groups
hub_name
order_created_day

Report 2 – SSRS Drill-Down Report

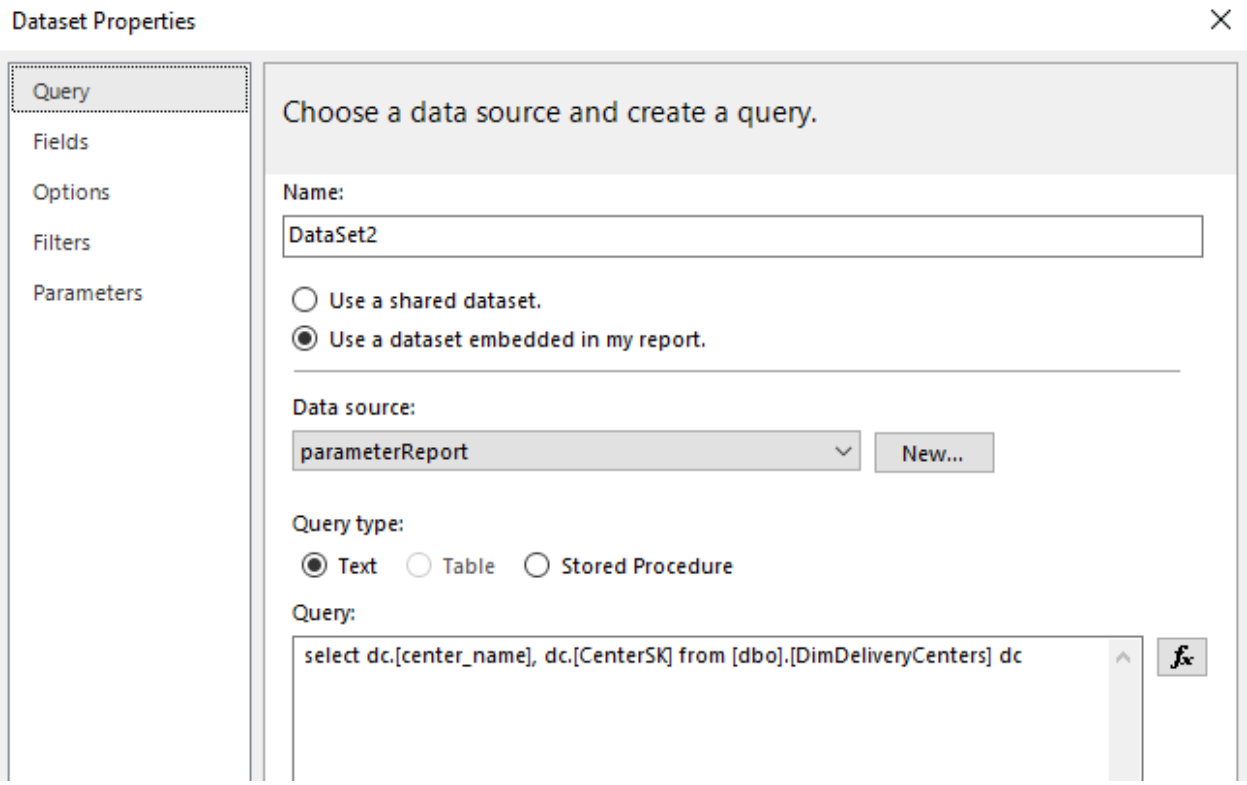
A drill-down report a report which contains row groups or column groups which are expandable as same as in excel drill-down OLAP operation. Here, a report was built which gives Daily incomes and discounts done in delivery centers. The drill-down feature is enabled by allowing the day to be further expandable to view hours transactions using the expand/collapse buttons in the report.

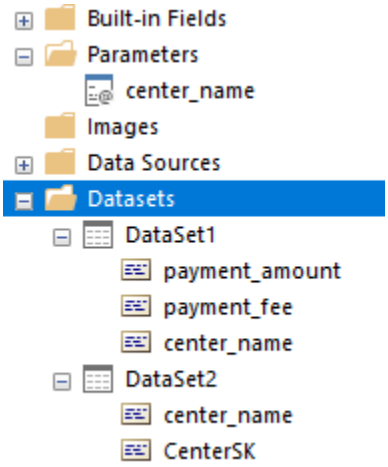


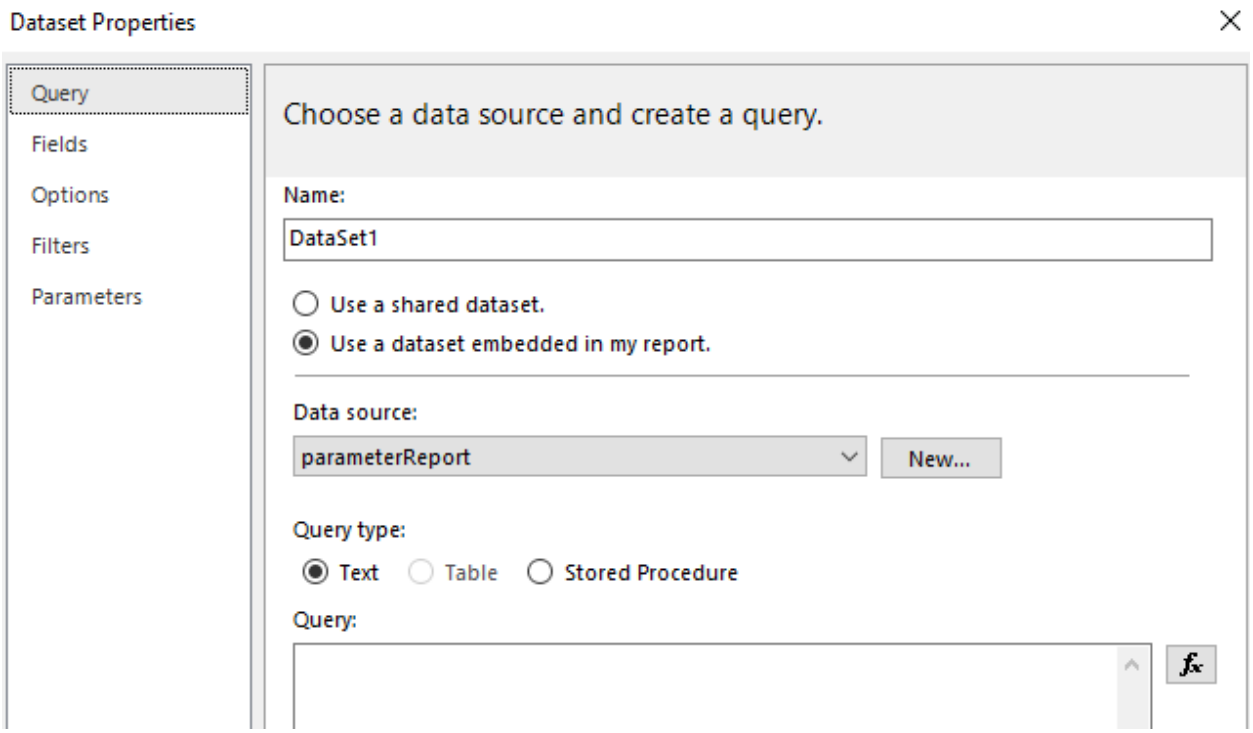


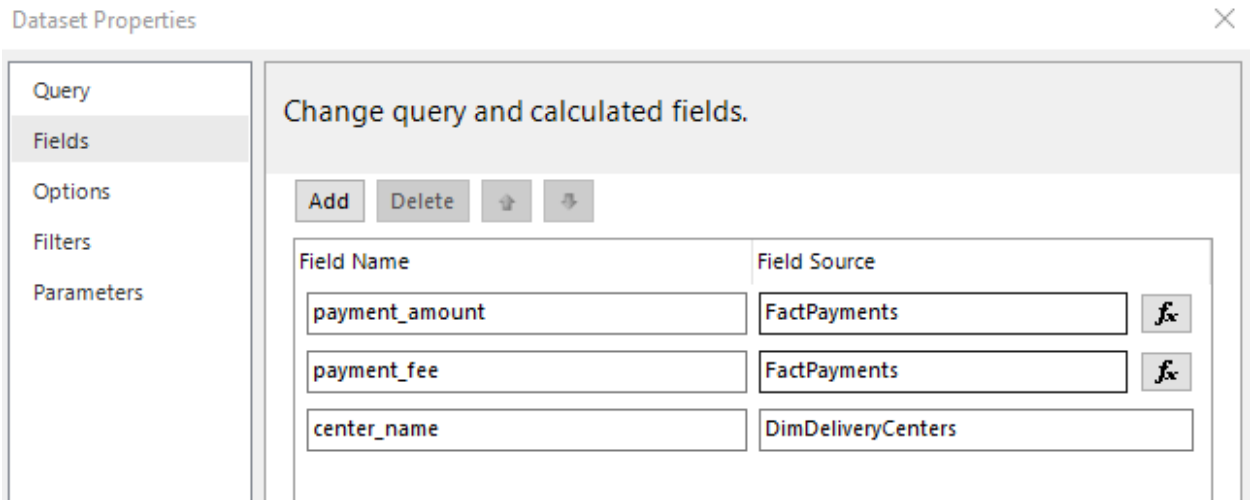
Report 3 – Report with more than one parameter

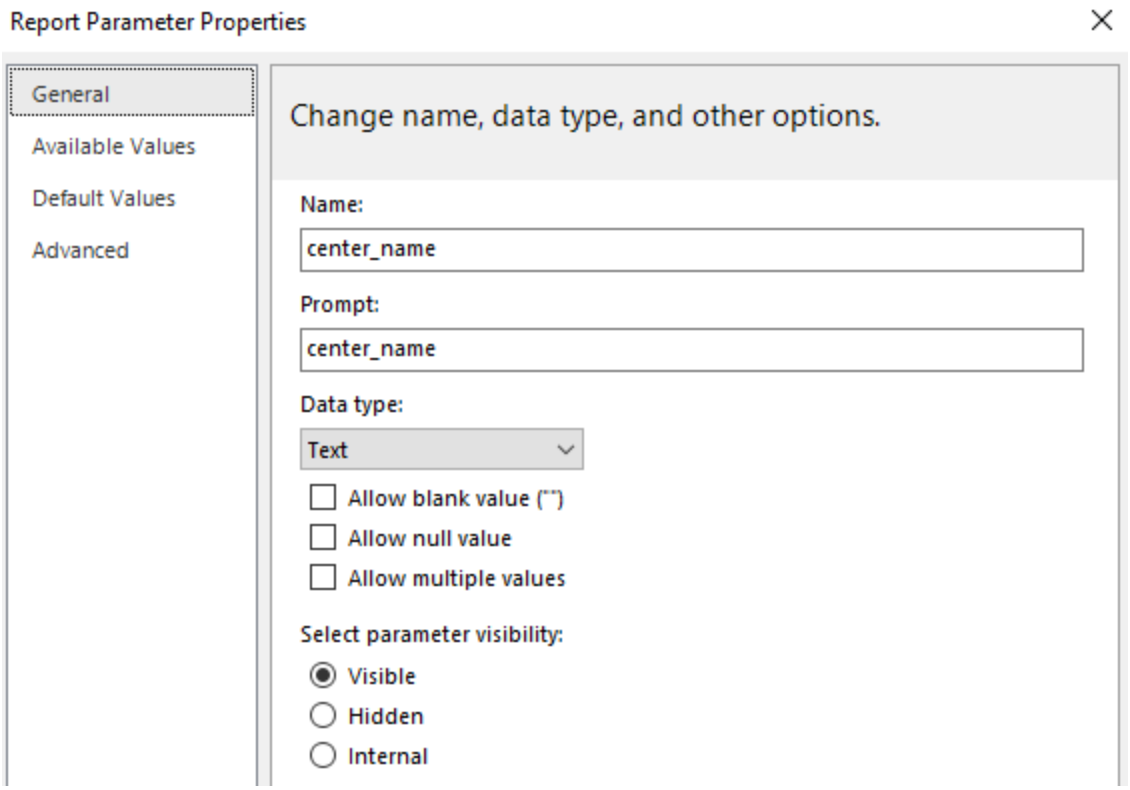
A new report was created, and the data source and the dataset were added, and a table was inserted which has center_name, hub_name, and Order_created_day as columns. The expected report has two parameter lists center name list and hub names list. When the one type of a list item is selected, the other list is updated according to it.

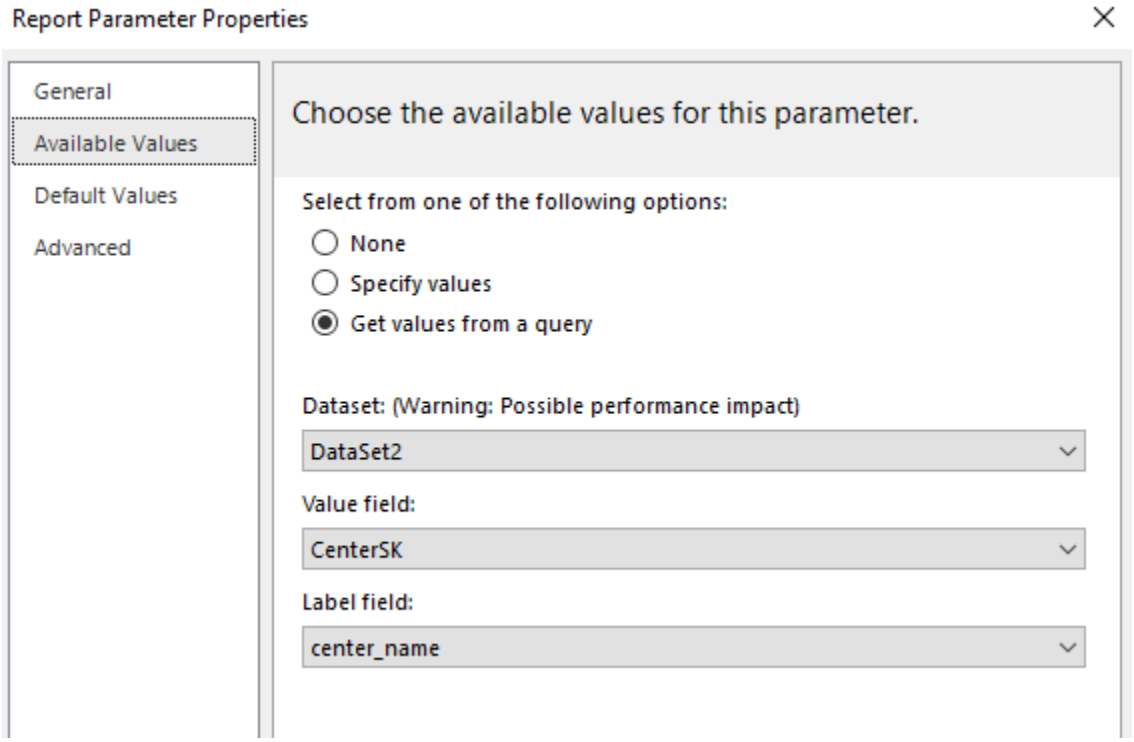





Dataset Properties
Query
Fields
Options
Filters
Parameters
Choose a data source and create a query.
Name:
DataSet1
Use a shared dataset.
Use a dataset embedded in my report.
Data source:
parameterReport
New...
Query type:
Text
Table
Stored Procedure
Query:


Dataset Properties
Query
Fields
Options
Filters
Parameters
Change query and calculated fields.
Add
Delete
Field Name
Field Source
payment_amount
FactPayments
payment_fee
FactPayments
center_name
DimDeliveryCenters


Report Parameter Properties
General
Available Values
Default Values
Advanced
Change name, data type, and other options.
Name:
center_name
Prompt:
center_name
Data type:
Text
Allow blank value ("")
Allow null value
Allow multiple values
Select parameter visibility:
Visible
Hidden
Internal


Report Parameter Properties
General
Available Values
Default Values
Advanced
Choose the available values for this parameter.
Select from one of the following options:
None
Specify values
Get values from a query
Dataset: (Warning: Possible performance impact)
DataSet2
Value field:
CenterSK
Label field:
center_name