DOMINICK’S FINE FOOD

INTEGRATED REPORT 4

BI Report Design and Implementation for DFF

along with the Integrated Group Report

PREPARED BY:

SECTION 600 GROUP 5

ABHISHEK GHOSH

MOUMITA MUKHERJEE

SZU KAI CHEN

ZHONGZHU ZHOU

(Email ID: zhongzhu.zhou@tamu.edu)

TABLE OF CONTENTS

INTRODUCTION

The term data warehousing would refer to the use of data analysis and reporting using historical and current data to creating insightful theories to assist the process of decision making. We help the managerial staff using these insights by generating analytics reports which are created by making use of Online Analytical Processing Technology (OLAP). We can improve the efficiency of the system and maximize profits and productivity of a business. In this project, our objective is to create business questions that would target sales, marketing strategy and other relevant business processes helping the organization to prosper.

Dominick's Fine Food (DFF) is a grocery store chain and subsidiary of Safeway Inc. with locations mainly in the Chicago area. We are planning to design and develop a data warehouse for this retail store chain. The data is collected from the DFF database from Chicago Booth which consists of 25 product categories in over 100 retail chain stores. Given these important data files, such as customer, location, and sale information, we would generate a bunch of tables with attributes that help managers to make decisions regarding product sales.

Business problems

Some of the key problems of DFF that we identified are as follows:

- Retail stores witness a surge in price during peak seasons such as Thanksgiving and Christmas. DFF should find out the optimal price to maximize their profit.
- Shelf management is an in-store tactic to boost sales. UPC scanners make it possible to understand the heterogeneity of local area demand. DFF is trying to figure out how it should allocate shelf space to increase the product sales.
- DFF needs to understand how age, income, household size can impact the rate of sales. Understanding these demographics could help them target customers efficiently.
- DFF has been giving away coupons and discounts but they need to understand which promotional tactic is the most beneficial for them.
- With more than 100 stores across Chicago, store-level research could help in understanding how positioning of stores could increase their profit. The reasons that shoppers choose one store over the other.

Problems we faced during identification

To deal with many data files and a huge amount of data, we faced several challenges as stated below:

- The first challenge we faced was to understand the data. With many columns in each data file, we had to check the files one by one manually. Also, using Excel tools to make them more readable. For the source data, we have four data sets and several tables to work on.

- Designing a data warehouse is a problem. Before loading data into our data warehouse, we should figure out which data is useful for future analysis and which data is not. In order to deal with this problem, we need to first draw an ERD (Entity-relationship model) to show the data we care about.
- Dirty data is another big difficulty for us to extract and load into our data warehouse. From source data sets, many values are missing or dirty. We need to utilize ETL tools to clean the data first. This process takes a lot of time. Also, this had to be done for each business question.
- The most important part is to list all the business we care about. One of the reasons why we develop the data warehouse for DFF is to help their managers to make decisions in the future. Data warehouse is not just a daily transaction system such as a database. We should figure out several situations that the data warehouse could be helpful. As a result, we need to list the business questions. Based on those questions, we build the data warehouse later.

DETAILS ABOUT THE DATA

Understanding the data

We should develop our business questions based on the source data, so we need to understand the data at the first step. There are four data sets and several tables. We check the source data by extracting small sample data from data sets and use Excel tools to show the information.

CCount: daily sales records. This data file contains the number of customers visiting the stores and purchase information, such as total sales of products and total coupons used for purchasing.

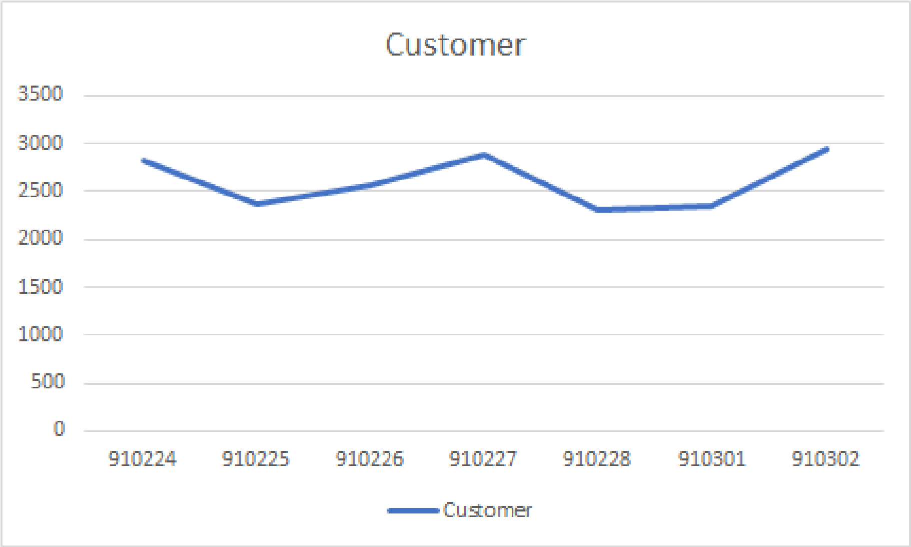


Figure 2.1.1 - Weekly report for customers in one store

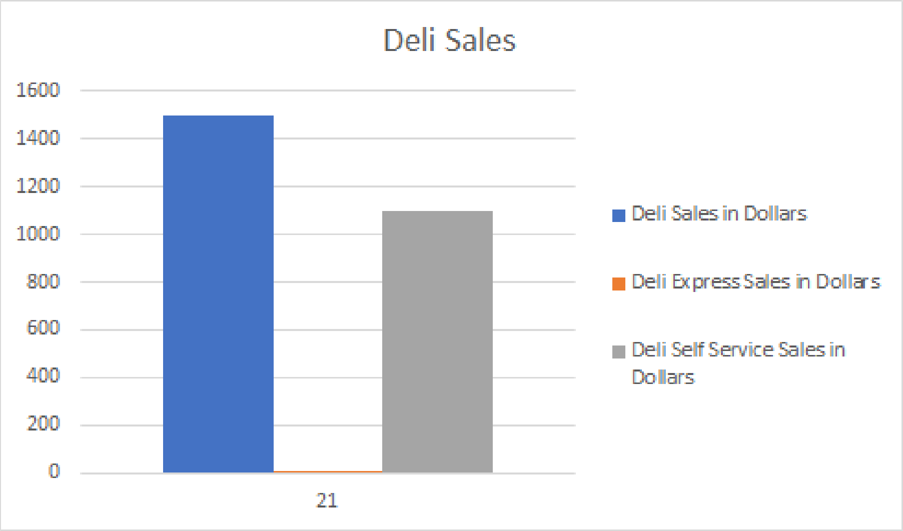
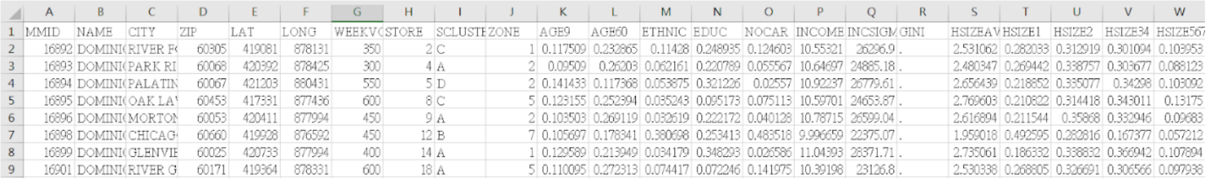


Figure 2.1.2 - Deli Sales in different types in one store

Demographics: This is the data set of census for the Chicago metropolitan area. This data provides the information about the locations, the various age groups of customers, households information, and the ability to purchase products.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	MMID	NAME	CITY	ZIP	LAT	LONG	WEEKV(	STORE	SCLUSTE	ZONE	AGE9	AGE60	ETHNIC	EDUC	NOCAR	INCOME	INCSIGM	GINI	HSIZEAV	HSIZE1	HSIZE2	HSIZE34	HSIZE567
2	16892	DOMINI(	RIVER F(	60305	419081	878131	350	2	C	1	0.117509	0.232865	0.11428	0.248935	0.124603	10.55321	26296.9	.	2.531062	0.282033	0.312919	0.301094	0.103953
3	16893	DOMINI(	PARK RI	60068	420392	878425	300	4	A	2	0.09509	0.26203	0.062161	0.220789	0.055567	10.64697	24885.18	.	2.480347	0.269442	0.338757	0.303677	0.088123
4	16894	DOMINI(	PALATIN	60067	421203	880431	550	5	D	2	0.141433	0.117368	0.053875	0.321226	0.02557	10.92237	26779.61	.	2.656439	0.218852	0.335077	0.34298	0.103092
5	16895	DOMINI(	OAK LA'	60453	417331	877436	600	8	C	5	0.123155	0.252394	0.035243	0.095173	0.075113	10.59701	24653.87	.	2.769603	0.210822	0.314418	0.343011	0.13175
6	16896	DOMINI(	MORTON	60053	420411	877994	450	9	A	2	0.103503	0.269119	0.032619	0.222172	0.040128	10.78715	26599.04	.	2.616894	0.211544	0.35868	0.332946	0.09683
7	16898	DOMINI(	CHICAG	60660	419928	876592	450	12	B	7	0.105697	0.178341	0.380698	0.253413	0.483518	9.996659	22375.07	.	1.959018	0.492595	0.282816	0.167377	0.057212
8	16899	DOMINI(	GLENVIE	60025	420733	877994	400	14	A	1	0.129589	0.213949	0.034179	0.348293	0.026586	11.04393	28371.71	.	2.735061	0.186332	0.338832	0.366942	0.107894
9	16901	DOMINI(	RIVER G	60171	419364	878331	600	18	A	5	0.110095	0.272313	0.074417	0.072246	0.141975	10.39198	23126.8	.	2.530338	0.268805	0.326691	0.306566	0.097938

Figure 2.1.3 - Locations information in Demographics

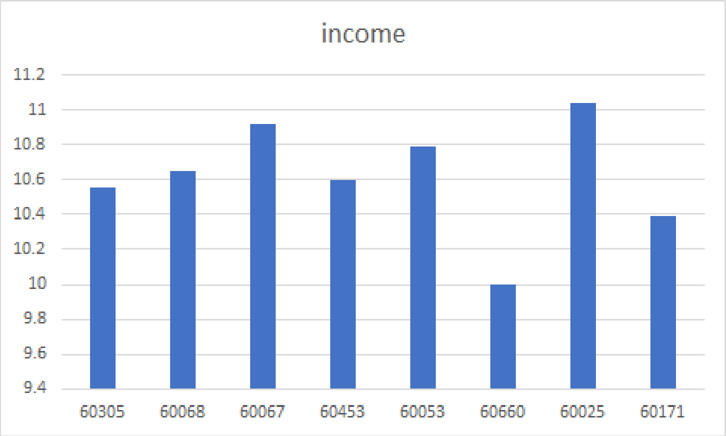


Figure 2.1.4 - income in different locations

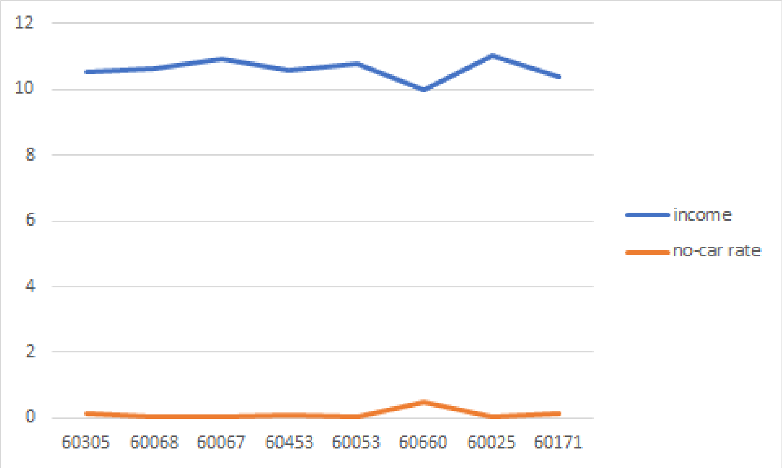
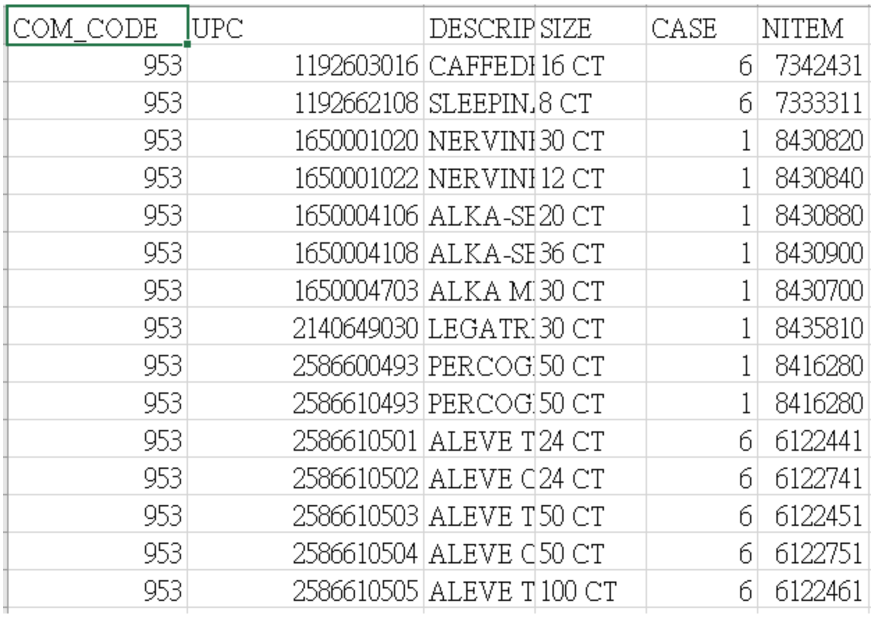


Figure 2.1.5 - income and no-car rate comparison

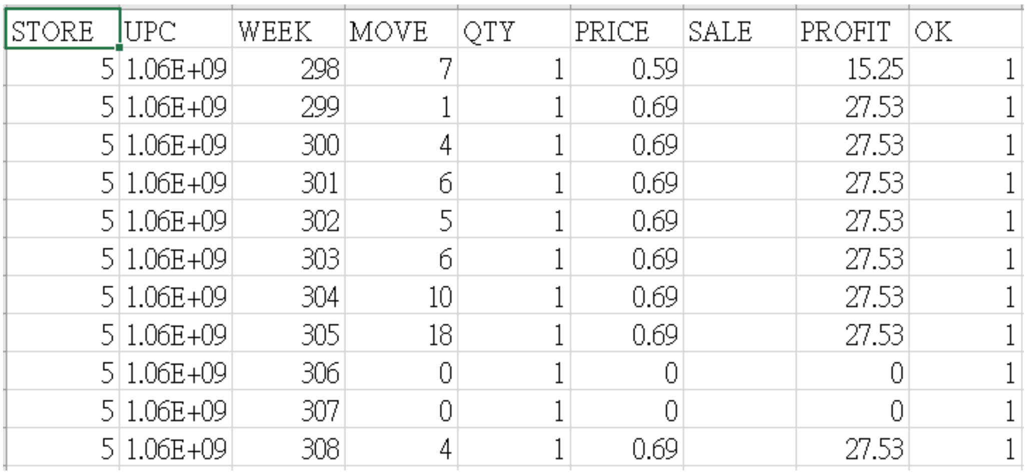
UPC: The data files describe each UPC in category, including products and their description.



COM_CODE	UPC	DESCRIP	SIZE	CASE	NITEM
953	1192603016	CAFFEDI	16 CT	6	7342431
953	1192662108	SLEEPIN.	8 CT	6	7333311
953	1650001020	NERVINI	30 CT	1	8430820
953	1650001022	NERVINI	12 CT	1	8430840
953	1650004106	ALKA-SE	20 CT	1	8430880
953	1650004108	ALKA-SE	36 CT	1	8430900
953	1650004703	ALKA MI	30 CT	1	8430700
953	2140649030	LEGATRI	30 CT	1	8435810
953	2586600493	PERCOGI	50 CT	1	8416280
953	2586610493	PERCOGI	50 CT	1	8416280
953	2586610501	ALEVE T	24 CT	6	6122441
953	2586610502	ALEVE C	24 CT	6	6122741
953	2586610503	ALEVE T	50 CT	6	6122451
953	2586610504	ALEVE C	50 CT	6	6122751
953	2586610505	ALEVE T	100 CT	6	6122461

Figure 2.1.6 - products information in one of the UPC files

Movement: this data set shows the sale information at store level.



STORE	UPC	WEEK	MOVE	QTY	PRICE	SALE	PROFIT	OK
5	1.06E+09	298	7	1	0.59		15.25	1
5	1.06E+09	299	1	1	0.69		27.53	1
5	1.06E+09	300	4	1	0.69		27.53	1
5	1.06E+09	301	6	1	0.69		27.53	1
5	1.06E+09	302	5	1	0.69		27.53	1
5	1.06E+09	303	6	1	0.69		27.53	1
5	1.06E+09	304	10	1	0.69		27.53	1
5	1.06E+09	305	18	1	0.69		27.53	1
5	1.06E+09	306	0	1	0		0	1
5	1.06E+09	307	0	1	0		0	1
5	1.06E+09	308	4	1	0.69		27.53	1

Figure 2.1.7 - Movement information

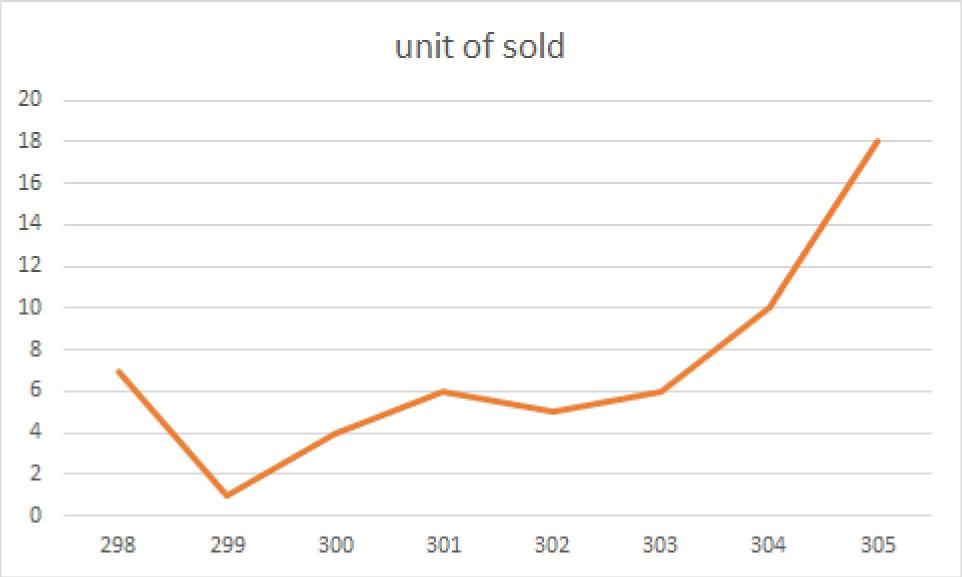


Figure 2.1.7 - weekly report for a product

DFF's store and Week decode table: The files contained in this table are the store's code, and the week code with some festival, which may affect the sales for the weekly report.

Metadata for all OLTP source files

Metadata is “data of the data”, we can use metadata to represent the data in OLTP source files. The metadata for this project are listed below.

Metadata for CCount may contain the number of customers visiting the stores, the data of record, each product's sales, and each coupons redeemed.

Variable	Description	Variable	Description
Date	Date of the Observation	GM	General Merchandise Sales in Dollars
Week	Week Number	GMCOUP	General Coupons Redeemed
Store	Store Code	GROCCOUP	Grocery Coupons Redeemed
BAKCOUP	Bakery Coupons Redeemed	GROCERY	Grocery Sales in Dollars
BAKERY	Bakery Sales in Dollars	HABA	Health and Beauty Aids Sales in Dollars
BEER	Beer Sales in Dollars	HABACOUP	Health and Beauty Aids Coupons Redeemed
BOTTLE	Bottle Sales in Dollars	JEWELRY	Jewelry Sales in Dollars
BULK	Bulk Sales in Dollars	LIQCOUP	Liquor Coupons Redeemed
BULKCOUP	Bulk Coupons Redeemed	MANCOUP	Manufacturer Coupons Redeemed
CAMERA	Camera Sales in Dollars	MEAT	Meat Sales in Dollars
CHEESE	Cheese Sales in Dollars	MEATCOUP	Meat Coupons Redeemed
CONVFOOD	Conventional Foods Sales in Dollars	MEATFROZ	Meat -Frozen Sales in Dollars
COSMCOUP	Cosmetic Coupons Redeemed	MISCSCP	Misc. Coupons Redeemed
COSMETIC	Cosmetic Sales in Dollars	MVPCLUB	MVP
COSTCOUN	Customer Count	PHARCOUP	Pharmacy Coupons Redeemed
DAIRCOUP	Dairy Coupons Redeemed	PHARMACY	Pharmacy Sales in Dollars
DAIRY	Dairy Sales in Dollars	PHOTCOUP	Photo Coupons Redeemed
DELI	Deli Sales in Dollars	PHOTOFIN	Photo
DELICOUP	Deli Coupons Redeemed	PRODCOUP	Produce Coupons Redeemed

DELIEXPR	Deli Express Sales in Dollars	PRODUCE	Produce Sales in Dollars
DELISELF	Deli Self Service Sales in Dollars	PROMCOUP	Promotion Coupons Redeemed
FISH	Fish Sales in Dollars	PROMO	Promotion Sales in Dollars
FISHCOUP	Fish Coupons Redeemed	SALADBAR	Salad Bar Sales in Dollars
FLORAL	Floral Sales in Dollars	SALCOUP	Salad Bar Coupons Redeemed
FLORCOUP	Floral Coupons Redeemed	SPIRITS	Spirits Sales in Dollars
FROZCOUP	Frozen Items Coupons Redeemed	SSDELICP	Self Service Deli Sales in Dollars

FROZEN	Frozen Items Sales in Dollars	VIDCOUP	Video Coupons Redeemed
FTGCCOUP	Food-to-Go Coupons Redeemed	VIDEO	Video Sales in Dollars
FTGCHIN	Food-to-Go Sales in Dollars	VIDEOREN	Video Rentals (Dollar Amounts)
FTGICOUP	Food-to-Go Italian Coupons Redeemed	WINE	Wine Sales in Dollars

FTGITAL	**Food-to-Go Italian Sales in Dollars**		

Metadata for Demographics may contain distribution of population and the ability to shop.

Variable Name	**Description**
age9	% of population under age 9
age60	% of population above age 60
ethnic	% Blacks & Hispanics
educ	% College Graduate
nocar	% With No Vehicle
income	Log of Median income
incsigma	Std dev of Income Distribution
hsizeavg	Average Household Size
hsize1	% of households with 1 person
hsize2	% of households with 2 people
hsize34	% of households with 3 or 4 people
hsize567	% of households with 5 or more people
hh3plus	% of households with 3 or more people
hh4plus	% of households with 4 or more people
hhsingle	% of households with 1 person
hhlarge	% of households with 5 or more people
workmom	% of working women with full-time jobs
sinhouse	% Detached Houses
density	Trading Area in Sq Miles per Capita

hval150	**% of households with value over $150,000**
hval200	% of households with value over $200,000
hvalmean	Mean Household value
single	% of singles
retired	% of retired
unemp	% of Unemployed
wrkch5	% of working women with children under 5
wrkch17	% of working women with children 6-17
nwrkch5	% of non-working women with children under 5
nwrkch17	% of non-working women with children 6-17
wrkch	% of working with children
nwrkch	% of non-working with children
wrkwch	% of working women with children under 5
wrkwnch	% of working women with no children
teltphn	% of households with telephones
mortgage	% of households with mortgages

nwthite	% of population that is non-white
poverty	% of population with income under $15,000
shopcons	% of Constrained Shoppers
shophurr	% of Hurried Shoppers

shopavid	**% of Avid Shoppers**
shopstr	% of Shopping Stranges
shopunft	% of Unfettered Shoppers
shopbird	% of Shopper Birds
shopindx	Ability to Shop
shpindx	Ability to Shop

Metadata for UPC may contain UPC number, Dominick's Commodity Code, Dominick's item code, Product Name, Product Size, and Case.

Variable	**Description**
upc	UPC number
com_code	Dominick's Commodity Code
nitem	Dominick's item Code
descrip	Product Name
size	Product size
case	Number of items in a case

Metadata for Movement may contain UPC number, week number, number of units sold, price, quantity, profit, sale, and ok.

Variable	**Description**
upc	UPC number
store	Store number
week	Week number
move	Number of unit sold
price	Retail Price
qty	Number of item bundled together
profit	Gross margin
sale	Sale code
ok	1 for valid data, 0 for trash

Entity Relationship Diagram

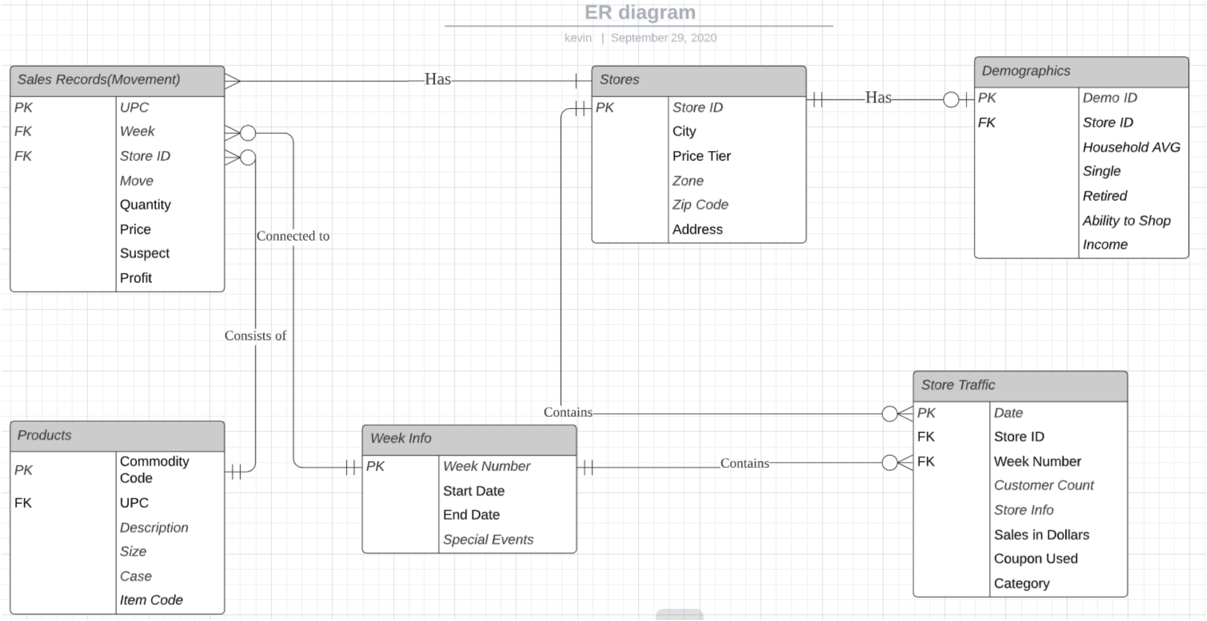


Figure 2.3.1 - Entity-relationship Diagram

DOMAIN UNDERSTANDING

The retail industry faces many challenges which become prominent while we try to implement a data warehouse or data mart solution. The research papers in the website helped us in understanding how relevant data can be derived from historical data. We studied some of the papers to learn the strategies they followed during designing and have jotted down some of the problems that are relevant to our case below.

The retail industry experiences an enormous surge in demand during the festive season - Thanksgiving and Christmas in particular. We found some of the common issues during this season highlighted in Levy's paper (1). The increase in rush causes dissatisfaction among the customers as well as the workers. Due to high demand, store workers are mostly busy with restocking and checkouts. This gives the laborers little time to focus on other tasks such as changing the price tag(1). This leads to discrepancy between the marked price and the system price. Thus, price rigidity is a common problem which has a long-term impact on the retail business(1).

Rossi's paper points out the issues that we face while evaluating historical data and how we easily overlook issues related to targeting demographics. We need to be flexible and have the ability to accommodate both observable and unobservable factors if we are required to evaluate historical information. Some of the marketing strategies fail to target customers for being too uniform. Loyalty programs which awards discounts manage to collect the products customers buy and not the factors that could influence their purchases.

The paper by Kamakura shows the issues that arise when considering marketing at chain or store level. It is difficult to track and measure how the pricing of products affect one another (3). Individual markets and retain change are drastically different. So, implementing policies across an entire chain can have a negative impact on a local area which could be difficult to predict. Sales of private labels and national brands can affect sales of local items. Data estimations are possible in a limited environment but not considering the outside environment creates a blind spot due to limited sample observations. Category management can only be done at the SKU or brand level (3). The aggregation of sales at the categorical level can result in biases which lead to wrong estimations. (3)

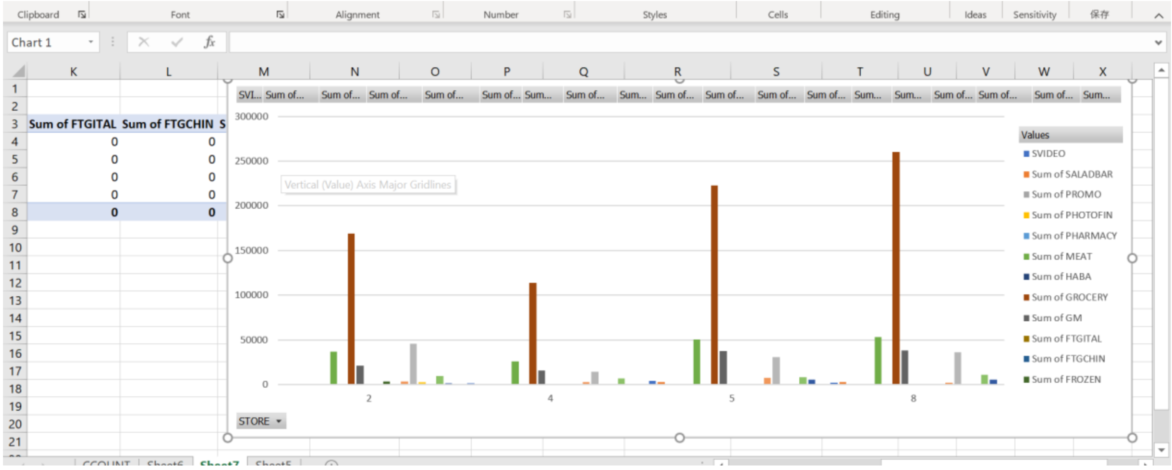
In the research paper by Nevo (4), the effects of Coupons are summarized in the increase or decrease of sales. It shows that giving coupons is better than spending a handful amount on advertisements. It also highlights that the price of commodities whose quality is not comparable to its competitors will decrease in the long term. This could be useful for our project while deriving insightful information from data warehousing for increasing sales. The paper discusses the possible reason for introducing discounts. One reason could be the introduction of a new product in the market. Discounts could attract a wide range of customers. It could also elevate the sales of a product which has been stored in the inventory for a long time, as explained in the relationships between Coupons and Shelf price (4). The perception of a customer towards a product can also change if coupons are given continuously.

From a simple internet search, we found that the common issues prevailing in the retail industry deals with the evolving marketplace. Customers are non-static. (5) Their shopping preferences vary with age, income, season. With increasing demands and digitalization, the time frame required to meet the customer needs is reducing. Customers have the option to shop both in-store and online which is resulting in the evolution of the retail industry.

BUSINESS QUESTIONS

1. ***Which category has the highest selling for each store over a span of one week?***

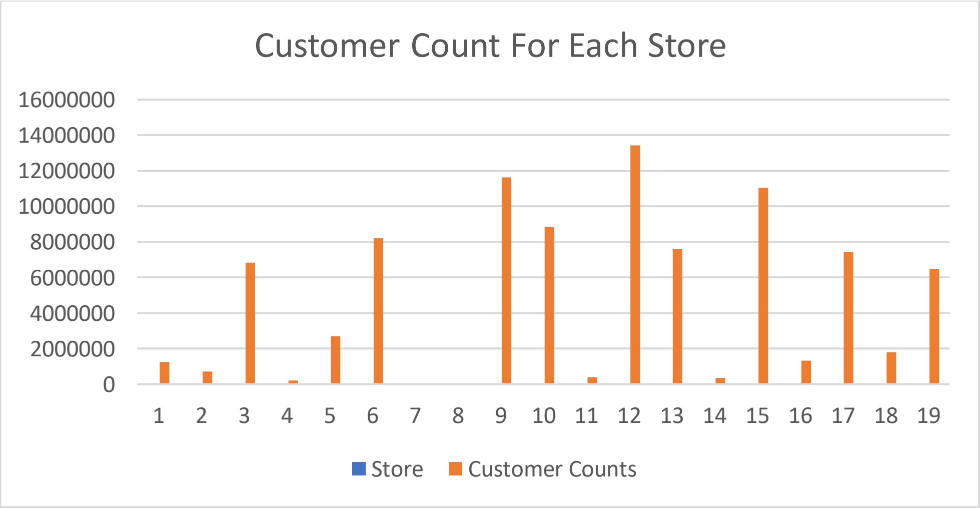
   It is important because if a category has a highest selling, the management should negotiate with its vendors to get the best price.



	K	L
1		
2		
3	Sum of FTGITAL	Sum of FTGCHIN S
4	0	0
5	0	0
6	0	0
7	0	0
8	**0**	**0**
9		
10		
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		
21		

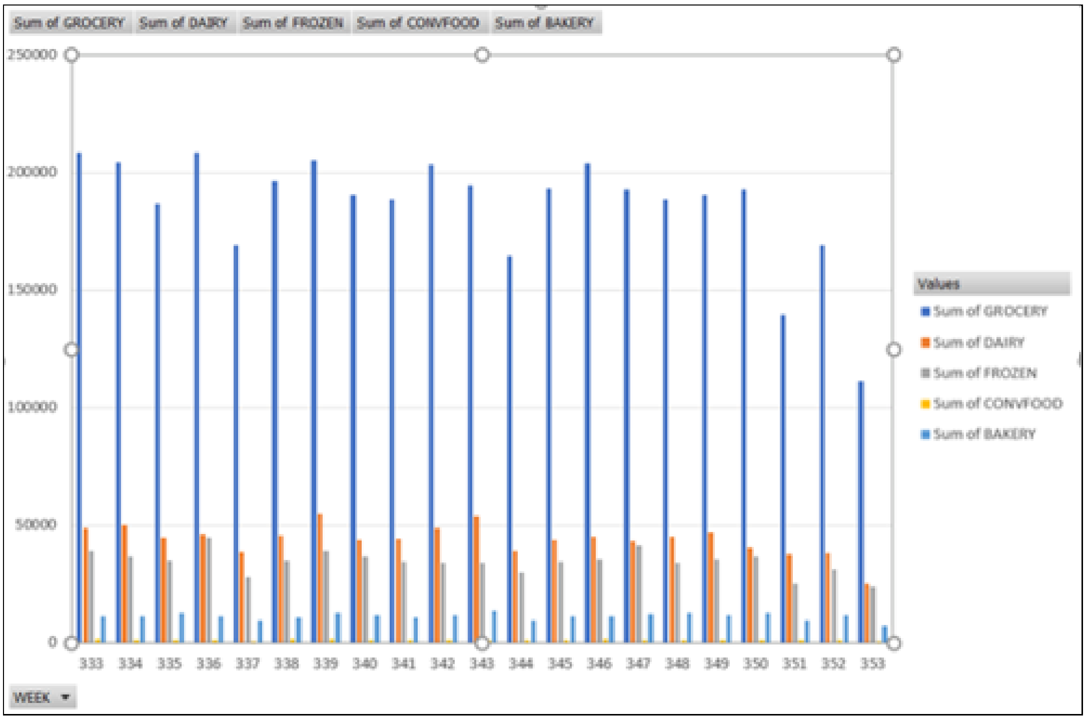
2. ***Which Store has the highest customer counts over a span of one week?***

   It is important because the highest traffic means a possible place for advertisement. When DFF decides to promote a new product, a store with highest customer traffic will be an ideal place.



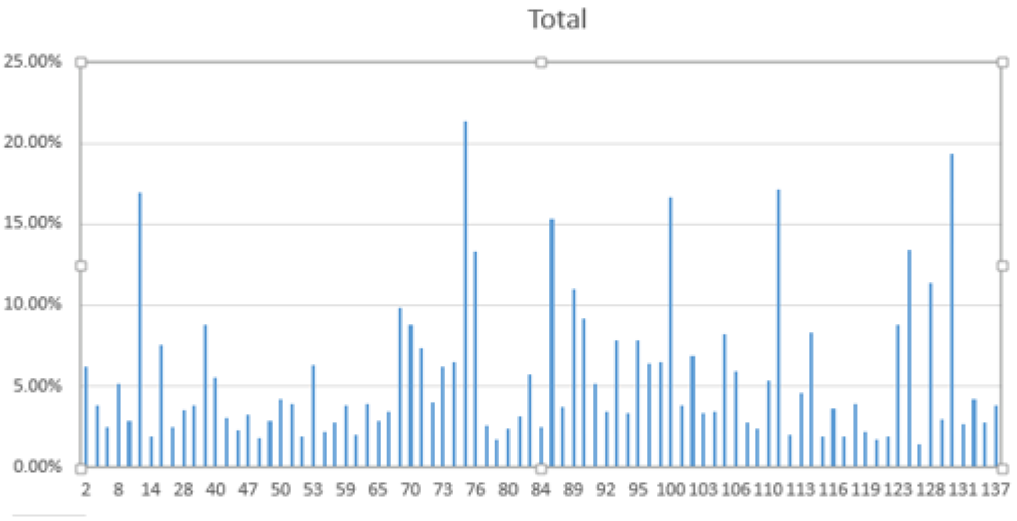
3. *Which product category sees the least selling record over a span of 6 months?*

We are checking the least sold category over 6 months to understand which product category is the least consumed by the customers. This would help DFF know which category of products customers do not prefer buying from them. This could be due to quality, promotional tactics, and price. Thus they would have to change their selling strategies for those products to increase sales of those categories.



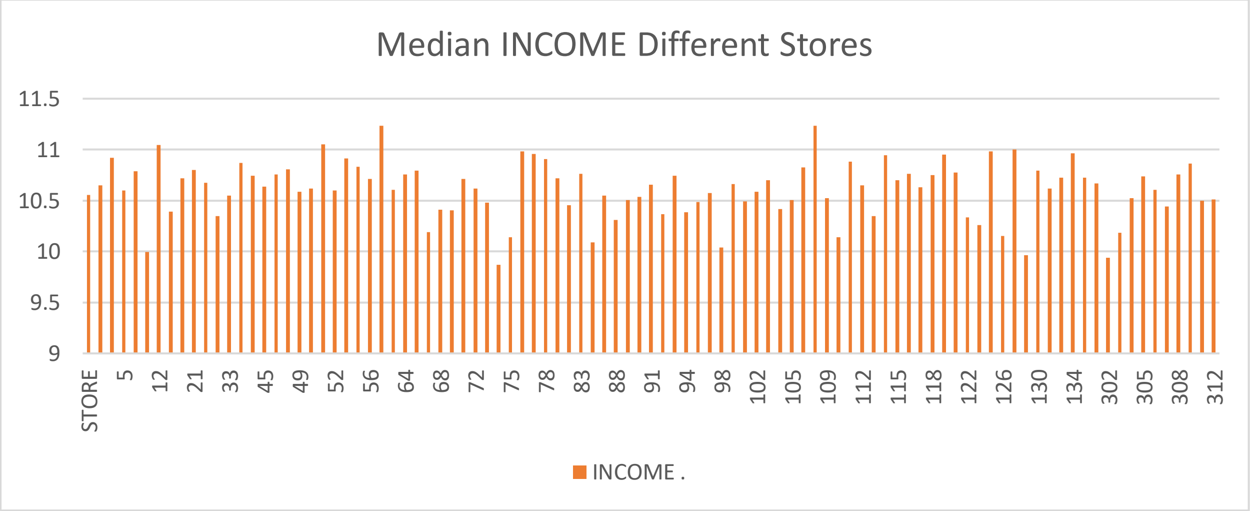
4. *Which store sees a higher number of footfalls from people below the poverty line?*

We can see some of the stores witness a comparatively higher number of people below the poverty line visiting. The difference between these stores are significant. This can help the stores determine what kind of promotional tactics should be used in these stores to lure more customers. DFF can also know what make the customers below poverty line visit these stores in particular compared to the other stores.



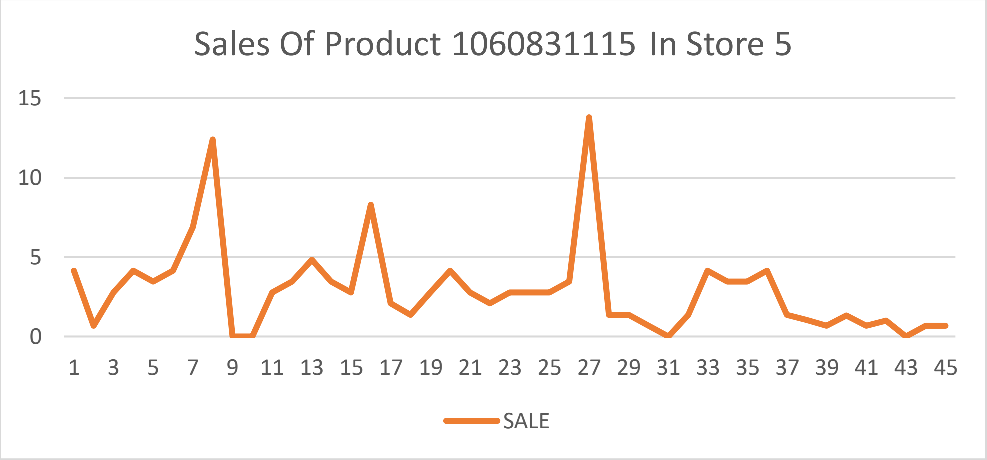
5. ***Which store is in the district with the highest median income?***

We want to pay more attention to wealthy district since it is where our profit comes from. We want to assign the most skillful employees to the most important locations.



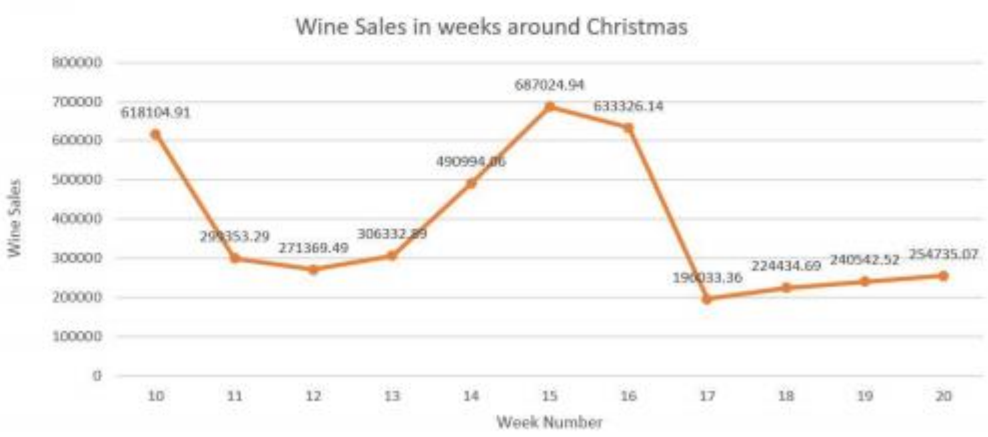
6. ***Which are the products that have shown slow or static growth?***

If a product has static or slow growth, we might need to stop selling it in our store so that we could save storage space for other popular products.



7. ***What is the percentage contribution of beer category towards sales during peak holiday season?***

Some products witness a significant increase in sales during the holiday season such as beer. As we see from the graph, week 13 to week 17 which is the period between Christmas to New Year saw a sudden spike. This could help DFF find which products witness seasonal growth in sales. DFF can manage products more effectively.



8. ***What is the trend of coupons redeemed identified by customer visits over a span of 7 weeks?***

Coupon is a tool we use to attract customers. We determined how launching coupons introduced customer visits. We could repeat similar tactic for other products and see the impact. This will help us to identify the best promotional strategies.

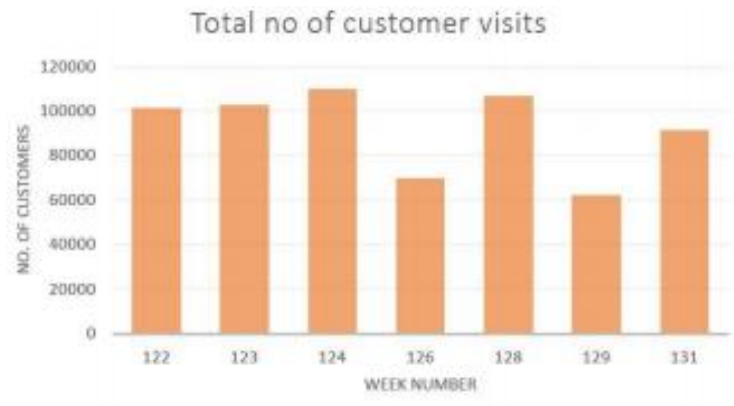


Figure: Count of customers per week

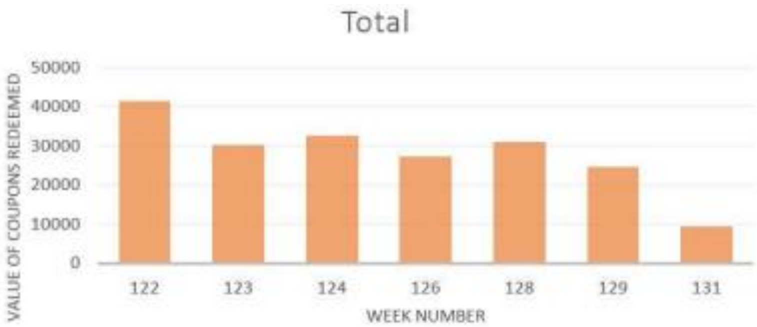
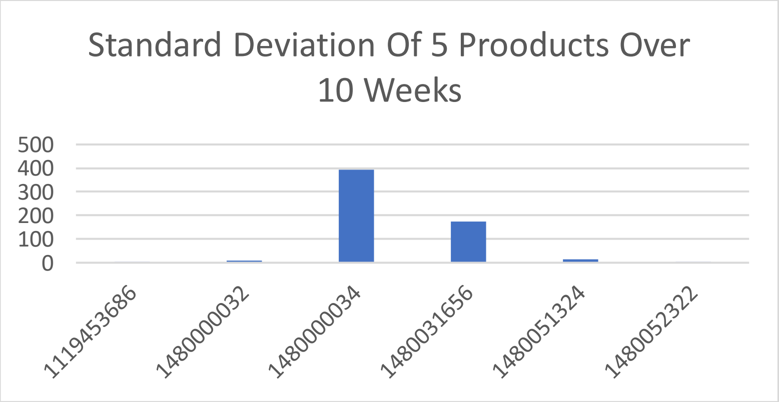


Figure: Counts of coupons redeemed per week

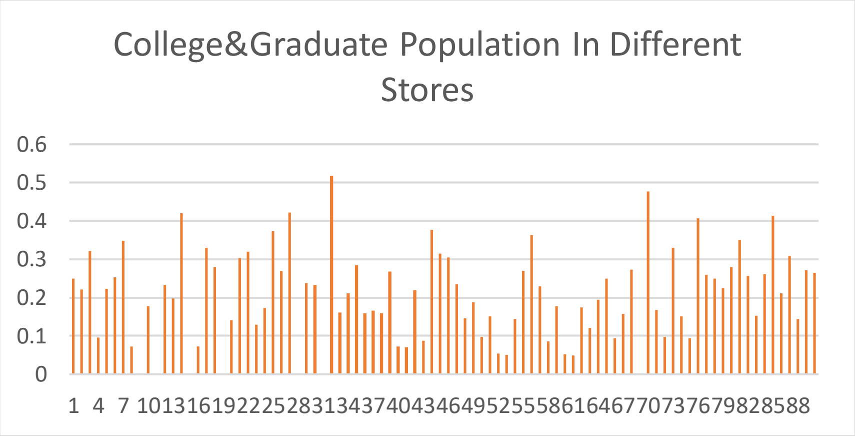
9. ***Which products sales record is the most volatile over a span of 10 weeks?***

For products with unstable sales records, we want to develop a flexible contract with our vendors to make sure we get the best deal.



10. Which store is in a district with the most college students?

If there are many college students in a district, it means more chance for us to recruit part-time workers. Part-time workers will save us money for employee salaries.



INDEPENDENT CONFORMABLE DATA MART DESIGN USING KIMBALL APPROACH

We are using the Kimball approach to design the data warehouse for Dominick Fine Foods retail store. We design an independent and conformable data mart to answer our business questions. The dimensional modelling technique which uses the STAR schema approach is being used to create the data marts. The following are the detailed steps we have followed to design our data marts to answer the proposed business questions:

STEP 1: Requirement Analysis

We have identified the final 5 business questions which are the most important for our business. They are as follows:

BQ 1: Which category has the highest selling for each store over a span of one week?

BQ 4. Which store sees a higher number of footfalls from people below the poverty line?

BQ 6. What are the products that have shown slow or static growth?

BQ 9. Which product sales record is the most volatile over a span of 40 weeks?

BQ 10. Which store is in a district with the most college students?

STEP 2A: List Dimensions

We have answered the business questions by creating the following dimension tables:

- Time
- Category
- Products
- Dominick Store
- District Demographic

STEP 2B: List Data Marts

We are creating 4 data marts which corresponds to our 5 business questions:

- **DM1- Store Demographics**: Answer business questions 10
- **DM2- Category Sales History**: Answer business question 1
- **DM3- Product Sales History**: Answer business question 6,9
- **DM4- Store Customer Count History**: Answer business question 4

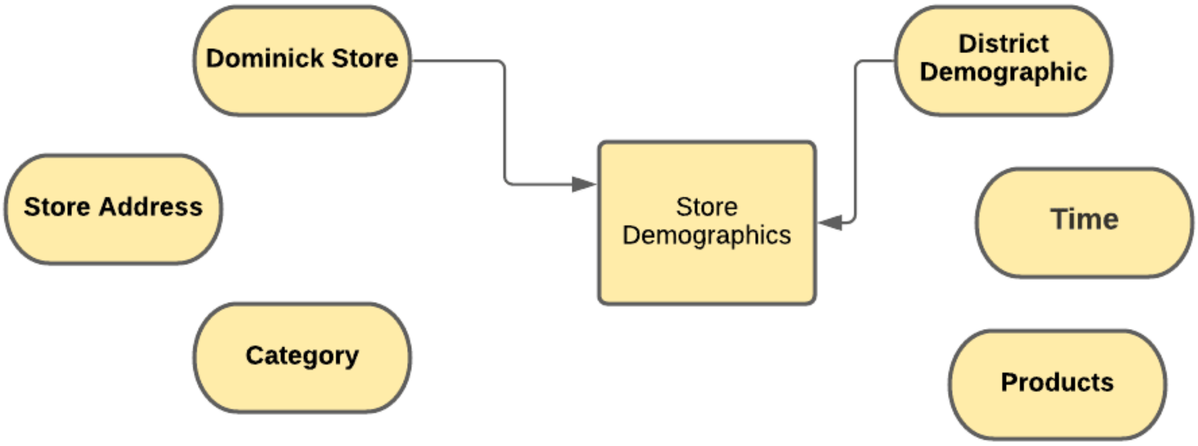
STEP 2C: Develop Matrix

	Time	Category	Products	Dominick Store	District Demographic
Store Demographics				✓	✓
Category Sales History	✓	✓		✓	
Product Sales History	✓		✓		
Store Customer Count History	✓			✓	✓

STEP 3A: Develop Fact Tables

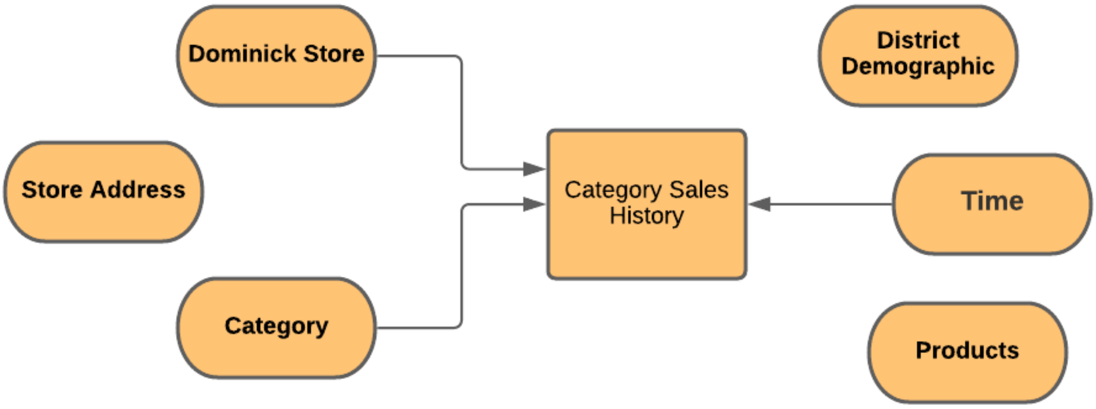
DM1 - Store Demographics

Grain: Each record represents a physical store of Dominick



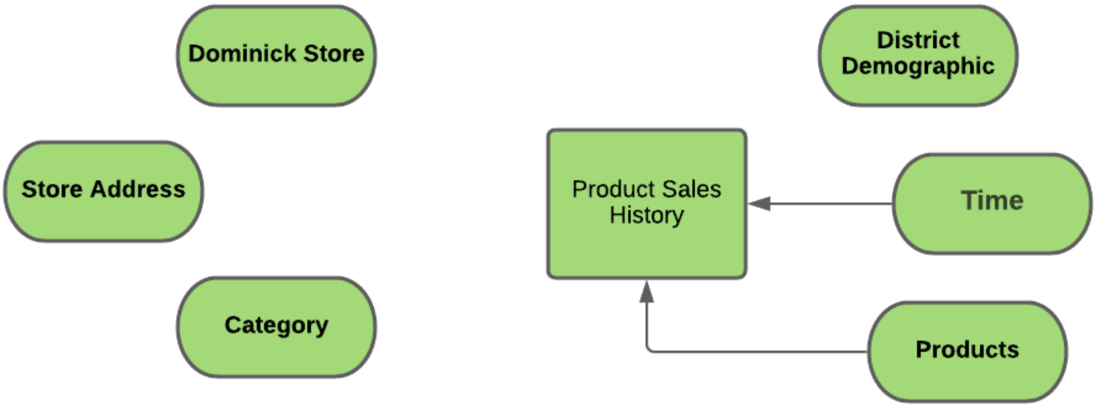
DM2 - Category Sales History

Grain: Each record represents the total sales record for a specific category during a week



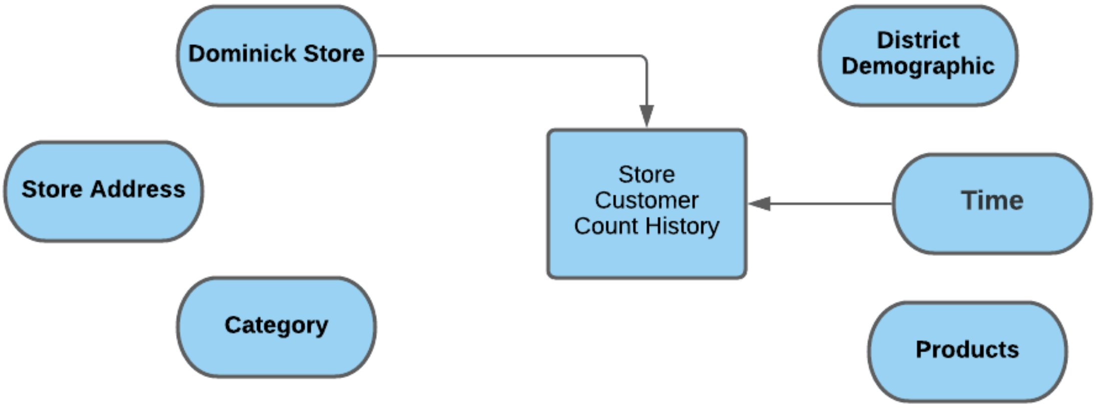
DM3 - Product Sales History

Grain: Each record represents the total amount of products sold during a week for all Dominick business.



DM4 – Store Customer Count

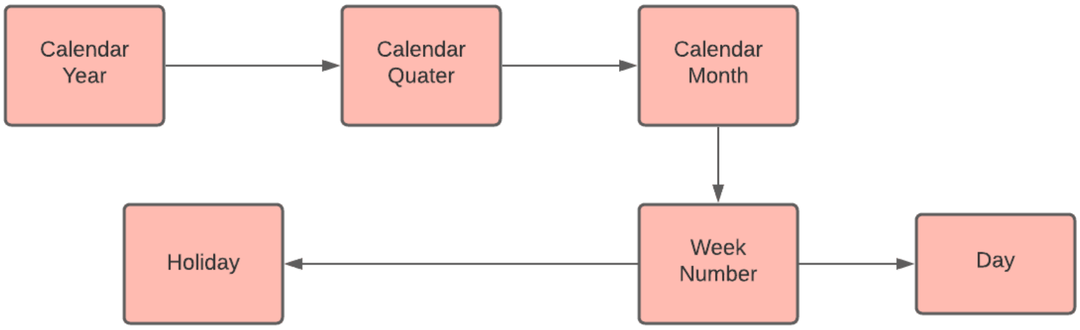
Grain: Each record represents the customer traffic number for each store for one week



STEP 3.B Develop Dimension Tables

Time Dimension

The time dimension table stores information regarding the time attributes of all DFF retail stores. The data is referenced from the DimDate table. The data can be used to calculate sales and customer counts over a certain period of time across stores.

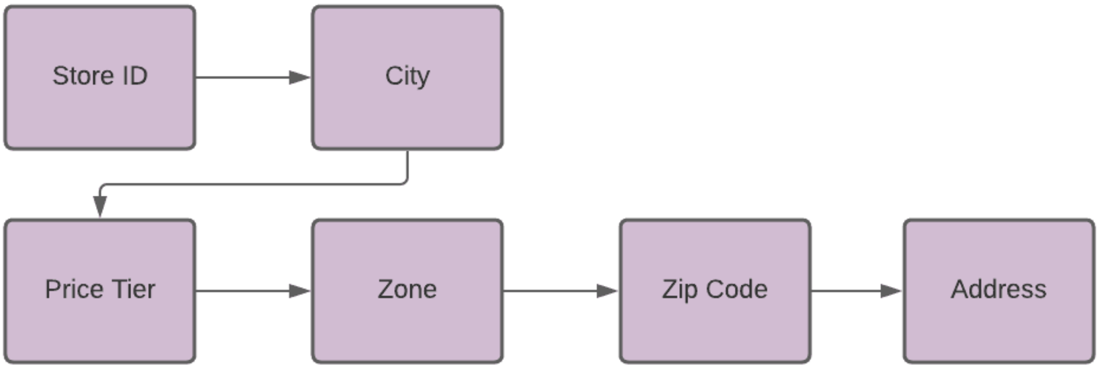


It contains the following attributes:

- **CalendarYear** - This attribute is used to provide information regarding the years of operation. It is derived using the start and end date fields from the Week decode table.
- **CalendarQuarter** - This attribute provides information regarding the quarter for which we are aggregating data. It is derived using the start and end date fields from the Week's decode table.
- **CalendarMonth** - This attribute provides information regarding the month of operation. It is derived using the start and end date fields from the Week decode table.
- **WeekNumber** - This attribute stores data regarding the week in which the operations under consideration took place.
- **Day**- This attribute provides information regarding a day of operation. It is derived using the start and end date fields from the Week decode table.
- **Holiday**- This attribute is used to identify any special day or holiday in a year.

Store Dimension

The store dimension table contains information about each store and its location. The table can be used to compute the customer distributions and the product sales distribution across stores.

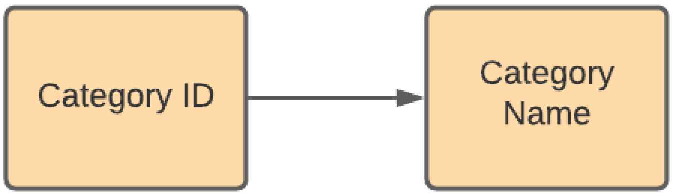


It has the following attributes:

- **StoreID**- This is a primary key that uniquely identifies each store of DFF across the country.
- **City** - This attribute tells city names where the store is located. It helps in identifying profitable locations where business operations are flourishing
- **PriceTier** - This attribute shows the price bracket each store falls into. It can identify what type of customers visit the store. For example, a store with less price tier would be visited by college students
- **Zone** - This attribute provides the information regarding the zone in which each store falls. There are 16 zones in which the DFF stores are segregated.
- **ZipCode** - This attribute stores the zip code information for each DFF store.
- **Address** - This attribute gives the exact location of each store for better understanding of store dynamics.

Category Dimension

The category dimension table contains information about a product category. This can be used to answer the question related to the highest sale of a product category. It segregates products into different data sets based on product categories.

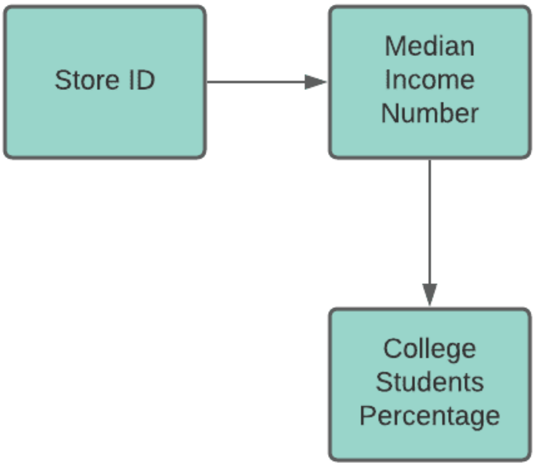


It has the following attributes:

- **CategoryID**: It is the unique primary key which would be used to identify a particular category
- **CategoryName**: This attribute provides information regarding the category. Many different products can have the same category name which can be aggregated to find out which category has the highest sales over a period of time.

Store Demographic Dimension

The store demographic dimension table is used to store the details of the customers that visit the store. Details regarding the age and their income are stored.

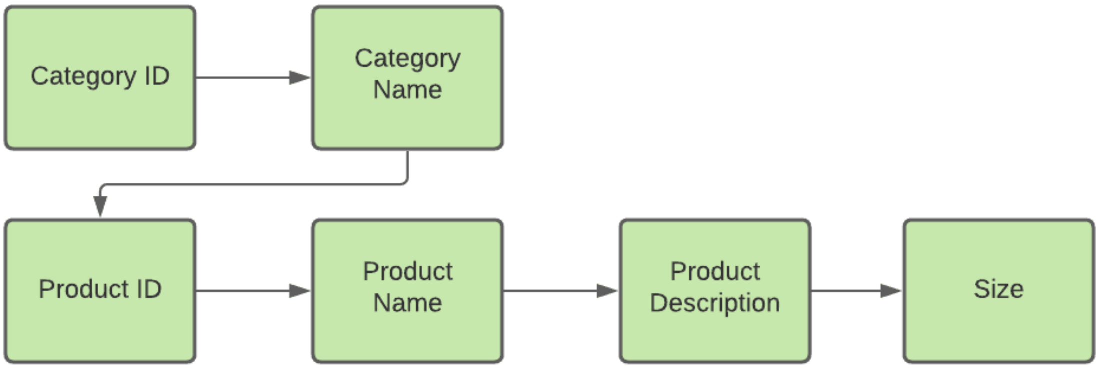


It has the following attributes:

- **StoreID**: It is the primary key which uniquely identifies a particular store
- **Median Income Number**: This attribute provides information regarding the median income of the customers that visit the stores
- **College Students Percentage**: This attribute identifies the proportion of college students who visit the store. This would help specific stores target college students and sell items relevant to students.

Product Dimension Table:

The product dimension table stores information regarding all the products sold by the DFF retail store. It contains data that is used to compute aggregate functions such as sales of certain products across stores



The attributes are as follows:

- **ProductID**: This attribute uniquely identifies a particular product sold at DFF store and is generated as a surrogate key.

- **Product Description**: This attribute is used to identify the product name that can be identified through Product ID.
- **ProductName**: This attribute is used to identify the product by its name
- **Size** – This attribute identifies the size of the product.
- **Category ID**: This is a unique identifier which can state that the product could be differentiated into what category
- **Category Name**: This attribute gives the name of each category which could help users identify what product and what category it is.

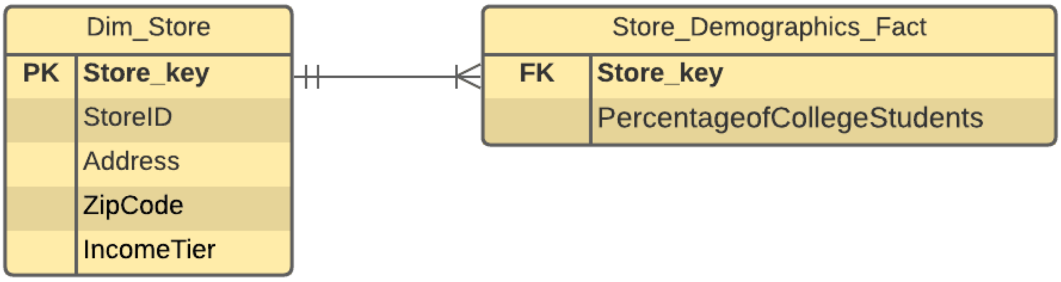
DATA WAREHOUSE SCHEMA

Star Schema

Once we have created the fact tables and dimension tables, we can use them to create the required data marts. In our case we have 4 data marts namely the store demographics, product sales, category sales and the customer count. These data marts would help us answer all our business questions gathered during the requirement analysis phase.

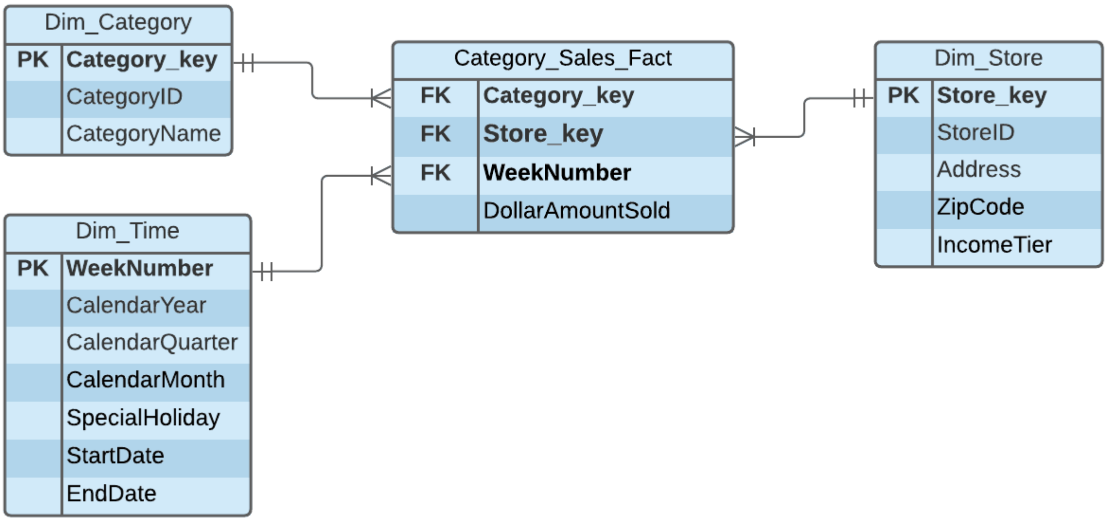
DM1 - Store Demographics

This uses a dimension table namely the store dimension table which has details regarding the location of the stores. The fact table stores aggregate information like the percentage of college students that visit the store. This particular data mart is used to answer business question 10.



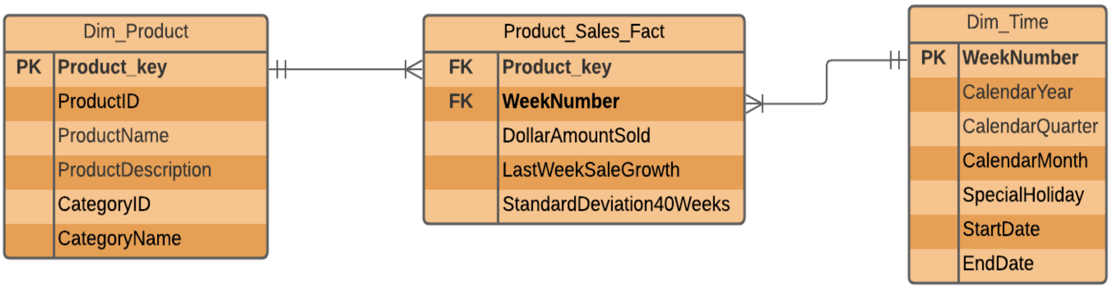
DM2 – Category Sales History

This uses three of dimension tables namely category, time and store dimension table. This holds information regarding the store location, information regarding a category of a product and time attributes such as date, quarter, month, week and day respectively in each table. The category sales are stored in the fact table in the granularity level of week and for each store. This is used to answer business question 1.



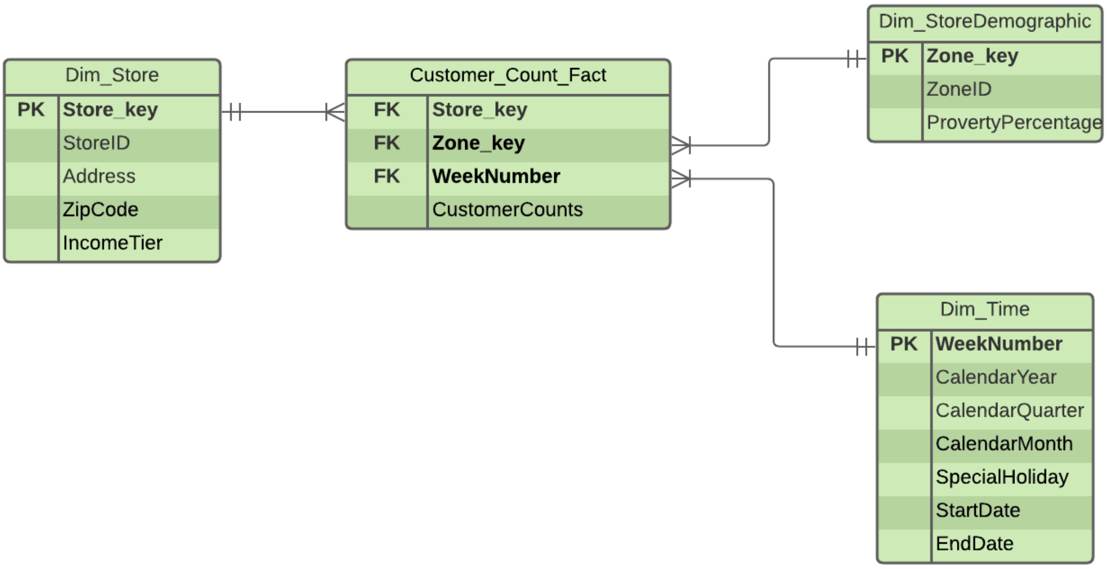
DM3 – Product Sales History

This uses 2 dimension tables namely product dimension table and time dimension table. The product dimension table includes details about the product and its category. The store and time dimension table are used for the same information as described above. The fact table here is the product sales fact table which has an aggregate field which shows the standard deviation of 40 weeks. This provides information about the sales of the product in a 40 weeks span. This data mart can be used to answer business questions 6 and 9.

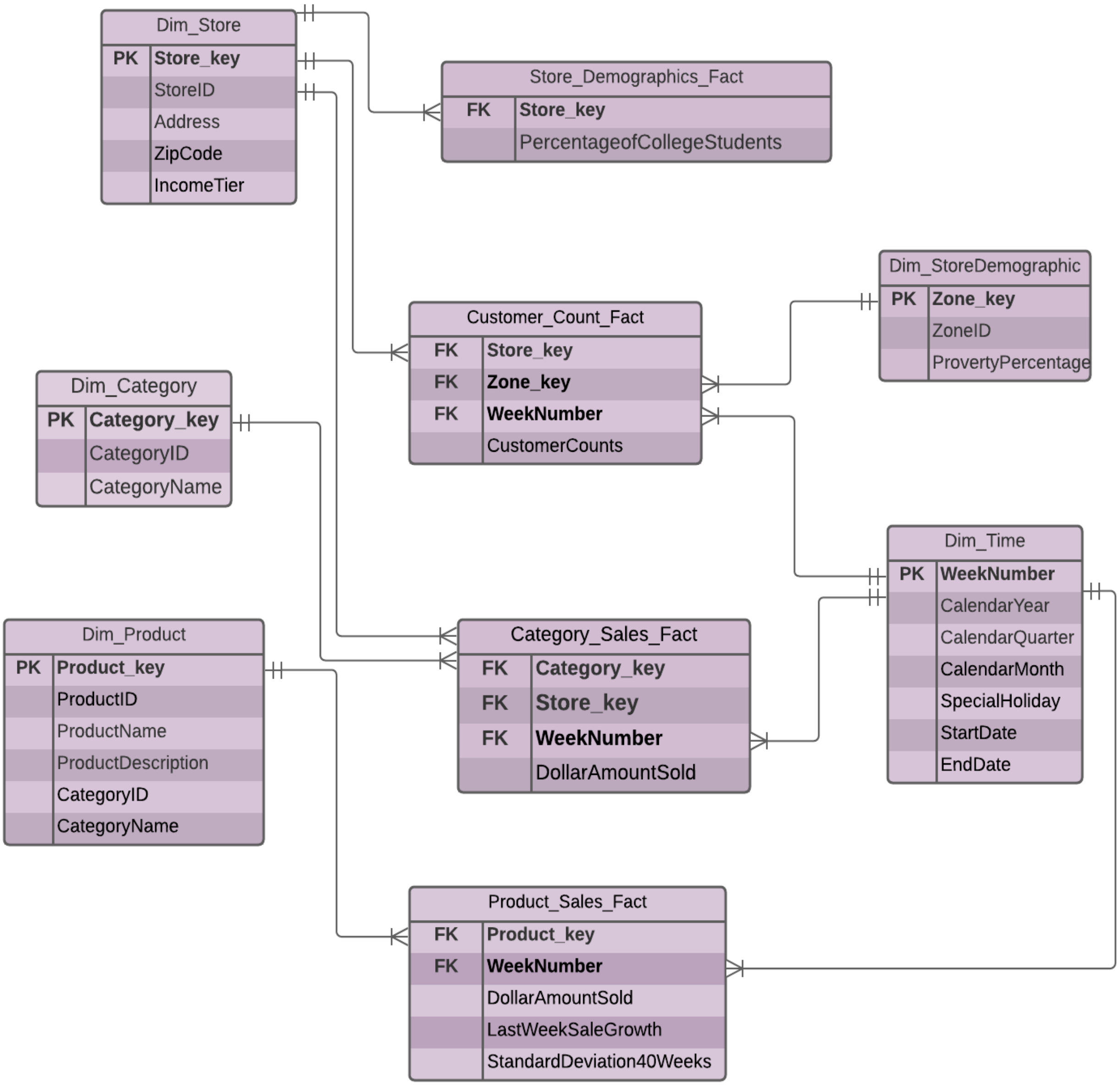


DM4 – Store Customer Count History

This data mart consists of 3 dimension tables which are the store, store demographics and time dimension tables. The store demographics identify the median income of the customers visiting the stores, the store dimension table shows the locations of the store and time identifies the time of visit. The customer count fact table has the aggregate field of customer count and shows the customers who are below poverty line visiting the store in a week. It is used to answer business question 4.



DATA WAREHOUSE SCHEMA



MAPPING TABLES

Store Demographics Fact Mapping Table

Data Warehouse Fact Table	Data Warehouse Fact Attribute	Source Table	Source Table Attributes	Mapping Function
STORE_DEMOGRAPHICS_FACT	Store_key	Store Dimension Table	-	Primary Key/Surrogate Key of Store Dimension Table
	PercentageofCollegeStudents	Demographics	-	Derived using formula from source table

Customer Count Fact Mapping table

Data Warehouse Fact Table	Data Warehouse Fact Attribute	Source Table	Source Table Attributes	Mapping Function
CUSTOMER_ COUNT_FACT	Store_key	Store Dimension Table	-	Primary Key/Surrogate Key of Store Dimension Table
	WeekNumber	Time Dimension Table	Week Number	Primary Key/Surrogate Key of Time Dimension Table
	CustomerCounts	CCount Table	CustCoun	Copy directly from source to target
	Zone_key	Store Demographic Dimension Table	-	Primary Key/Surrogate Key of Store Demographic Dimension Table

Category Sales Fact Mapping Table

Data Warehouse Fact Table	Data Warehouse Fact Attribute	Source Table	Source Table Attributes	Mapping Function
CATEGORY _SALES_FACT	Category_key	Category Dimension Table	-	Primary Key/Surrogate Key of Category Dimension Table
	WeekNumber	Time Dimension Table	Week Number	Primary Key/Surrogate Key of Time Dimension Table
	Store_key	Store Dimension Table	-	Primary Key/Surrogate Key of Store Dimension Table
	DollarAmountSold	Movement Table	Sale Price	Derived

Product Sales Fact Mapping Table

_Data Warehouse Fact Table	Data Warehouse Fact Attribute	Source Table	Source Table Attributes	Mapping Function
PRODUCT_ SALES_FACT	WeekNumber	Time Dimension Table	Time ID	Primary Key/Surrogate Key of Time Dimension Table

	Product_key	Product Dimension Table	-	Primary Key/Surrogate Key of Product Dimension Table
	DollarAmountSold	Movement Table	Quantity, Price	Derived
	LastWeekSaleGrowth	Movement Table	Quantity, Price	Derived
	StandardDeviation40weeks	Movement Table	Quantity, Price	Derived

Mapping table for Dimension Tables

Store Dimension Mapping Table

Data Warehouse Dimension Table	Data Warehouse Attribute	Source Table	Source Table Attribute	Mapping function
Dim_Store	Store_key	STORE	-	Surrogate
	StoreID		Store ID	Copy directly from source to target
	IncomeTier		Price Tier	Sort sales with price
	ZipCode		Zip Code	Copy directly from source to target
	Address		Address	Copy directly from source to target

Time Dimension Mapping Table

Data Warehouse Dimension Table	Data Warehouse Attribute	Source Table	Source Table Attribute	Mapping function
Dim_Time	WeekNumber	WEEK INFO	Week Number	Copy directly from source to target
	StartDate		Start Date	Copy directly from source to target
	EndDate		End Date	Copy directly from source to target
	SpecialHoliday		Special Events	Copy directly from source to target
	CalendarYear		Calendar Year	Derived
	CalendarMonth		Calendar Month	Derived
	CalendarQuarter		Calendar Quarter	Derived

Store Demographics Dimension Mapping Table

Data Warehouse Dimension Table	Data Warehouse Attribute	Source Table	Used	Source Table Attribute	Mapping function
Dim_StoreDemographic	ZoneID	Demographics	Yes	Demo ID	Copy directly from source to target
	PovertyPercentage		Yes	Income	Copy directly

					from source to target

Product Dimension Mapping Table

Data Warehouse Dimension Table	Data Warehouse Attribute	Used	Source Table	Source Table Attribute	Mapping function
Dim_Product	ProductID	Yes	Products	Commodity Code	Copy directly from source to target
	ProductName	Yes		Description	Separate staging attribute into two dim attributes
	ProductDescription	Yes		Description	Separate staging attribute into two-dimension attributes
	Product_key	Yes		Commodity Code	Surrogate

Category Dimension Mapping Table

Data Warehouse Dimension Table	Data Warehouse Attribute	Source Table	Source Table Attribute	Mapping function
Dim_Category	CategoryID	UPC	Category ID	Copy directly from source to target
	CategoryName		Category Name	Copy directly from source to target
	Category_key	-	-	Surrogate

BUSINESS QUESTIONS AND STAR SCHEMA JUSTIFICATION

1. ***Which category has the highest selling for each store over a span of one week?***

   For question 1, we need to figure out which category has the highest selling for each store over a span of one week. We have a category sale history fact table and three dimension tables. Time dimension table could provide time information to compare the category sale. Category dimension table provides the comparison of sales for a specific category. Besides, the Store dimension table provides the comparison of sales for a particular store. Combining those dimension tables, we can get the answer for question 1.

4. ***Which store sees a higher number of footfalls from people below the poverty line?***

   We need to find the store that sees a higher number of footfalls from people below the property line. For this question, we have a fact table called store customer count history with three of dimension tables. Time dimension table provides us the time measurement. Store dimension table provides us with information about each store. In the district demographic dimension table, we can find which location is below the property line. Altogether, the fact table could show which store has a higher number of footfalls in a particular period.

6. ***What are the products that have shown slow or static growth?***

   In question 6, we need to show which products have slow selling growth. We have a fact table, product sales history, and three dimension tables. As usual, we have a time dimension table to show the timeline. Second, the products dimension table can provide selling records for all the products in every store. Finally, we can use those tables to answer this question.

9. ***Which product sales record is the most volatile over the weeks?***

   For this question, we need to observe which product sales have much bigger change among all the products in a period. Like question 6, the time dimension table could help us to show the timeline. To be more specific, we need to expand the time to the span of 40 weeks. Besides, we use the products dimension table to obtain each product's sales. Finally, we can use the time dimension table and products table to figure out the answer for this question.

10. ***Which store is in a district with the most college students?***

   For question 10, we need to show the information about each store’s customer. We have a fact table called store demographics. Store dimension table takes care of the information about the store, such as the store's address. District demographic dimension table provides us the number of college students in each store’s customer. Combining those two tables, we can answer this question.

PHYSICAL DESIGN PLAN

1. Data aggregate plan

Storing the data in the lowest granularity level would help the users move up and down any level of aggregation they want and extract the specific information for their report. However, it could impact the time and performance. So, we need to put ourselves in the shoes of a data architect and consider the granularity level we want to use for storing data. A data aggregation would significantly improve the time and performance of the data warehousing. Before we dive in, we also need to consider our business requirement and understand what aggregation plan the best for us is. In our case, business question 4 requires us to calculate a standard deviation of 40 weeks. So, we precalculated the result for each week starting from week 41. For example, the standard deviation of week 41 will be based on data from week 1-40. Similarly, the standard deviation of week 42 will be based on data from week 2-41 and so on. For other business questions, we have category sales and customer traffic which needs to be calculated at granular level. So, we keep the sales records and customer traffic at granular level in our fact table to fit our business requirement.

2. Indexing plan

Indexing the data warehouse will reduce the amount of time taken to query results but too many indexing could result in slow loading of data. Since we are dealing with a large amount of data, a proper indexing plan should be our priority. We have the option to select between two indexing techniques - bit mapping which is used for low selectivity and b-tree indexing which is used for high selectivity. Since most of the primary key columns would have high selectivity, our fact tables and dimension tables will be indexing with b-trees. It can be used for multilevel indexing thus reducing the time and efficiency of the query.

3. Data standardization plan

To maintain a consistent flow of data, we would need a data standardization plan. We should be able to enforce the standards in each phase from identifying data sources, naming them to loading the data into the warehouse. Our data warehouse should not be ambiguous which would increase complexity. We will distinctly define the names of the dimension table, fact table and attributes. This would be more comprehensible for the users. We are also following a namespacing standard for the fact and the dimension table. Our dimension table's name would be like "DIM-XXX" and the fact table name would be "FACT-XXX". We will make sure the mandatory field values are not null and the primary keys are unique. Also, the data stored in the data table should have the specified data type and no exceptions will be made. The data standardization plan helps us convey relevant information to laymen without any technical complexities.

4. Storage plan

Since we are dealing with historical data to analyze trends, the amount of data captured within the data warehouse is large. Along with historical data, we are also loading batch updates of current data. Hence the need for storage data is inevitable which not only takes into account the current requirements but also provides the ability to accommodate future batch updates.

We have 4 data marts which includes store, categories, product sales and customer counts. We plan to load the customer count data mark at per week granularity. So, we need to update the data into the data warehouse every week. The source files could be anything anfing from Excel sheets, CSV to OLTP tables. Thus, we need proper planning for Extraction, Transformation and Loading. The chances of having a new category of items look bleak but our business demand could change and there could be a new store open or a new product added. We should keep the scope of expansion in mind and thus should plan accordingly so that a new product or store could be added in the respective table in the future. In such a case, we need to have a scalability plan for bulk and batch updates of historical products or categories into the data marts. So, our storage plan would account for any future change in requirements.

ETL DEVELOPMENT PLAN

Determine the target data

Our dimension model contains 5 dimension tables and 4 fact tables. The definition for each of them at Data Warehouse is given below:

DIMENSION TABLES

a) Dim_Store

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Dim_Store]	Store_key	int
	StoreID	int
	Address	varchar
	ZipCode	int
	IncomeTier	varchar

b) Dim_Time

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Dim_Time]	WeekNumber	int
	CalendarYear	int
	CalendarQuarter	int
	CalendarMonth	int
	SpecialHoliday	varchar
	StartDate	date
	EndDate	date

c) Dim_StoreDemograhic

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Dim_StoreDemograhic]	Zone_key	int
	ZoneID	int
	PovertyPercentage	float

d) Dim_Product

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Dim_Product]	Product_key	int
	ProductID	varchar
	ProductName	varchar
	ProductDescription	varchar
	CategoryID	varchar
	CategoryName	varchar

e) Dim_Category

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Dim_Category]	Category_key	int
	CategoryID	varchar
	CategoryName	varchar

FACT TABLES

a) Category_Sales_Fact

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Category_Sales_Fact]	Store_key	int
	Category_key	int
	WeekNumber	int
	DollarAmountSold	float

b) Customer_Count_Fact

Target Table	Target Column	Target Data Type

[600_group5_datawarehouse].[dbo].[Customer_Count_Fact]	Store_key	int
	Zone_key	int
	WeekNumber	int
	CustomerCounts	float

c) **Product_Sales_Fact**

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Product_Sales_Fact]	Product_key	int
	WeekNumber	int
	DollarAmountSold	float
	LastWeekSalesGrowth	float

d) **Store_Demographics_Fact**

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Store_Demograhpics_Fact]	Store_key	int
	PercentageofCollegeStudents	float

Determine the source data

The table provided in the data manual namely, the Store table and Week table was used as our source data. We also used the CSV files namely, Customer Count file, Demographics file and Movement files. The data needed to answer our business questions were extracted from these sources. It was then cleaned and transformed according to our specific requirements.

For example, one of our business questions asked to determine the distribution of customers who are below the poverty line. We identified this data can be found from the Store Demographic table. We fetched the data from the stated table and then cleaned and transformed it according to our requirements. We also required data from the Movements table.

For another business question we were required to find the products with the lowest rate of sales. For this we required the sales data for the product categories in a particular store. The required data could be found in the CCount file. We used the needed columns from the CCount file and removed the u nnecessary ones from the file. Similarly, for one of the question we were supposed to find the most volatile product. So, we calculated the rate of change of sales of the product. We calculated the rate in excel and then loaded it in the fact table. The rate of change could be visible through the graphs and we could identify the one product which has the most ups and downs on the graph. Hence, we needed data from Movement File, Customer count and UPC to answer the rest of the business questions.

Mapping tables for staging and data mart loads

Dim_Store:

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule
[600_group5_datawarehouse].[dbo].[Dim_Store]	Store_key	int		Surrogate Key	
	StoreID	int	Dominick's Stores	Store	
	Address	varchar	Dominick's Stores	Address	
	ZipCode	int	Dominick's Stores	ZipCode	
	IncomeTier	varchar	Dominick's Stores	Price Tier	

Dim_Time

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule
	WeekNumber	int		Surrogate Key	

[600_group5_datawarehouse].[dbo].[Dim_Time]	CalendarYear	int	Weeks Decode table	Start	The start column is of format MM/DD/YY. Split it to get YY
	CalendarQuarter	int	Weeks Decode table	Start	The start column is of format MM/DD/YY. Divided MM into one to four
	CalendarMonth	int	Weeks Decode table	Start	The start column is of format MM/DD/YY. Split it to get MM
	SpecialHoliday	varchar	Weeks Decode table	Special Events	
	StartDate	date	Weeks Decode table	Start	
	EndDate	date	Weeks Decode table	End	

Dim_StoreDemograhic

Target Table	**Target Column**	**Target Data Type**	**Source Table**	**Source Column**	**Transformation Rule**

[600_group5_datawarehouse].[dbo].[Dim_StoreDemograhic]	Zone_key	int		Surrogate Key	
	ZoneID	int	Staging_DEMO	Store	Used storeID to represent zone
	PovertyPercentage	float	Staging_DEMO	poverty	Selected the places where poverty percentage is higher than 10%

Dim_Product

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule
[600_group5_datawarehouse].[dbo].[Dim_Product]	Product_key	int		Surrogate Key	
	ProductID	varchar	Staging_UPC	UPC_PRODUCT	The last five digit of UPC number identify the product
	ProductName	varchar	Staging_UPC	DESCRIP	DESCRIP in UPC is the name of the product
	ProductDescription	varchar	Staging_UPC	CASE	Used CASE to describe the number of items in a case
	CategoryID	varchar	Staging_UPC	UPC	UPC identify the categoryID
	CategoryName	varchar	Staging_UPC	CATEGORY	

Dim_Category

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule

[600_group5_datawarehouse].[dbo].[Dim_Category]	Category_key	int		Surrogate Key	
	CategoryID	varchar	Staging_Movement_New	UPC	UPC identify the categoryID
	CategoryName	varchar	Staging_Movement_New	Category Name	

Category_Sales_Fact

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule
[600_group5_datawarehouse].[dbo].[Category_Sales_Fact]	Store_key	int	Dim_Store	Store_key	
	Category_key	int	Dim_Category	Category_key	
	WeekNumber	int	Dim_Time	WeekNumber	
	DollarAmountSold	float			Sum up all the products sales in category

Customer_Count_Fact

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule

[600_group5_datawarehouse].[dbo].[Customer_Count_Fact]	Store_key	int	Dim_Store	Store_key	
	Zone_key	int	Dim_StoreDemographic	Zone_key	
	WeekNumber	int	Dim_Time	WeekNumber	
	CustomerCounts	float	CCount	CUSTCOUN	

Product_Sales_Fact

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule
[600_group5_datawarehouse].[dbo].[Product_Sales_Fact]	Product_key	int	Dim_Product	Dim_Product	
	WeekNumber	int	Dim_Time	Dim_Time	
	DollarAmountSold	float			Each product sales
	LastWeekSalesGrowth	float			Calculated the growth rate for each product

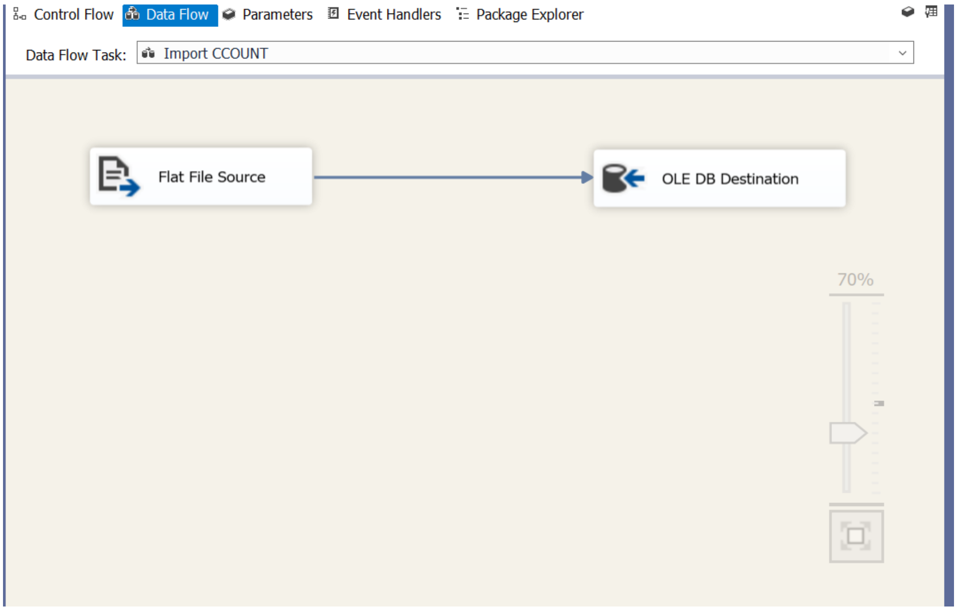
Store_Demographics_Fact

Target Table	Target Column	Target Data Type	Source Table	Source Column	Transformation Rule
[600_group5_datawarehouse].[dbo].[Store_Demograhpics_Fact]	Store_key	int	Dim_Store	Store_key	
	PercentageofCollegeStudents	float			Calculated from demo source data

Comprehensive Data Extraction Rules

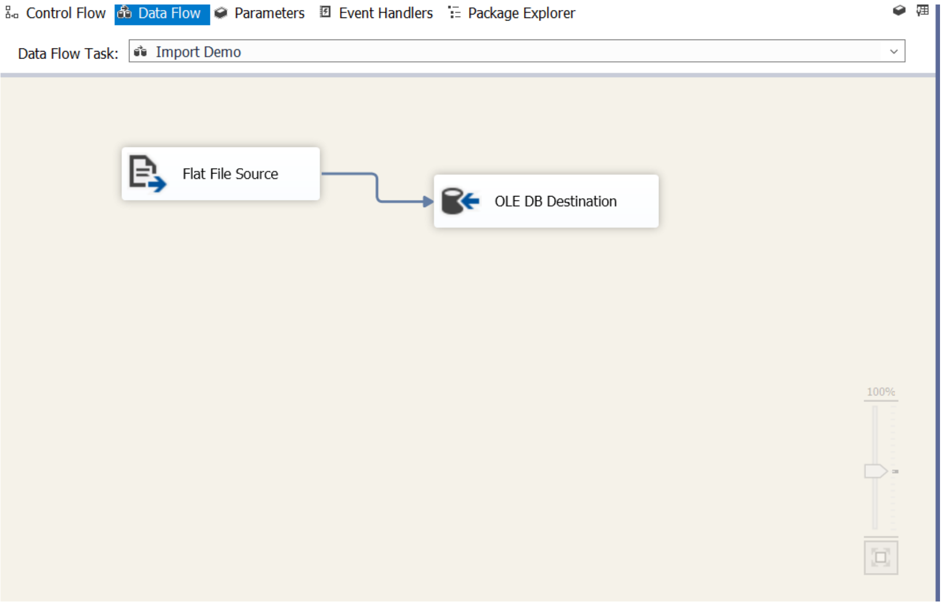
We extracted the data from the csv files and the data manuals of the DFF that were provided. We did not filter any data during this stage and all data in the csv file has been loaded into our staging database without any modifications. Please find below the steps we performed along with the screenshots attached.

A) Import CCOUNT



Data Flow Task: Import CCOUNT

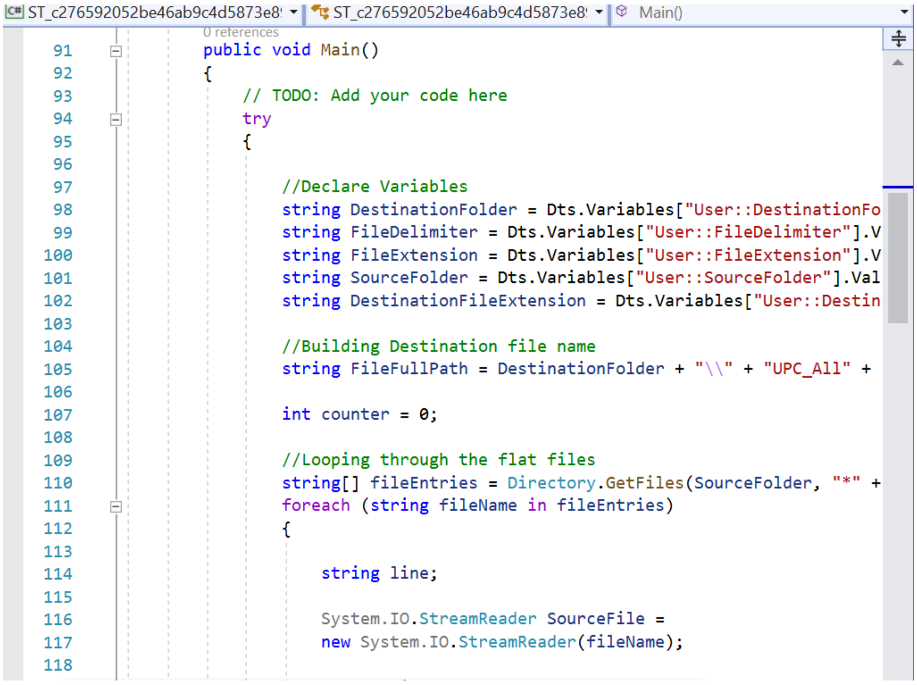
B) Import Demographic Information



Data Flow Task: Import Demo

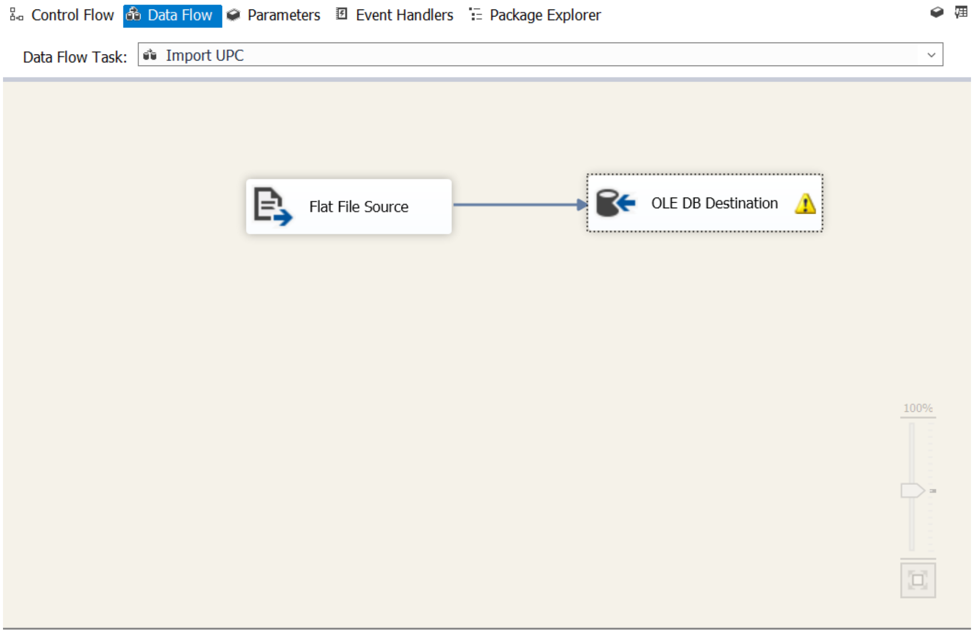
C) Import UPC file :

We used Script Task and C# coding to iterate all UPC files and combine them into a single table and finally load it to the staging table. The UPC file names were captured by script as category name



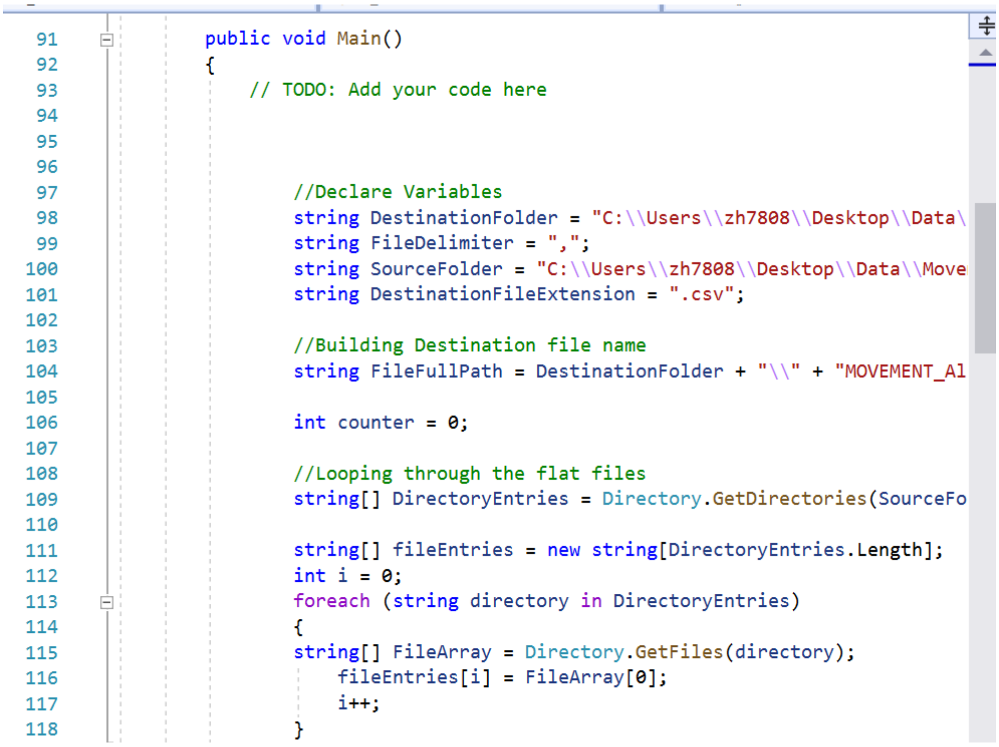
ST_c276592052be46ab9c4d5873e8!
ST_c276592052be46ab9c4d5873e8!
Main()
0 references

```
        public void Main()
        {
            // TODO: Add your code here
            try
            {

                //Declare Variables
                string DestinationFolder = Dts.Variables["User::DestinationFo
                string FileDelimiter = Dts.Variables["User::FileDelimiter"].V
                string FileExtension = Dts.Variables["User::FileExtension"].V
                string SourceFolder = Dts.Variables["User::SourceFolder"].Val
                string DestinationFileExtension = Dts.Variables["User::Destin

                //Building Destination file name
                string FileFullPath = DestinationFolder + "\\" + "UPC_All" +

                int counter = 0;

                //Looping through the flat files
                string[] fileEntries = Directory.GetFiles(SourceFolder, "*" +
                foreach (string fileName in fileEntries)
                {

                    string line;

                    System.IO.StreamReader SourceFile =
                    new System.IO.StreamReader(fileName);
```

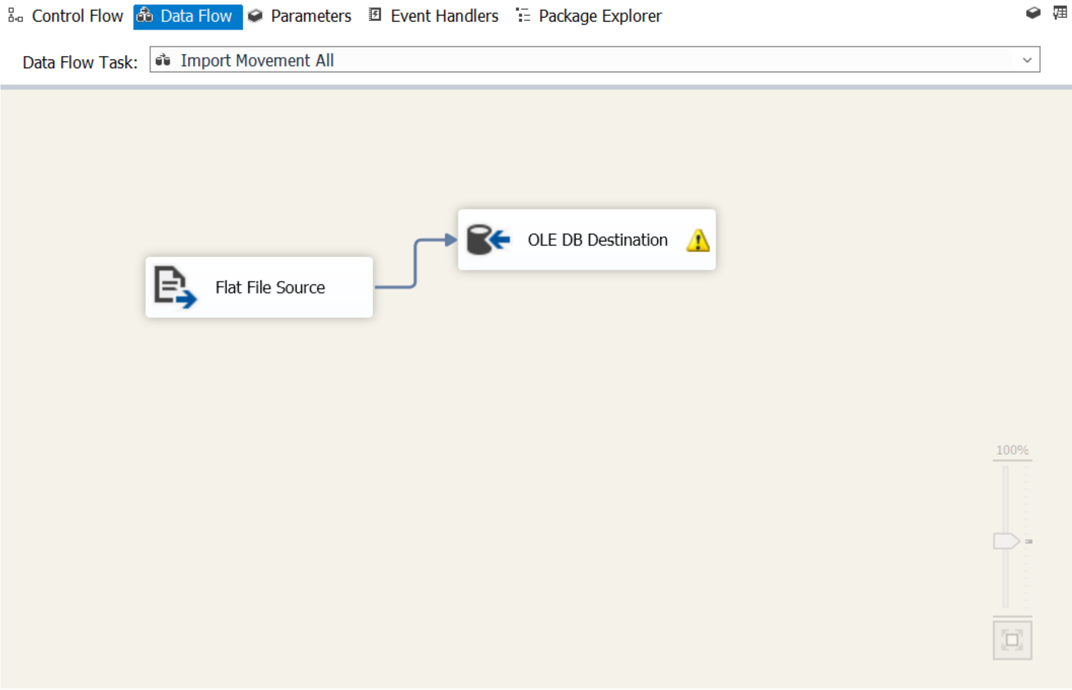


D) Import Movement File:

- We selected four categories as our movement data: beer, cigarette, dish detergent and frozen dinner.
- There are over 13 million rows from the above categories and there are over 165 million rows from all movement files.



```
public void Main()
{
    // TODO: Add your code here



        //Declare Variables
        string DestinationFolder = "C:\\Users\\zh7808\\Desktop\\Data\
        string FileDelimiter = ",";
        string SourceFolder = "C:\\Users\\zh7808\\Desktop\\Data\\Move
        string DestinationFileExtension = ".csv";

        //Building Destination file name
        string FileFullPath = DestinationFolder + "\\" + "MOVEMENT_Al

        int counter = 0;

        //Looping through the flat files
        string[] DirectoryEntries = Directory.GetDirectories(SourceFo

        string[] fileEntries = new string[DirectoryEntries.Length];
        int i = 0;
        foreach (string directory in DirectoryEntries)
        {
        string[] FileArray = Directory.GetFiles(directory);
            fileEntries[i] = FileArray[0];
            i++;
        }
```



Data Flow Task: Import Movement All

Data Transformation and Cleansing Rules

The transformation and data cleansing rules are as follows:

Demographic data transformation:

- Removed all the non-numeric data

Customer Count data transformation:

- We removed non-integer stores and negative stores for customer counts.
- We removed quotation mark for data fields and removed non-numeric dates for customer counts
- We stored this as Customer_final table

UPC data transformation:

- We used the right method to obtain the last 5 digits in UPC as a category identifier.

Movement table data transformation:

- The movement table had 165 million rows which is humongous. So we decided only a fraction of them had around 16 million rows and named it as **movement_new** table.
- The non-integer values were removed from the column Move and Qty
- The data types were converted while loading the data into the table
- Convert acronym categories to full-name categories
- Calculate the weekly sales for each category
- Calculate the weekly sales for each product
- Calculate the weekly growth rate for each product

Plan for Aggregation

Storing the data in the lowest granularity level would help the users move up and down any level of aggregation they want and extract the specific information for their report. However, it could impact the time and performance. So, we need to put ourselves in the shoes of a data architect and consider the granularity level we want to use for storing data.

A data aggregation would significantly improve the time and performance of the data warehousing. Before we dive in, we also need to consider our business requirement and understand what aggregation plan the best for us is. In our case, business question 4 requires us to calculate a standard deviation of 40 weeks. So, we precalculated the result for each week starting from week 41. For example, the standard deviation of week 41 will be based on data from week 1-40. Similarly, the standard deviation of week 42 will be based on data from week 2-41 and so on.

For other business questions, we have category sales and customer traffic which needs to be calculated at granular level. So, we keep the sales records and customer traffic at granular level in our fact table to fit our business requirement.

Aggregation in Fact_Store_Demo: For Fact_Store_Demo, we used the store and "EDUC" columns from source data "DEMO", then we used lookup function to connect with Dim_Store, getting the store_key. Finally, we get a store table with the rate of college students

Organization of Data Staging Area

The extracted data from the data source has been stored in 600_group5_staging area database. We selected only the required columns from the source excel files. This data from staging area will be further transformed for data marts. The screenshots of different tables in staging area are as follows:

Mapping definition describing the source to end table for all dimension and fact table.

SQL statements used for ETL

Create Data marts for DFF

```
Create table Dim_Store (
Store_key int NOT NULL IDENTITY(1,1),
StoreID int,
[Address] varchar(50),
ZipCode int,
IncomeTier varchar(50),
PRIMARY KEY (Store_key)
)
Go

Create table Dim_Category (
Category_key int NOT NULL IDENTITY(1,1),
CategoryID varchar(50),
CategoryName varchar(50),
PRIMARY KEY (Category_key)
)
Go

Create table Dim_Product (
Product_key int NOT NULL IDENTITY(1,1),
ProductID varchar(50),
ProductName varchar(50),
ProductDescription varchar(50),
CategoryID varchar(50),
CategoryName varchar(50)
PRIMARY KEY (Product_key)
)
```

```
Go

Create table Dim_StoreDemographic (
Zone_key int NOT NULL IDENTITY(1,1),
ZoneID int,
ProvertyPercentage float,
PRIMARY KEY (Zone_key)
)
Go

Create table Dim_Time (
WeekNumber int NOT NULL IDENTITY(1,1),
CalendarYear int,
CalendarQuarter int,
CalendarMonth int,
SpecialHoliday varchar(50),
StartDate date,
EndDate date,
PRIMARY KEY (WeekNumber)
)
Go

Create table Store_Demograhpics_Fact (
Store_key int,
PercentageofCollegeStudents float,
PRIMARY KEY (Store_key),
FOREIGN KEY ([Store_key]) REFERENCES [dbo].[Dim_Store] ([Store_key])
)
Go

Create table Customer_Count_Fact (
Store_key int,
Zone_key int,
WeekNumber int,
CustomerCounts float,
PRIMARY KEY (Store_key, Zone_key, WeekNumber),
FOREIGN KEY ([Store_key]) REFERENCES [dbo].[Dim_Store] ([Store_key]),
FOREIGN KEY ([Zone_key]) REFERENCES [dbo].[Dim_StoreDemographic] ([Zone_key]),
FOREIGN KEY ([WeekNumber]) REFERENCES [dbo].[Dim_Time] ([WeekNumber])
)
Go

Create table Category_Sales_Fact (
Store_key int,
Category_key int,
WeekNumber int,
DollarAmountSold float,
PRIMARY KEY (Store_key, Category_key, WeekNumber),
FOREIGN KEY ([Store_key]) REFERENCES [dbo].[Dim_Store] ([Store_key]),
FOREIGN KEY ([Category_key]) REFERENCES [dbo].[Dim_Category] ([Category_key]),
FOREIGN KEY ([WeekNumber]) REFERENCES [dbo].[Dim_Time] ([WeekNumber])
)
Go
```

```
Create table Product_Sales_Fact (
Product_key int,
WeekNumber int,
DollarAmountSold float,
LastWeekSaleGrowth float,
StandardDeviation40weeks float,
PRIMARY KEY (Product_key, WeekNumber),
FOREIGN KEY ([Product_key]) REFERENCES [dbo].[Dim_Product] ([Product_key]),
FOREIGN KEY ([WeekNumber]) REFERENCES [dbo].[Dim_Time] ([WeekNumber])
)
Go
```

Commands for Movement Data Transformation

```
BEGIN TRANSACTION TRANSAC6

DELETE

FROM [600_group5_staging_area].[dbo].[Movement_New]

WHERE ["MOVE"] LIKE '%[^0-9]%' OR ["QTY"] LIKE '%[^0-9]%'

COMMIT TRANSACTION TRANSAC6

BEGIN TRANSACTION Tran7

ALTER TABLE [600_group5_staging_area].[dbo].[Movement_New]

ADD SALES FLOAT

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [SALES] = CAST(["PRICE"] AS FLOAT)*CAST(["MOVE"] AS FLOAT)/CAST(["QTY"] AS INT)

FROM [600_group5_staging_area].[dbo].[Movement_New]

SELECT COUNT(*) FROM [600_group5_staging_area].[dbo].[Movement_New] WHERE [SALES] IS NULL

SELECT

 CAST(["PRICE"] AS FLOAT)*CAST(["MOVE"] AS FLOAT)/CAST(["QTY"] AS INT)

FROM [600_group5_staging_area].[dbo].[Movement_New]

COMMIT TRANSACTION Tran7
```

```
BEGIN TRANSACTION Tran8

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [CATEGORY] = RIGHT([CATEGORY],3)

SELECT * FROM [600_group5_staging_area].[dbo].[Movement_New]

COMMIT TRANSACTION Tran8

BEGIN TRANSACTION Tran9

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [CATEGORY] = 'DID'

WHERE [CATEGORY] = 'one'

SELECT [CATEGORY] FROM [600_group5_staging_area].[dbo].[Movement_New] GROUP BY [CATEGORY]

COMMIT TRANSACTION Tran9

CREATE TABLE Category_Dim(

CategoryID INT NOT NULL IDENTITY(1,1),

CategoryName varchar(50))

BEGIN TRANSACTION Tran9

ALTER TABLE [600_group5_staging_area].[dbo].[Movement_New]

ADD CATEGORY_NAME VARCHAR(50)

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [CATEGORY_NAME] = 'Beer'

WHERE [CATEGORY] = 'BER'

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [CATEGORY_NAME] = 'Cigarette'

WHERE [CATEGORY] = 'CIG'

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [CATEGORY_NAME] = 'Dish Detergent'
```

```
WHERE [CATEGORY] = 'DID'

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [CATEGORY_NAME] = 'Frozen Dish'

WHERE [CATEGORY] = 'FRD'

SELECT * FROM [600_group5_staging_area].[dbo].[Movement_New]

INSERT INTO Category_Dim ([CategoryName]) SELECT [CATEGORY] FROM
[600_group5_staging_area].[dbo].[Movement_New] GROUP BY [CATEGORY]

SELECT [CATEGORY],["WEEK"],["STORE"], SUM(CAST(["SALE"] AS FLOAT)) AS WEEK_SALES
FROM [600_group5_staging_area].[dbo].[Movement_New]

GROUP BY [CATEGORY], ["WEEK"], ["STORE"]

ORDER BY CAST(["WEEK"] AS INT) ASC

COMMIT TRANSACTION Tran9

BEGIN TRANSACTION Tran11

ALTER TABLE [600_group5_staging_area].[dbo].[Movement_New]

ADD ProductID VARCHAR(50)

UPDATE [600_group5_staging_area].[dbo].[Movement_New]

SET [ProductID] = RIGHT(["UPC"],5)

SELECT * FROM [600_group5_staging_area].[dbo].[Movement_New]

COMMIT TRANSACTION Tran11

BEGIN TRANSACTION Tran12

USE [600_group5_staging_area]

GO

CREATE TABLE Product_Sale(

ProductID VARCHAR(50),

WeekNumber INT,

DollarAmountSold FLOAT)

INSERT INTO Product_Sale (ProductID,WeekNumber, DollarAmountSold)
```

```
SELECT [ProductID],CAST(["WEEK"] AS INT), SUM(CAST(["SALE"] AS FLOAT)) AS
DollarAmountSold
FROM [600_group5_staging_area].[dbo].[Movement_New]
GROUP BY [ProductID],["WEEK"]
COMMIT TRANSACTION Tran12
BEGIN TRANSACTION Tran13
USE [600_group5_staging_area]
GO
CREATE TABLE Product_Sale_Temp(
ProductID VARCHAR(50),
WeekNumber INT,
DollarAmountSold FLOAT,
Growth_Rate FLOAT)
INSERT INTO Product_Sale_Temp(
ProductID ,
WeekNumber ,
DollarAmountSold ,
Growth_Rate )
SELECT s1.[ProductID],s1.[WeekNumber],s2.[DollarAmountSold],
CASE
WHEN s1.[DollarAmountSold] = s2.[DollarAmountSold] AND s1.[DollarAmountSold] = 0
THEN 0
WHEN s2.[DollarAmountSold] = 0 AND s1.[DollarAmountSold] <> 0
THEN 1
ELSE
s1.[DollarAmountSold] / s2.[DollarAmountSold]
END AS Growth_Rate
```

```
FROM [600_group5_staging_area].[dbo].[Product_Sale] AS s1

INNER JOIN [600_group5_staging_area].[dbo].[Product_Sale] AS s2

ON s1.[ProductID] = s2.[ProductID] AND CAST(s1.[WeekNumber] AS INT) =
CAST(s2.[WeekNumber] AS INT) + 1

COMMIT TRANSACTION Tran13
```

Commands for Customer Count Data Transformation

```
BEGIN TRANSACTION [Tran1]

DELETE

FROM [600_group5_staging_area].[dbo].[Customer Count]

WHERE ISNUMERIC(["STORE"]) = 0;

COMMIT TRANSACTION [Tran1];

BEGIN TRANSACTION [Tran2]

DELETE

FROM [600_group5_staging_area].[dbo].[Customer Count]

WHERE CAST(["STORE"] AS INT) <= 0 ;

SELECT ["STORE"]

FROM [600_group5_staging_area].[dbo].[Customer Count]

GROUP BY ["STORE"];

COMMIT TRANSACTION [Tran2];

BEGIN TRANSACTION Tran3

UPDATE [600_group5_staging_area].[dbo].[Customer Count]

SET ["DATE"] = REPLACE(["DATE"],'"','');

COMMIT Transaction Tran3;

BEGIN TRANSACTION Tran4

DELETE FROM [600_group5_staging_area].[dbo].[Customer Count]
```

```
WHERE ISNUMERIC(["DATE"]) = 0;

COMMIT TRANSACTION Tran4;

BEGIN TRANSACTION [Tran5]

DELETE FROM [600_group5_staging_area].[dbo].[Customer Count]

WHERE ["WEEK"]   like '%[^0-9]%'

DELETE FROM [600_group5_staging_area].[dbo].[Customer Count]

WHERE CAST(["WEEK"] AS INT) < 0 OR CAST(["WEEK"] AS INT) > 400;

SELECT * FROM [600_group5_staging_area].[dbo].[Customer Count]

WHERE ["WEEK"]  NOT like '%[^0-9]%'

COMMIT TRANSACTION [Tran5]

BEGIN TRANSACTION Tran10
```

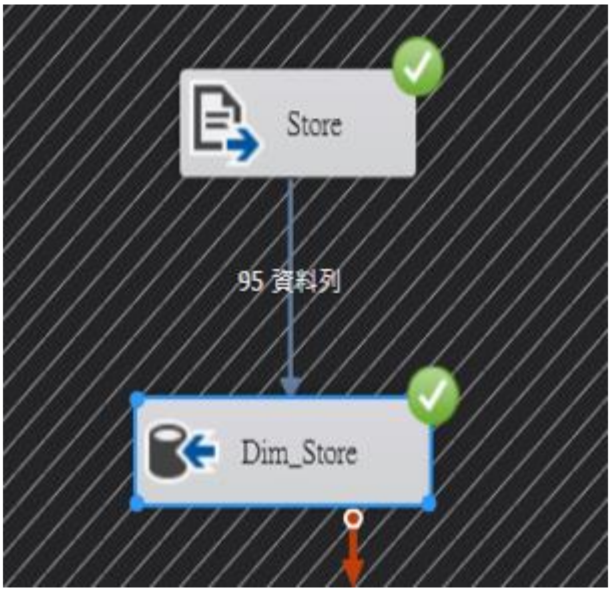
Procedures for all data extraction and loading

The main source of data for this project is from the 5 csv files namely customer count, demographics and 3 movement files (WTTI,WFSF,WLND) and store and week decode table. The data was directly loaded into data staging area of the warehouse and data cleansing operations were performed as explained above. The cleaned data was further transformed using different ETL processes as given below for the 4 final dimension tables - Dim_Store, Dim_Time, Dim_StoreDemo, Dim_Prod and # final fact tables - ____

ETL FOR DIMENSION TABLES

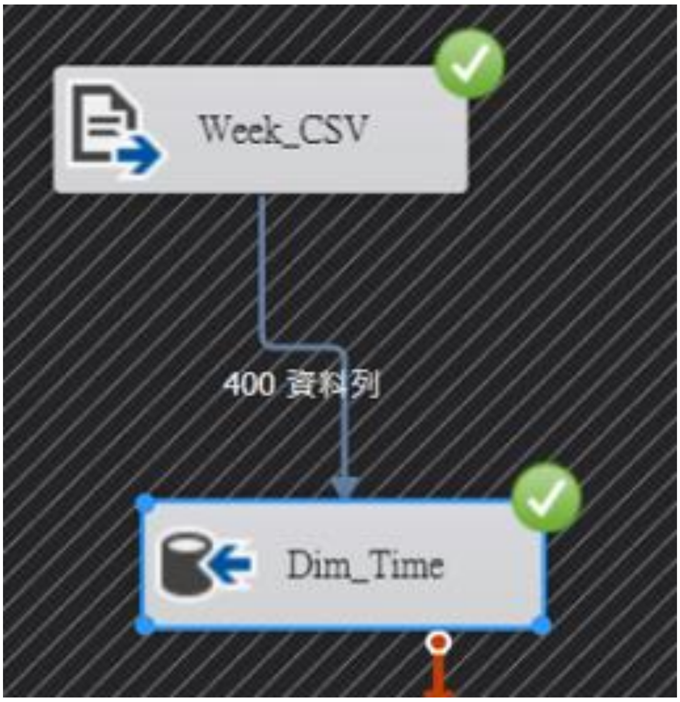
Dim_Store:

Source data for Dim_Store are the raw data from DFF's cookbook. It has already been cleaned. And we changed the data types of some columns, such as StoreID and ZipCode. The primary key, Store_key, is auto-increment key.



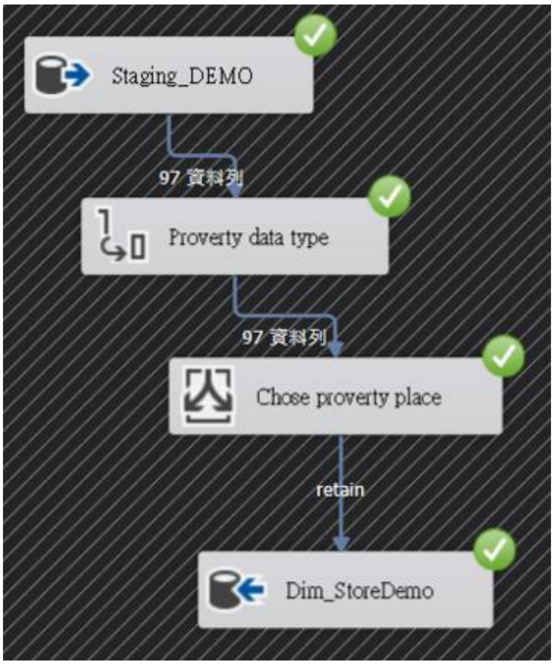
Dim_Time

Dim_Time data is similar with Dim_Store data. We get the data directly from DFF's cookbook. We divided data into three columns, Year, Quarter, and Month.



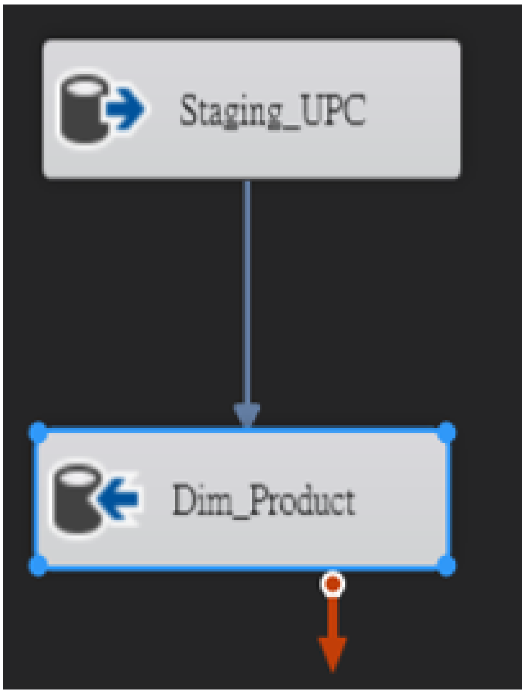
Dim_StoreDemographic

For Dim_StoreDemo, we load the data from the staging area. We first changed the data type of poverty rate into float type. For the next step, we selected the stores where the poverty rate in the places are higher than 10%. Finally, we loaded the specific columns into Dim_StoreDemo.



Dim_Product

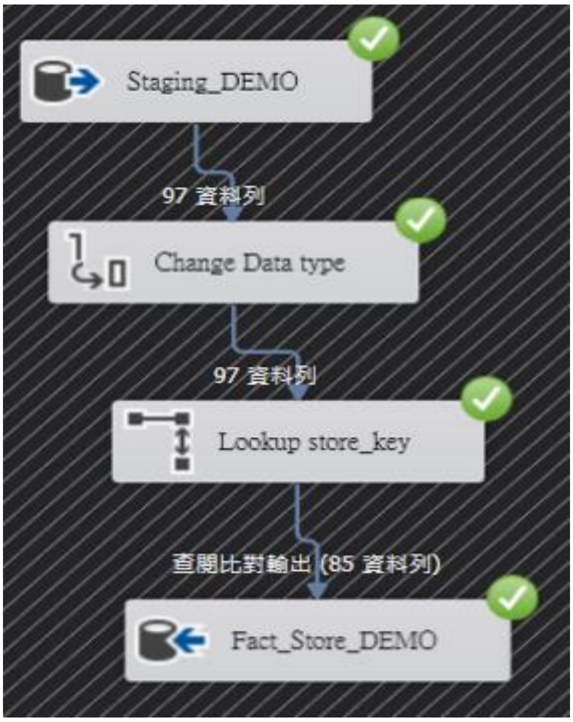
For Dim_Product, we load the data from the staging area.



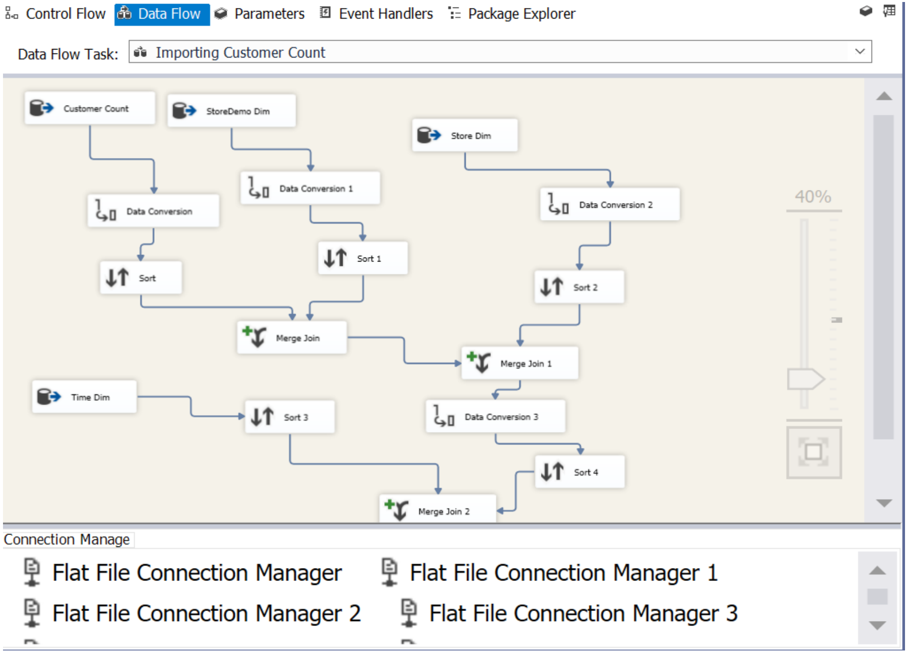
ETL for fact Tables

Fact_Store_Demo

For the first fact table, we first selected the Staging_Demo in the staging area, and we used a lookup function to connect with Dim_Store. Finally, we can get the fact table.



Fact_Customer_Count



Data Flow Task: Importing Customer Count

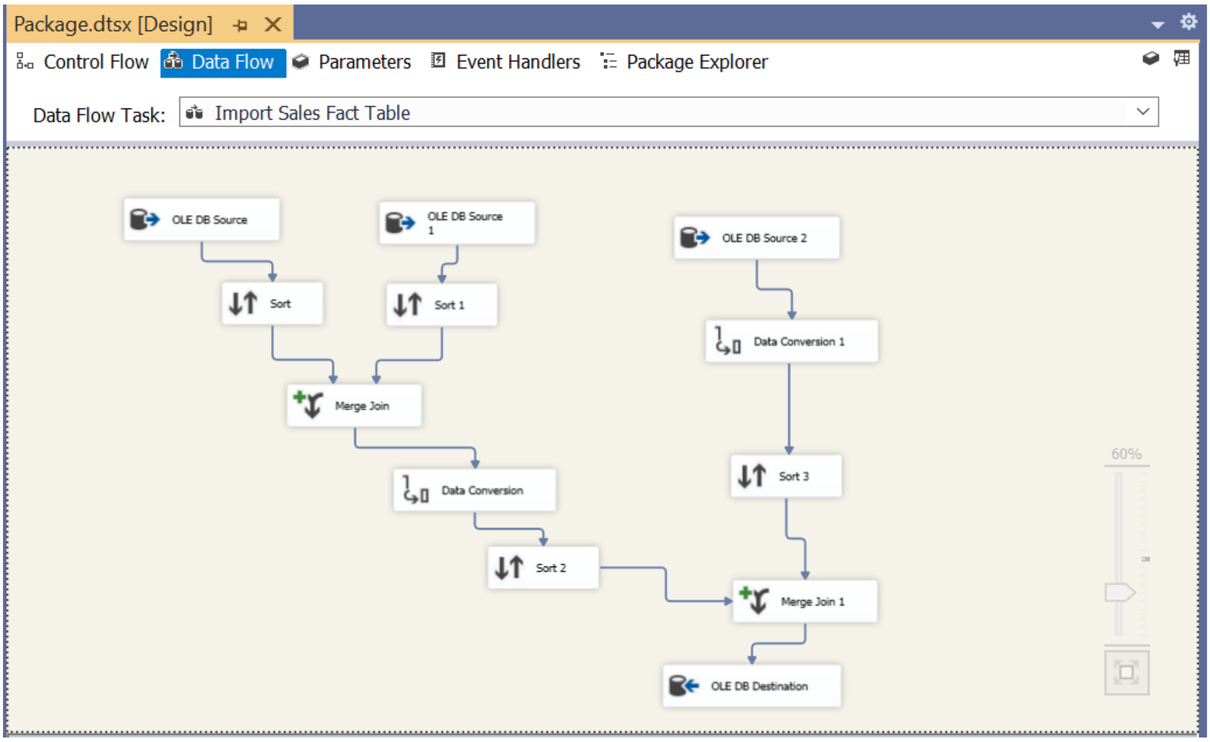
Flat File Connection Manager

Flat File Connection Manager 1

Flat File Connection Manager 2

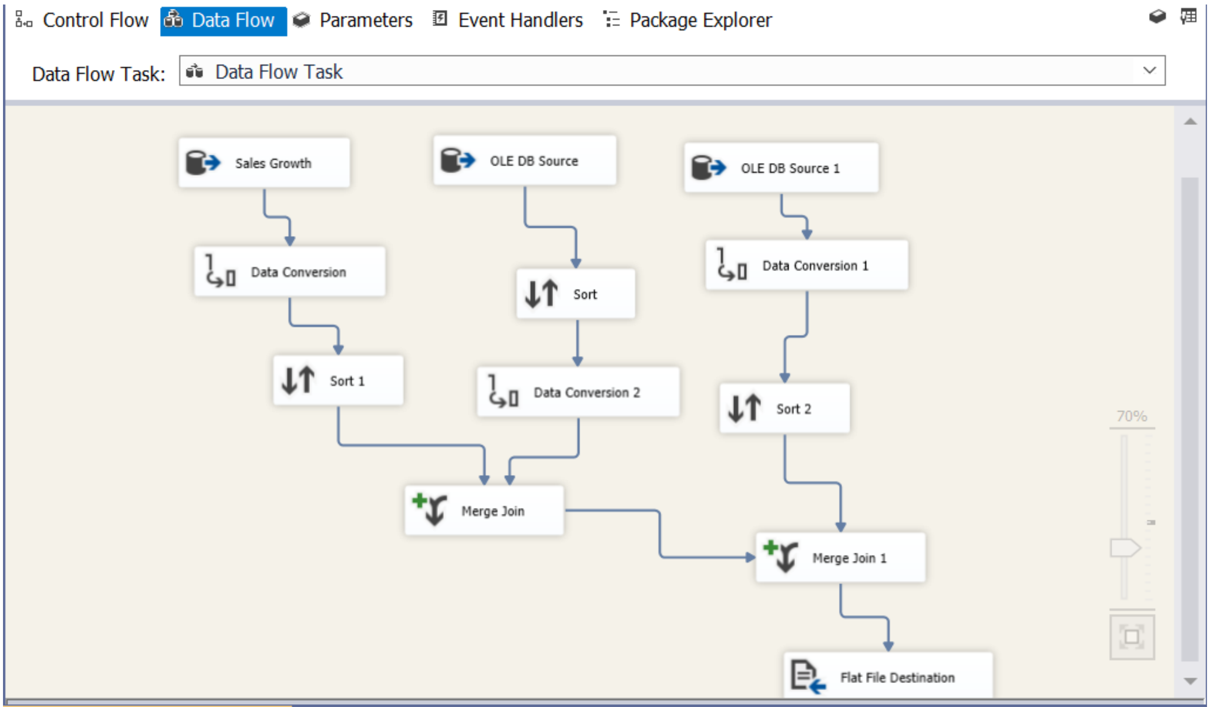
Flat File Connection Manager 3

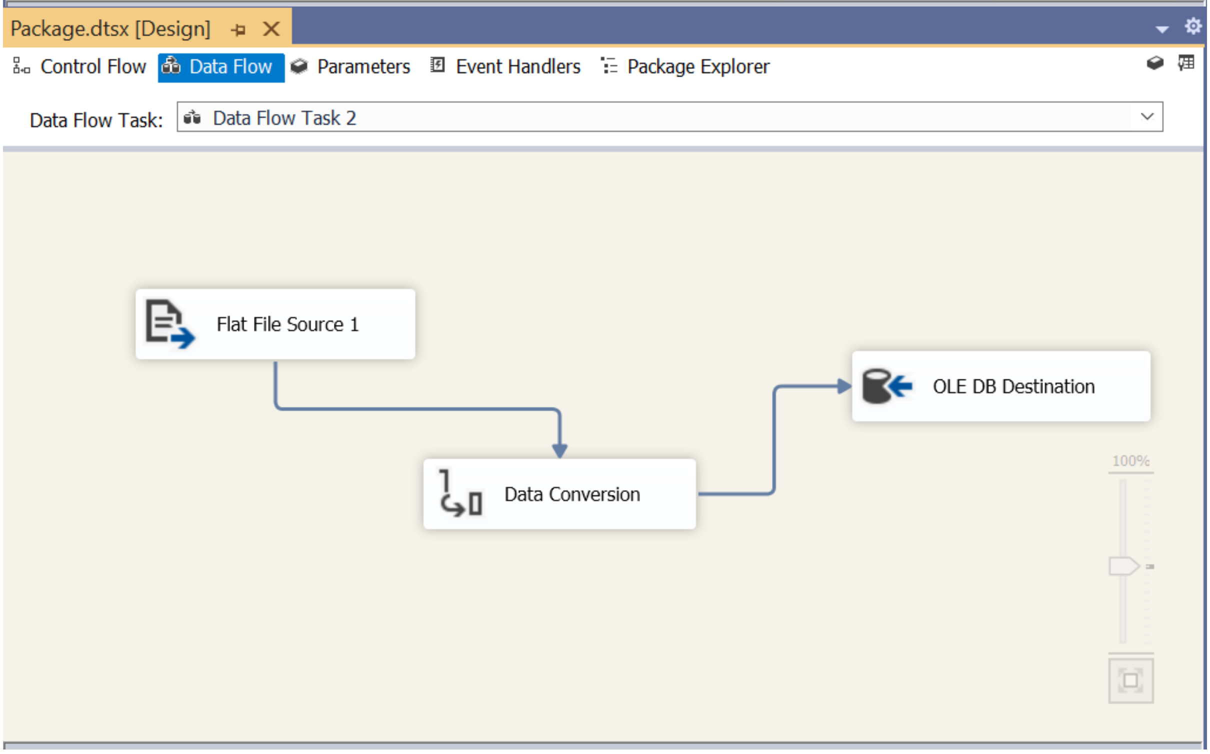
Fact_Category_Sales



Data Flow Task: Import Sales Fact Table

Product_Sale_Fact

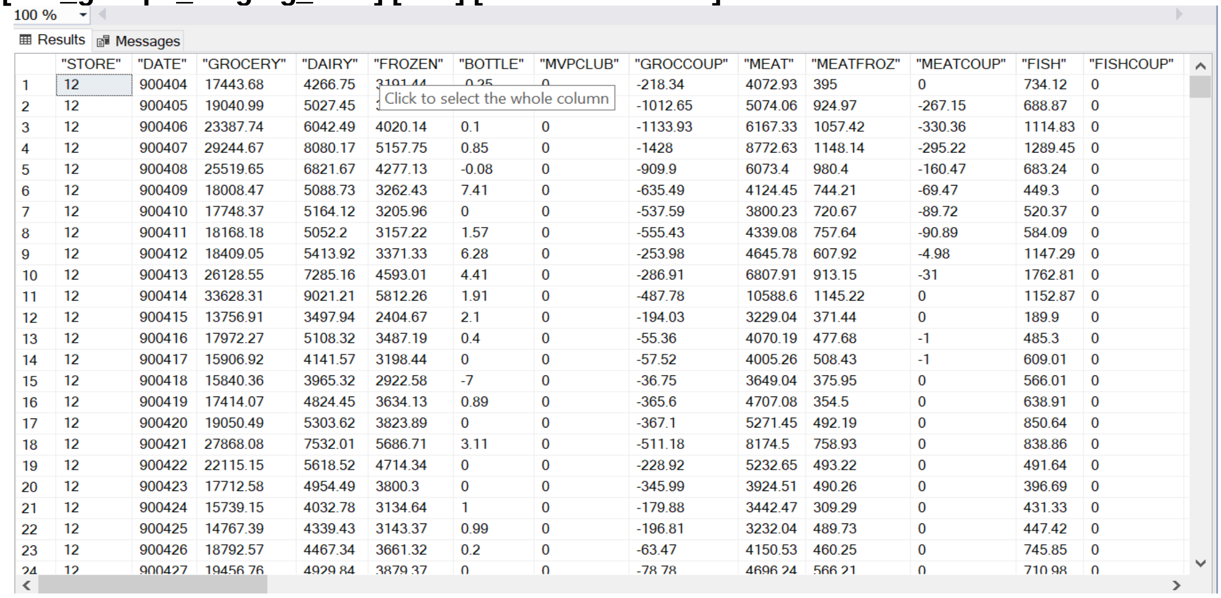




Comparison of Before-After Table Contents

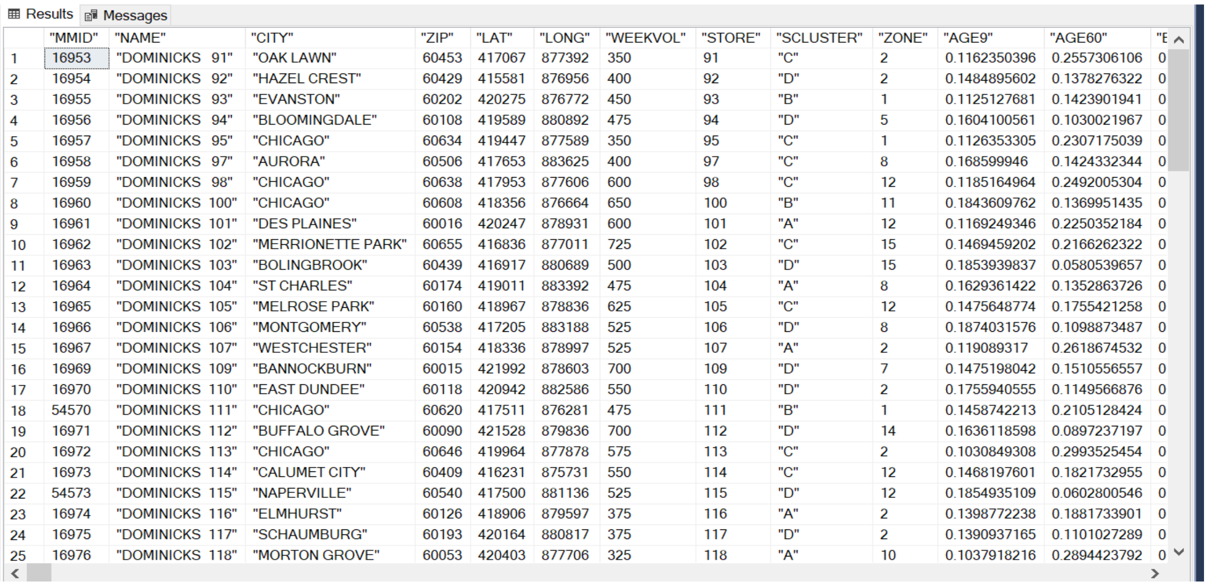
Before ETL Table screenshots

Customer Count
[600_group5_staging_area].[dbo].[Customer Count]



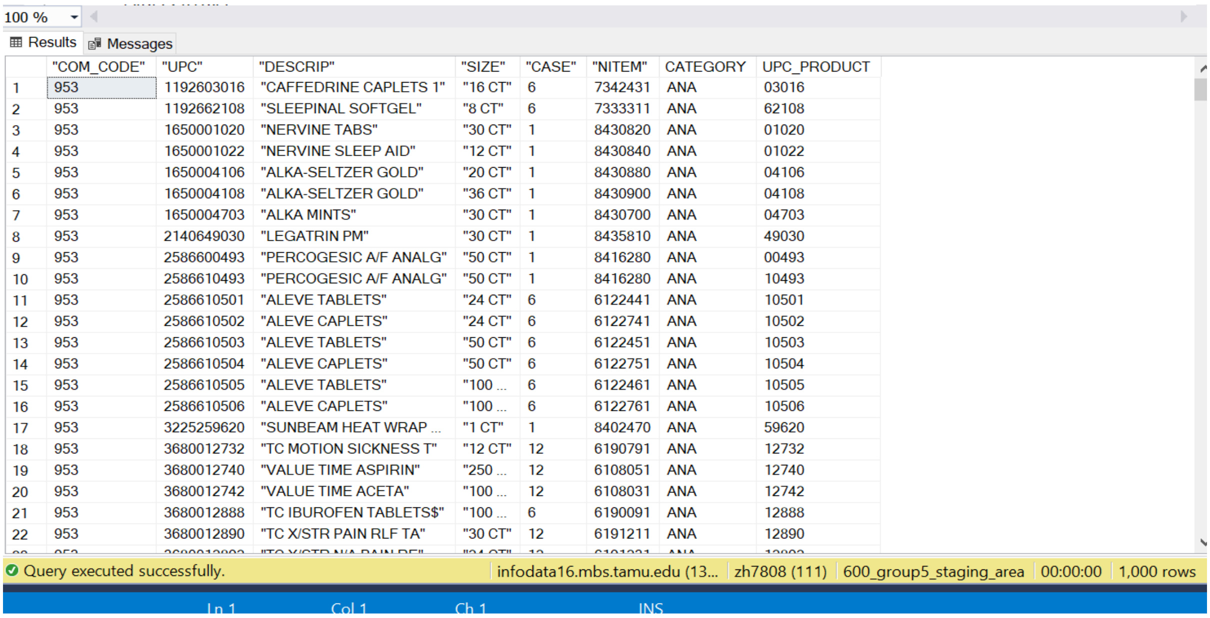
	"STORE"	"DATE"	"GROCERY"	"DAIRY"	"FROZEN"	"BOTTLE"	"MVPCLUB"	"GROCCOUP"	"MEAT"	"MEATFROZ"	"MEATCOUP"	"FISH"	"FISHCOUP"
1	12	900404	17443.68	4266.75	3191.44	0.25	0	-218.34	4072.93	395	0	734.12	0
2	12	900405	19040.99	5027.45	[illegible]	[illegible]	[illegible]	-1012.65	5074.06	924.97	-267.15	688.87	0
3	12	900406	23387.74	6042.49	4020.14	0.1	0	-1133.93	6167.33	1057.42	-330.36	1114.83	0
4	12	900407	29244.67	8080.17	5157.75	0.85	0	-1428	8772.63	1148.14	-295.22	1289.45	0
5	12	900408	25519.65	6821.67	4277.13	-0.08	0	-909.9	6073.4	980.4	-160.47	683.24	0
6	12	900409	18008.47	5088.73	3262.43	7.41	0	-635.49	4124.45	744.21	-69.47	449.3	0
7	12	900410	17748.37	5164.12	3205.96	0	0	-537.59	3800.23	720.67	-89.72	520.37	0
8	12	900411	18168.18	5052.2	3157.22	1.57	0	-555.43	4339.08	757.64	-90.89	584.09	0
9	12	900412	18409.05	5413.92	3371.33	6.28	0	-253.98	4645.78	607.92	-4.98	1147.29	0
10	12	900413	26128.55	7285.16	4593.01	4.41	0	-286.91	6807.91	913.15	-31	1762.81	0
11	12	900414	33628.31	9021.21	5812.26	1.91	0	-487.78	10588.6	1145.22	0	1152.87	0
12	12	900415	13756.91	3497.94	2404.67	2.1	0	-194.03	3229.04	371.44	0	189.9	0
13	12	900416	17972.27	5108.32	3487.19	0.4	0	-55.36	4070.19	477.68	-1	485.3	0
14	12	900417	15906.92	4141.57	3198.44	0	0	-57.52	4005.26	508.43	-1	609.01	0
15	12	900418	15840.36	3965.32	2922.58	-7	0	-36.75	3649.04	375.95	0	566.01	0
16	12	900419	17414.07	4824.45	3634.13	0.89	0	-365.6	4707.08	354.5	0	638.91	0
17	12	900420	19050.49	5303.62	3823.89	0	0	-367.1	5271.45	492.19	0	850.64	0
18	12	900421	27868.08	7532.01	5686.71	3.11	0	-511.18	8174.5	758.93	0	838.86	0
19	12	900422	22115.15	5618.52	4714.34	0	0	-228.92	5232.65	493.22	0	491.64	0
20	12	900423	17712.58	4954.49	3800.3	0	0	-345.99	3924.51	490.26	0	396.69	0
21	12	900424	15739.15	4032.78	3134.64	1	0	-179.88	3442.47	309.29	0	431.33	0
22	12	900425	14767.39	4339.43	3143.37	0.99	0	-196.81	3232.04	489.73	0	447.42	0
23	12	900426	18792.57	4467.34	3661.32	0.2	0	-63.47	4150.53	460.25	0	745.85	0
24	12	900427	19456.76	4929.84	3879.37	0	0	-78.78	4696.24	566.21	0	710.98	0

Demographic
[600_group5_staging_area].[dbo].[DEMO]



	"MMID"	"NAME"	"CITY"	"ZIP"	"LAT"	"LONG"	"WEEKVOL"	"STORE"	"SCLUSTER"	"ZONE"	"AGE9"	"AGE60"	"E
1	16953	"DOMINICKS 91"	"OAK LAWN"	60453	417067	877392	350	91	"C"	2	0.1162350396	0.2557306106	0
2	16954	"DOMINICKS 92"	"HAZEL CREST"	60429	415581	876956	400	92	"D"	2	0.1484895602	0.1378276322	0
3	16955	"DOMINICKS 93"	"EVANSTON"	60202	420275	876772	450	93	"B"	1	0.1125127681	0.1423901941	0
4	16956	"DOMINICKS 94"	"BLOOMINGDALE"	60108	419589	880892	475	94	"D"	5	0.1604100561	0.1030021967	0
5	16957	"DOMINICKS 95"	"CHICAGO"	60634	419447	877589	350	95	"C"	1	0.1126353305	0.2307175039	0
6	16958	"DOMINICKS 97"	"AURORA"	60506	417653	883625	400	97	"C"	8	0.168599946	0.1424332344	0
7	16959	"DOMINICKS 98"	"CHICAGO"	60638	417953	877606	600	98	"C"	12	0.1185164964	0.2492005304	0
8	16960	"DOMINICKS 100"	"CHICAGO"	60608	418356	876664	650	100	"B"	11	0.1843609762	0.1369951435	0
9	16961	"DOMINICKS 101"	"DES PLAINES"	60016	420247	878931	600	101	"A"	12	0.1169249346	0.2250352184	0
10	16962	"DOMINICKS 102"	"MERRIONETTE PARK"	60655	416836	877011	725	102	"C"	15	0.1469459202	0.2166262322	0
11	16963	"DOMINICKS 103"	"BOLINGBROOK"	60439	416917	880689	500	103	"D"	15	0.1853939837	0.0580539657	0
12	16964	"DOMINICKS 104"	"ST CHARLES"	60174	419011	883392	475	104	"A"	8	0.1629361422	0.1352863726	0
13	16965	"DOMINICKS 105"	"MELROSE PARK"	60160	418967	878836	625	105	"C"	12	0.1475648774	0.1755421258	0
14	16966	"DOMINICKS 106"	"MONTGOMERY"	60538	417205	883188	525	106	"D"	8	0.1874031576	0.1098873487	0
15	16967	"DOMINICKS 107"	"WESTCHESTER"	60154	418336	878997	525	107	"A"	2	0.119089317	0.2618674532	0
16	16969	"DOMINICKS 109"	"BANNOCKBURN"	60015	421992	878603	700	109	"D"	7	0.1475198042	0.1510556557	0
17	16970	"DOMINICKS 110"	"EAST DUNDEE"	60118	420942	882586	550	110	"D"	2	0.1755940555	0.1149566876	0
18	54570	"DOMINICKS 111"	"CHICAGO"	60620	417511	876281	475	111	"B"	1	0.1458742213	0.2105128424	0
19	16971	"DOMINICKS 112"	"BUFFALO GROVE"	60090	421528	879836	700	112	"D"	14	0.1636118598	0.0897237197	0
20	16972	"DOMINICKS 113"	"CHICAGO"	60646	419964	877878	575	113	"C"	2	0.1030849308	0.2993525454	0
21	16973	"DOMINICKS 114"	"CALUMET CITY"	60409	416231	875731	550	114	"C"	12	0.1468197601	0.1821732955	0
22	54573	"DOMINICKS 115"	"NAPERVILLE"	60540	417500	881136	525	115	"D"	12	0.1854935109	0.0602800546	0
23	16974	"DOMINICKS 116"	"ELMHURST"	60126	418906	879597	375	116	"A"	2	0.1398772238	0.1881733901	0
24	16975	"DOMINICKS 117"	"SCHAUMBURG"	60193	420164	880817	375	117	"D"	2	0.1390937165	0.1101027289	0
25	16976	"DOMINICKS 118"	"MORTON GROVE"	60053	420403	877706	325	118	"A"	10	0.1037918216	0.2894423792	0

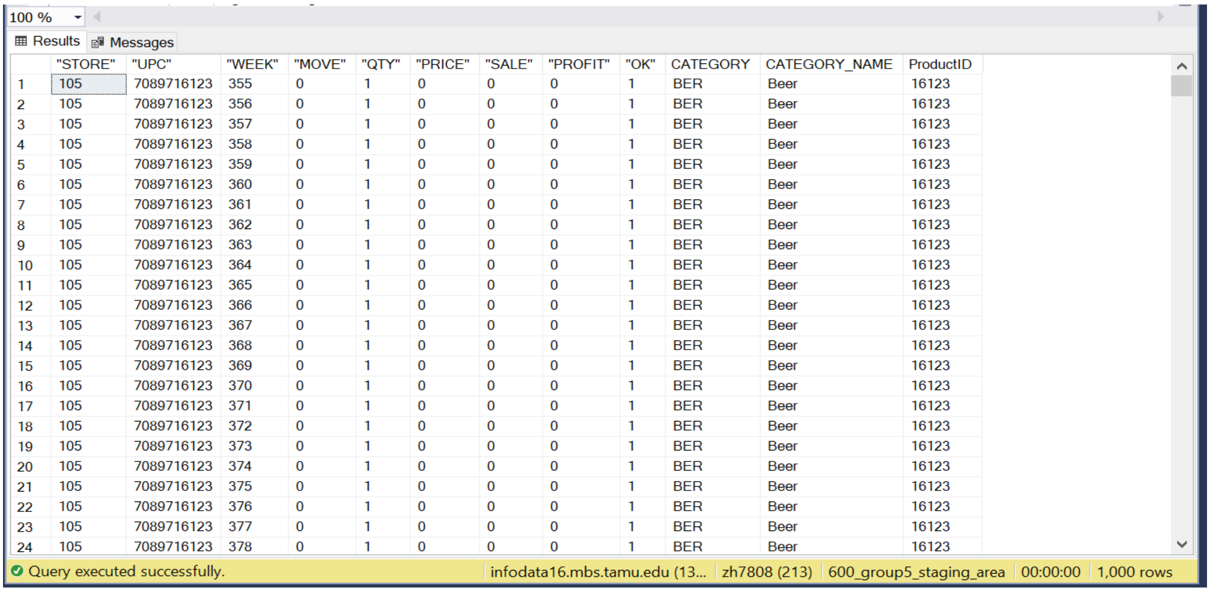
UPC Table
[600_group5_staging_area].[dbo].[UPC]



	"COM_CODE"	"UPC"	"DESCRIP"	"SIZE"	"CASE"	"NITEM"	CATEGORY	UPC_PRODUCT
1	953	1192603016	"CAFFEDRINE CAPLETS 1"	"16 CT"	6	7342431	ANA	03016
2	953	1192662108	"SLEEPINAL SOFTGEL"	"8 CT"	6	7333311	ANA	62108
3	953	1650001020	"NERVINE TABS"	"30 CT"	1	8430820	ANA	01020
4	953	1650001022	"NERVINE SLEEP AID"	"12 CT"	1	8430840	ANA	01022
5	953	1650004106	"ALKA-SELTZER GOLD"	"20 CT"	1	8430880	ANA	04106
6	953	1650004108	"ALKA-SELTZER GOLD"	"36 CT"	1	8430900	ANA	04108
7	953	1650004703	"ALKA MINTS"	"30 CT"	1	8430700	ANA	04703
8	953	2140649030	"LEGATRIN PM"	"30 CT"	1	8435810	ANA	49030
9	953	2586600493	"PERCOGESIC A/F ANALG"	"50 CT"	1	8416280	ANA	00493
10	953	2586610493	"PERCOGESIC A/F ANALG"	"50 CT"	1	8416280	ANA	10493
11	953	2586610501	"ALEVE TABLETS"	"24 CT"	6	6122441	ANA	10501
12	953	2586610502	"ALEVE CAPLETS"	"24 CT"	6	6122741	ANA	10502
13	953	2586610503	"ALEVE TABLETS"	"50 CT"	6	6122451	ANA	10503
14	953	2586610504	"ALEVE CAPLETS"	"50 CT"	6	6122751	ANA	10504
15	953	2586610505	"ALEVE TABLETS"	"100 ...	6	6122461	ANA	10505
16	953	2586610506	"ALEVE CAPLETS"	"100 ...	6	6122761	ANA	10506
17	953	3225259620	"SUNBEAM HEAT WRAP ...	"1 CT"	1	8402470	ANA	59620
18	953	3680012732	"TC MOTION SICKNESS T"	"12 CT"	12	6190791	ANA	12732
19	953	3680012740	"VALUE TIME ASPIRIN"	"250 ...	12	6108051	ANA	12740
20	953	3680012742	"VALUE TIME ACETA"	"100 ...	12	6108031	ANA	12742
21	953	3680012888	"TC IBUROFEN TABLETS$"	"100 ...	6	6190091	ANA	12888
22	953	3680012890	"TC X/STR PAIN RLF TA"	"30 CT"	12	6191211	ANA	12890

Query executed successfully. | infodata16.mbs.tamu.edu (13... | zh7808 (111) | 600_group5_staging_area | 00:00:00 | 1,000 rows

Movement File
[600_group5_staging_area].[dbo].[Movement_New]



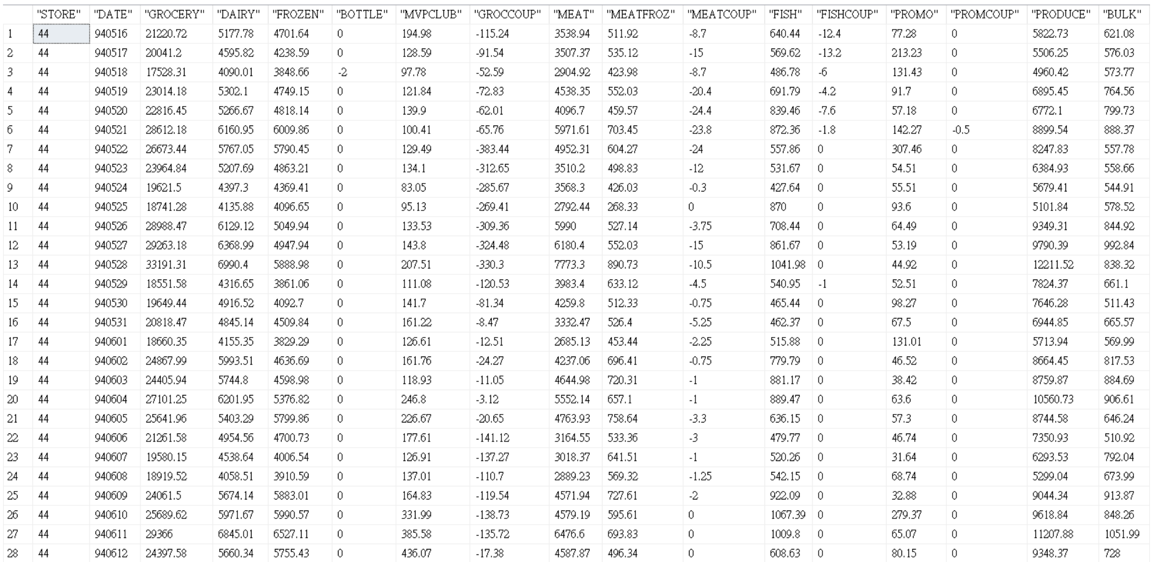
	"STORE"	"UPC"	"WEEK"	"MOVE"	"QTY"	"PRICE"	"SALE"	"PROFIT"	"OK"	CATEGORY	CATEGORY_NAME	ProductID
1	105	7089716123	355	0	1	0	0	0	1	BER	Beer	16123
2	105	7089716123	356	0	1	0	0	0	1	BER	Beer	16123
3	105	7089716123	357	0	1	0	0	0	1	BER	Beer	16123
4	105	7089716123	358	0	1	0	0	0	1	BER	Beer	16123
5	105	7089716123	359	0	1	0	0	0	1	BER	Beer	16123
6	105	7089716123	360	0	1	0	0	0	1	BER	Beer	16123
7	105	7089716123	361	0	1	0	0	0	1	BER	Beer	16123
8	105	7089716123	362	0	1	0	0	0	1	BER	Beer	16123
9	105	7089716123	363	0	1	0	0	0	1	BER	Beer	16123
10	105	7089716123	364	0	1	0	0	0	1	BER	Beer	16123
11	105	7089716123	365	0	1	0	0	0	1	BER	Beer	16123
12	105	7089716123	366	0	1	0	0	0	1	BER	Beer	16123
13	105	7089716123	367	0	1	0	0	0	1	BER	Beer	16123
14	105	7089716123	368	0	1	0	0	0	1	BER	Beer	16123
15	105	7089716123	369	0	1	0	0	0	1	BER	Beer	16123
16	105	7089716123	370	0	1	0	0	0	1	BER	Beer	16123
17	105	7089716123	371	0	1	0	0	0	1	BER	Beer	16123
18	105	7089716123	372	0	1	0	0	0	1	BER	Beer	16123
19	105	7089716123	373	0	1	0	0	0	1	BER	Beer	16123
20	105	7089716123	374	0	1	0	0	0	1	BER	Beer	16123
21	105	7089716123	375	0	1	0	0	0	1	BER	Beer	16123
22	105	7089716123	376	0	1	0	0	0	1	BER	Beer	16123
23	105	7089716123	377	0	1	0	0	0	1	BER	Beer	16123
24	105	7089716123	378	0	1	0	0	0	1	BER	Beer	16123

Query executed successfully. | infodata16.mbs.tamu.edu (13... | zh7808 (213) | 600_group5_staging_area | 00:00:00 | 1,000 rows

Final Staging and Data Mart Table Screenshot

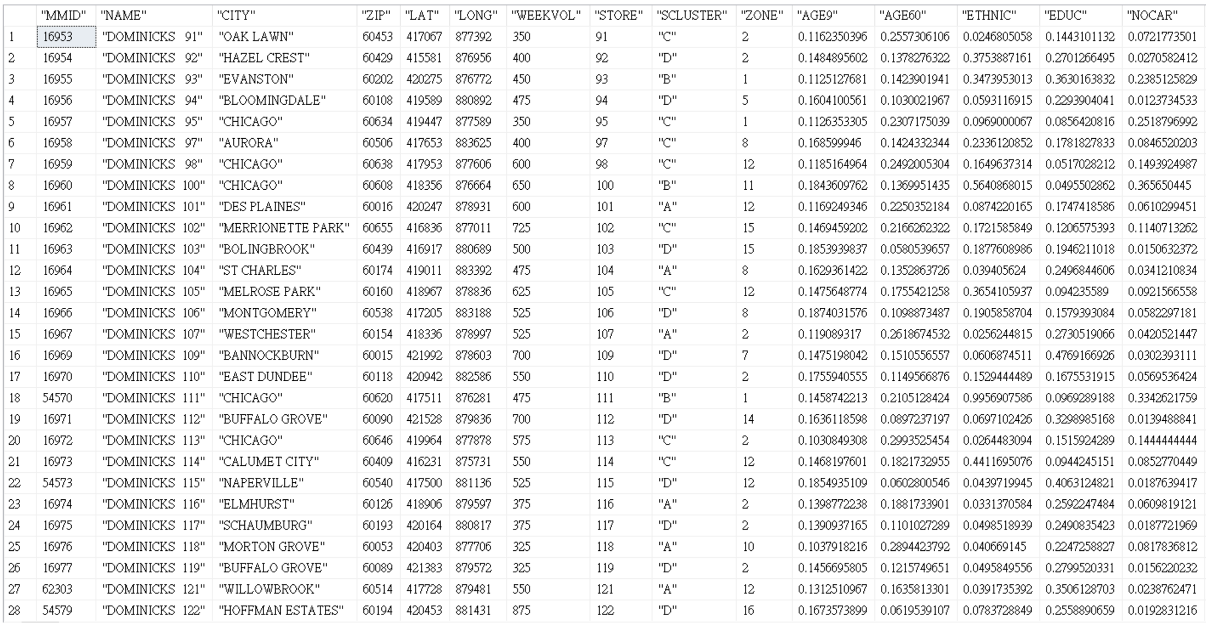
STAGING TABLES

a) Table: [600_group5_staging_area].[dbo].[Customer_Final]



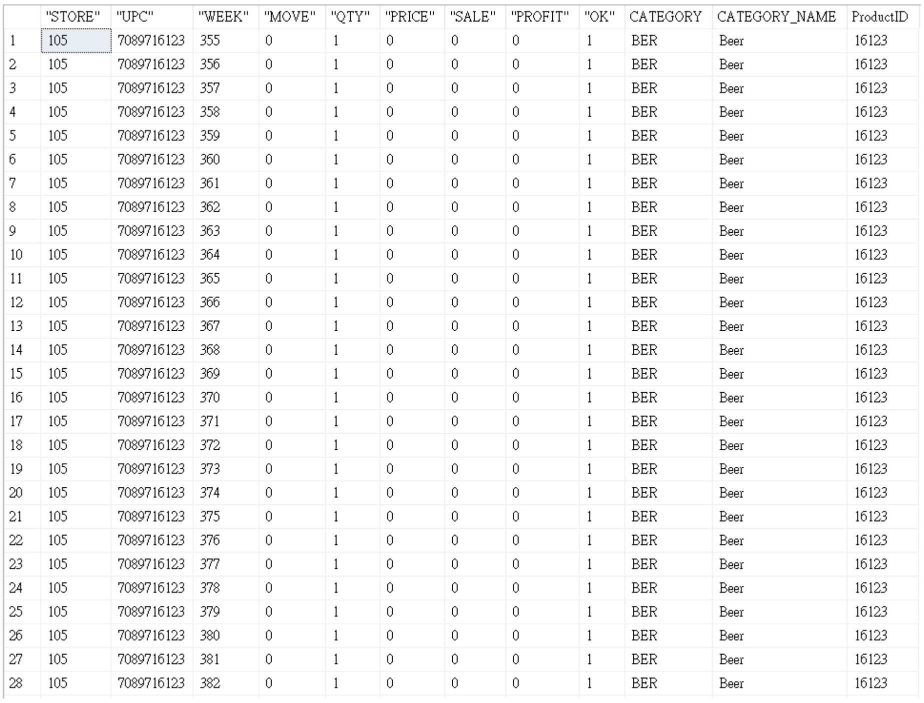
	"STORE"	"DATE"	"GROCERY"	"DAIRY"	"FROZEN"	"BOTTLE"	"MVPCLUB"	"GROCCOUP"	"MEAT"	"MEATFROZ"	"MEATCOUP"	"FISH"	"FISHCOUP"	"PROMO"	"PROMCOUP"	"PRODUCE"	"BULK"
1	44	940516	21220.72	5177.78	4701.64	0	194.98	-115.24	3538.94	511.92	-8.7	640.44	-12.4	77.28	0	5822.73	621.08
2	44	940517	20041.2	4595.82	4238.59	0	128.59	-91.54	3507.37	535.12	-15	569.62	-13.2	213.23	0	5506.25	576.03
3	44	940518	17528.31	4090.01	3848.66	-2	97.78	-52.59	2904.92	423.98	-8.7	486.78	-6	131.43	0	4960.42	573.77
4	44	940519	23014.18	5302.1	4749.15	0	121.84	-72.83	4538.35	552.03	-20.4	691.79	-4.2	91.7	0	6895.45	764.56
5	44	940520	22816.45	5266.67	4818.14	0	139.9	-62.01	4096.7	459.57	-24.4	839.46	-7.6	57.18	0	6772.1	799.73
6	44	940521	28612.18	6160.95	6009.86	0	100.41	-65.76	5971.61	703.45	-23.8	872.36	-1.8	142.27	-0.5	8899.54	888.37
7	44	940522	26673.44	5767.05	5790.45	0	129.49	-383.44	4952.31	604.27	-24	557.86	0	307.46	0	8247.83	557.78
8	44	940523	23964.84	5207.69	4863.21	0	134.1	-312.65	3510.2	498.83	-12	531.67	0	54.51	0	6384.93	558.66
9	44	940524	19621.5	4397.3	4369.41	0	83.05	-285.67	3568.3	426.03	-0.3	427.64	0	55.51	0	5679.41	544.91
10	44	940525	18741.28	4135.88	4096.65	0	95.13	-269.41	2792.44	268.33	0	870	0	93.6	0	5101.84	578.52
11	44	940526	28988.47	6129.12	5049.94	0	133.53	-309.36	5990	527.14	-3.75	708.44	0	64.49	0	9349.31	844.92
12	44	940527	29263.18	6368.99	4947.94	0	143.8	-324.48	6180.4	552.03	-15	861.67	0	53.19	0	9790.39	992.84
13	44	940528	33191.31	6990.4	5888.98	0	207.51	-330.3	7773.3	890.73	-10.5	1041.98	0	44.92	0	12211.52	838.32
14	44	940529	18551.58	4316.65	3861.06	0	111.08	-120.53	3983.4	633.12	-4.5	540.95	-1	52.51	0	7824.37	661.1
15	44	940530	19649.44	4916.52	4092.7	0	141.7	-81.34	4259.8	512.33	-0.75	465.44	0	98.27	0	7646.28	511.43
16	44	940531	20818.47	4845.14	4509.84	0	161.22	-8.47	3332.47	526.4	-5.25	462.37	0	67.5	0	6944.85	665.57
17	44	940601	18660.35	4155.35	3829.29	0	126.61	-12.51	2685.13	453.44	-2.25	515.88	0	131.01	0	5713.94	569.99
18	44	940602	24867.99	5993.51	4636.69	0	161.76	-24.27	4237.06	696.41	-0.75	779.79	0	46.52	0	8664.45	817.53
19	44	940603	24405.94	5744.8	4598.98	0	118.93	-11.05	4644.98	720.31	-1	881.17	0	38.42	0	8759.87	884.69
20	44	940604	27101.25	6201.95	5376.82	0	246.8	-3.12	5552.14	657.1	-1	889.47	0	63.6	0	10560.73	906.61
21	44	940605	25641.96	5403.29	5799.86	0	226.67	-20.65	4763.93	758.64	-3.3	636.15	0	57.3	0	8744.58	646.24
22	44	940606	21261.58	4954.56	4700.73	0	177.61	-141.12	3164.55	533.36	-3	479.77	0	46.74	0	7350.93	510.92
23	44	940607	19580.15	4538.64	4006.54	0	126.91	-137.27	3018.37	641.51	-1	520.26	0	31.64	0	6293.53	792.04
24	44	940608	18919.52	4058.51	3910.59	0	137.01	-110.7	2889.23	569.32	-1.25	542.15	0	68.74	0	5299.04	673.99
25	44	940609	24061.5	5674.14	5883.01	0	164.83	-119.54	4571.94	727.61	-2	922.09	0	32.88	0	9044.34	913.87
26	44	940610	25689.62	5971.67	5990.57	0	331.99	-138.73	4579.19	595.61	0	1067.39	0	279.37	0	9618.84	848.26
27	44	940611	29366	6845.01	6527.11	0	385.58	-135.72	6476.6	693.83	0	1009.8	0	65.07	0	11207.88	1051.99
28	44	940612	24397.58	5660.34	5755.43	0	436.07	-17.38	4587.87	496.34	0	608.63	0	80.15	0	9348.37	728

b) Table: [600_group5_staging_area].[dbo].[DEMO]



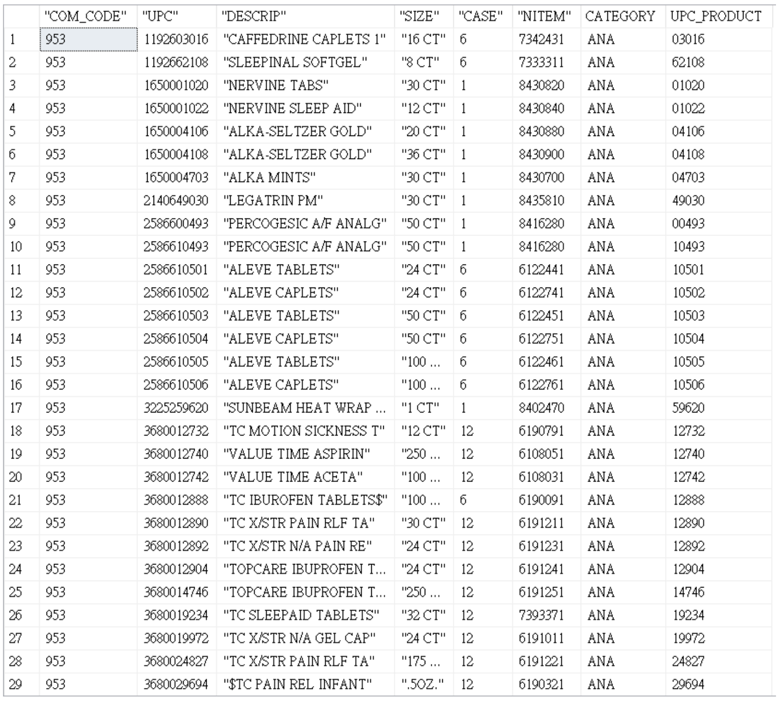
	"MMID"	"NAME"	"CITY"	"ZIP"	"LAT"	"LONG"	"WEEKVOL"	"STORE"	"SCLUSTER"	"ZONE"	"AGE9"	"AGE60"	"ETHNIC"	"EDUC"	"NOCAR"
1	16953	"DOMINICKS 91"	"OAK LAWN"	60453	417067	877392	350	91	"C"	2	0.1162350396	0.2557306106	0.0246805058	0.1443101132	0.0721773501
2	16954	"DOMINICKS 92"	"HAZEL CREST"	60429	415581	876956	400	92	"D"	2	0.1484895602	0.1378276322	0.3753887161	0.2701266495	0.0270582412
3	16955	"DOMINICKS 93"	"EVANSTON"	60202	420275	876772	450	93	"B"	1	0.1125127681	0.1423901941	0.3473953013	0.3630163832	0.2385125829
4	16956	"DOMINICKS 94"	"BLOOMINGDALE"	60108	419589	880892	475	94	"D"	5	0.1604100561	0.1030021967	0.0593116915	0.2293904041	0.0123734533
5	16957	"DOMINICKS 95"	"CHICAGO"	60634	419447	877589	350	95	"C"	1	0.1126353305	0.2307175039	0.0969000067	0.0856420816	0.2518796992
6	16958	"DOMINICKS 97"	"AURORA"	60506	417653	883625	400	97	"C"	8	0.168599946	0.1424332344	0.2336120852	0.1781827833	0.0846520203
7	16959	"DOMINICKS 98"	"CHICAGO"	60638	417953	877606	600	98	"C"	12	0.1185164964	0.2492005304	0.1649637314	0.0517028212	0.1493924987
8	16960	"DOMINICKS 100"	"CHICAGO"	60608	418356	876664	650	100	"B"	11	0.1843609762	0.1369951435	0.5640868015	0.0495502862	0.365650445
9	16961	"DOMINICKS 101"	"DES PLAINES"	60016	420247	878931	600	101	"A"	12	0.1169249346	0.2250352184	0.0874220165	0.1747418586	0.0610299451
10	16962	"DOMINICKS 102"	"MERRIONETTE PARK"	60655	416836	877011	725	102	"C"	15	0.1469459202	0.2166262322	0.1721585849	0.1206575393	0.1140713262
11	16963	"DOMINICKS 103"	"BOLINGBROOK"	60439	416917	880689	500	103	"D"	15	0.1853939837	0.0580539657	0.1877608986	0.1946211018	0.0150632372
12	16964	"DOMINICKS 104"	"ST CHARLES"	60174	419011	883392	475	104	"A"	8	0.1629361422	0.1352863726	0.039405624	0.2496844606	0.0341210834
13	16965	"DOMINICKS 105"	"MELROSE PARK"	60160	418967	878836	625	105	"C"	12	0.1475648774	0.1755421258	0.3654105937	0.094235589	0.0921566558
14	16966	"DOMINICKS 106"	"MONTGOMERY"	60538	417205	883188	525	106	"D"	8	0.1874031576	0.1098873487	0.1905858704	0.1579393084	0.0582297181
15	16967	"DOMINICKS 107"	"WESTCHESTER"	60154	418336	878997	525	107	"A"	2	0.119089317	0.2618674532	0.0256244815	0.2730519066	0.0420521447
16	16969	"DOMINICKS 109"	"BANNOCKBURN"	60015	421992	878603	700	109	"D"	7	0.1475198042	0.1510556557	0.0606874511	0.4769166926	0.0302393111
17	16970	"DOMINICKS 110"	"EAST DUNDEE"	60118	420942	882586	550	110	"D"	2	0.1755940555	0.1149566876	0.1529444489	0.1675531915	0.0569536424
18	54570	"DOMINICKS 111"	"CHICAGO"	60620	417511	876281	475	111	"B"	1	0.1458742213	0.2105128424	0.9956907586	0.0969289188	0.3342621759
19	16971	"DOMINICKS 112"	"BUFFALO GROVE"	60090	421528	879836	700	112	"D"	14	0.1636118598	0.0897237197	0.0697102426	0.3298985168	0.0139488841
20	16972	"DOMINICKS 113"	"CHICAGO"	60646	419964	877878	575	113	"C"	2	0.1030849308	0.2993525454	0.0264483094	0.1515924289	0.1444444444
21	16973	"DOMINICKS 114"	"CALUMET CITY"	60409	416231	875731	550	114	"C"	12	0.1468197601	0.1821732955	0.4411695076	0.0944245151	0.0852770449
22	54573	"DOMINICKS 115"	"NAPERVILLE"	60540	417500	881136	525	115	"D"	12	0.1854935109	0.0602800546	0.0439719945	0.4063124821	0.0187639417
23	16974	"DOMINICKS 116"	"ELMHURST"	60126	418906	879597	375	116	"A"	2	0.1398772238	0.1881733901	0.0331370584	0.2592247484	0.0609819121
24	16975	"DOMINICKS 117"	"SCHAUMBURG"	60193	420164	880817	375	117	"D"	2	0.1390937165	0.1101027289	0.0498518939	0.2490835423	0.0187721969
25	16976	"DOMINICKS 118"	"MORTON GROVE"	60053	420403	877706	325	118	"A"	10	0.1037918216	0.2894423792	0.040669145	0.2247258827	0.0817836812
26	16977	"DOMINICKS 119"	"BUFFALO GROVE"	60089	421383	879572	325	119	"D"	2	0.1456695805	0.1215749651	0.0495849556	0.2799520331	0.0156220232
27	62303	"DOMINICKS 121"	"WILLOWBROOK"	60514	417728	879481	550	121	"A"	12	0.1312510967	0.1635813301	0.0391735392	0.3506128703	0.0238762471
28	54579	"DOMINICKS 122"	"HOFFMAN ESTATES"	60194	420453	881431	875	122	"D"	16	0.1673573899	0.0619539107	0.0783728849	0.2558890659	0.0192831216

c) Table: [600_group5_staging_area].[dbo].[Movement_New]



	"STORE"	"UPC"	"WEEK"	"MOVE"	"QTY"	"PRICE"	"SALE"	"PROFIT"	"OK"	CATEGORY	CATEGORY_NAME	ProductID
1	105	7089716123	355	0	1	0	0	0	1	BER	Beer	16123
2	105	7089716123	356	0	1	0	0	0	1	BER	Beer	16123
3	105	7089716123	357	0	1	0	0	0	1	BER	Beer	16123
4	105	7089716123	358	0	1	0	0	0	1	BER	Beer	16123
5	105	7089716123	359	0	1	0	0	0	1	BER	Beer	16123
6	105	7089716123	360	0	1	0	0	0	1	BER	Beer	16123
7	105	7089716123	361	0	1	0	0	0	1	BER	Beer	16123
8	105	7089716123	362	0	1	0	0	0	1	BER	Beer	16123
9	105	7089716123	363	0	1	0	0	0	1	BER	Beer	16123
10	105	7089716123	364	0	1	0	0	0	1	BER	Beer	16123
11	105	7089716123	365	0	1	0	0	0	1	BER	Beer	16123
12	105	7089716123	366	0	1	0	0	0	1	BER	Beer	16123
13	105	7089716123	367	0	1	0	0	0	1	BER	Beer	16123
14	105	7089716123	368	0	1	0	0	0	1	BER	Beer	16123
15	105	7089716123	369	0	1	0	0	0	1	BER	Beer	16123
16	105	7089716123	370	0	1	0	0	0	1	BER	Beer	16123
17	105	7089716123	371	0	1	0	0	0	1	BER	Beer	16123
18	105	7089716123	372	0	1	0	0	0	1	BER	Beer	16123
19	105	7089716123	373	0	1	0	0	0	1	BER	Beer	16123
20	105	7089716123	374	0	1	0	0	0	1	BER	Beer	16123
21	105	7089716123	375	0	1	0	0	0	1	BER	Beer	16123
22	105	7089716123	376	0	1	0	0	0	1	BER	Beer	16123
23	105	7089716123	377	0	1	0	0	0	1	BER	Beer	16123
24	105	7089716123	378	0	1	0	0	0	1	BER	Beer	16123
25	105	7089716123	379	0	1	0	0	0	1	BER	Beer	16123
26	105	7089716123	380	0	1	0	0	0	1	BER	Beer	16123
27	105	7089716123	381	0	1	0	0	0	1	BER	Beer	16123
28	105	7089716123	382	0	1	0	0	0	1	BER	Beer	16123

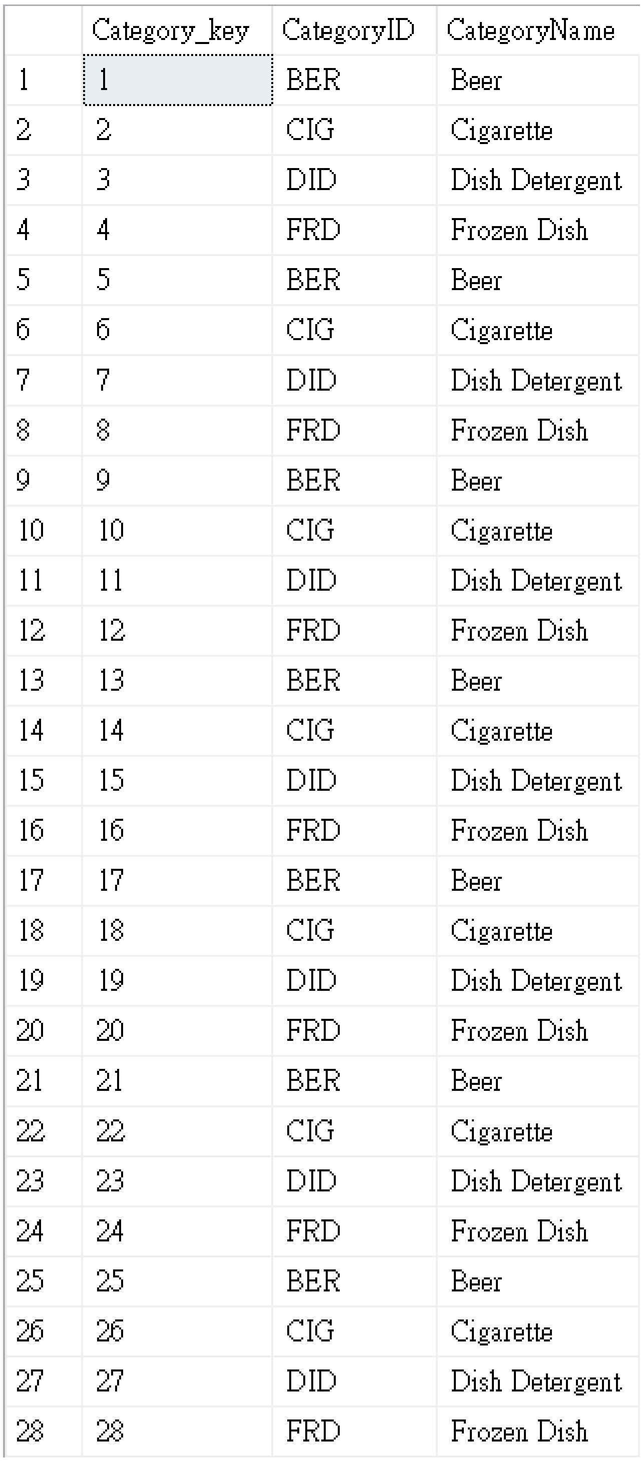
d) Table: [600_group5_staging_area].[dbo].[UPC]



	"COM_CODE"	"UPC"	"DESCRIP"	"SIZE"	"CASE"	"NITEM"	CATEGORY	UPC_PRODUCT
1	953	1192603016	"CAFFEDRINE CAPLETS 1"	"16 CT"	6	7342431	ANA	03016
2	953	1192662108	"SLEEPINAL SOFTGEL"	"8 CT"	6	7333311	ANA	62108
3	953	1650001020	"NERVINE TABS"	"30 CT"	1	8430820	ANA	01020
4	953	1650001022	"NERVINE SLEEP AID"	"12 CT"	1	8430840	ANA	01022
5	953	1650004106	"ALKA-SELTZER GOLD"	"20 CT"	1	8430880	ANA	04106
6	953	1650004108	"ALKA-SELTZER GOLD"	"36 CT"	1	8430900	ANA	04108
7	953	1650004703	"ALKA MINTS"	"30 CT"	1	8430700	ANA	04703
8	953	2140649030	"LEGATRIN PM"	"30 CT"	1	8435810	ANA	49030
9	953	2586600493	"PERCOGESIC A/F ANALG"	"50 CT"	1	8416280	ANA	00493
10	953	2586610493	"PERCOGESIC A/F ANALG"	"50 CT"	1	8416280	ANA	10493
11	953	2586610501	"ALEVE TABLETS"	"24 CT"	6	6122441	ANA	10501
12	953	2586610502	"ALEVE CAPLETS"	"24 CT"	6	6122741	ANA	10502
13	953	2586610503	"ALEVE TABLETS"	"50 CT"	6	6122451	ANA	10503
14	953	2586610504	"ALEVE CAPLETS"	"50 CT"	6	6122751	ANA	10504
15	953	2586610505	"ALEVE TABLETS"	"100 ...	6	6122461	ANA	10505
16	953	2586610506	"ALEVE CAPLETS"	"100 ...	6	6122761	ANA	10506
17	953	3225259620	"SUNBEAM HEAT WRAP ...	"1 CT"	1	8402470	ANA	59620
18	953	3680012732	"TC MOTION SICKNESS T"	"12 CT"	12	6190791	ANA	12732
19	953	3680012740	"VALUE TIME ASPIRIN"	"250 ...	12	6108051	ANA	12740
20	953	3680012742	"VALUE TIME ACETA"	"100 ...	12	6108031	ANA	12742
21	953	3680012888	"TC IBUROFEN TABLETS$"	"100 ...	6	6190091	ANA	12888
22	953	3680012890	"TC X/STR PAIN RLF TA"	"30 CT"	12	6191211	ANA	12890
23	953	3680012892	"TC X/STR N/A PAIN RE"	"24 CT"	12	6191231	ANA	12892
24	953	3680012904	"TOPCARE IBUPROFEN T...	"24 CT"	12	6191241	ANA	12904
25	953	3680014746	"TOPCARE IBUPROFEN T...	"250 ...	12	6191251	ANA	14746
26	953	3680019234	"TC SLEEPAID TABLETS"	"32 CT"	12	7393371	ANA	19234
27	953	3680019972	"TC X/STR N/A GEL CAP"	"24 CT"	12	6191011	ANA	19972
28	953	3680024827	"TC X/STR PAIN RLF TA"	"175 ...	12	6191221	ANA	24827
29	953	3680029694	"$TC PAIN REL INFANT"	".5OZ."	12	6190321	ANA	29694

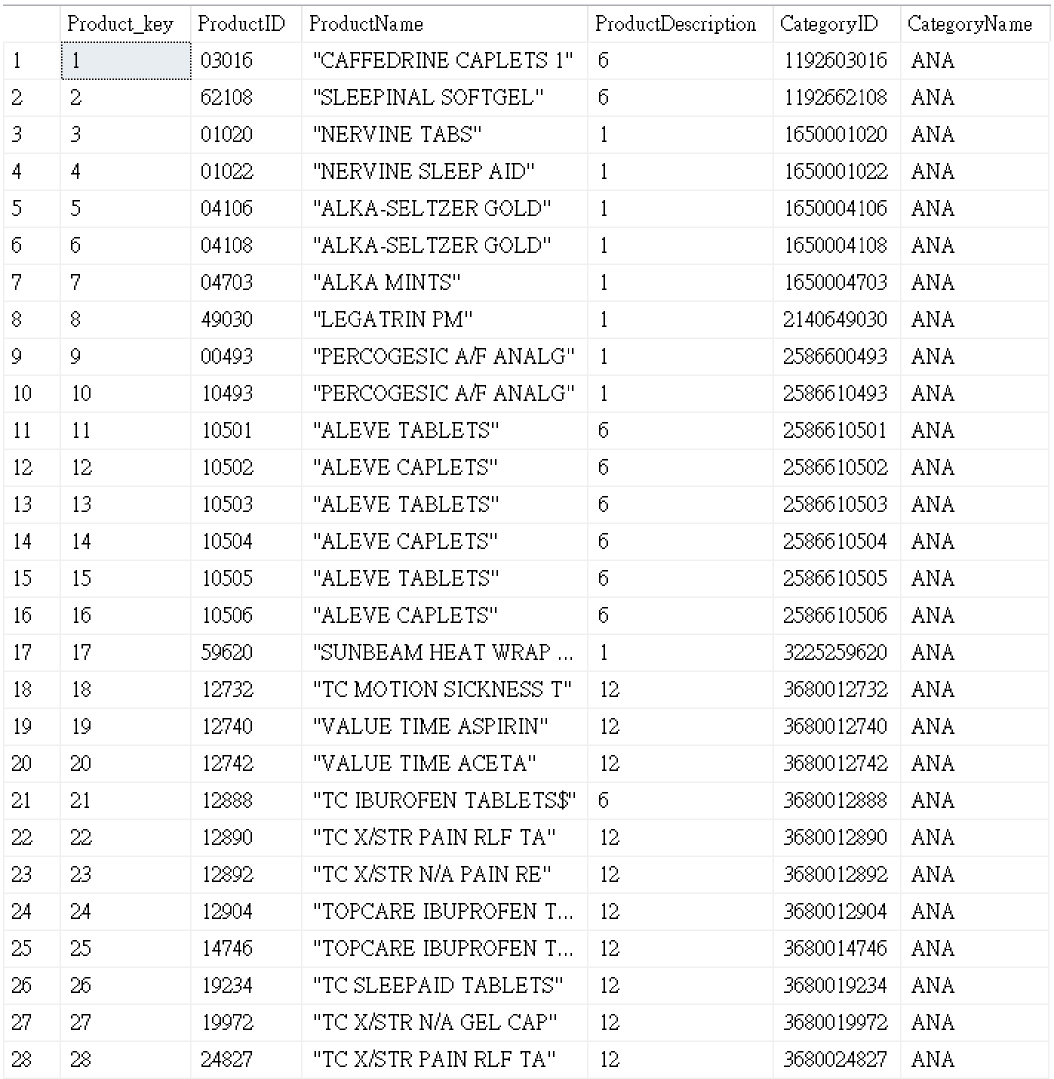
DATA MARTS TABLES

a) Table: [600_group5_datawarehouse].[dbo].[Dim_Category]



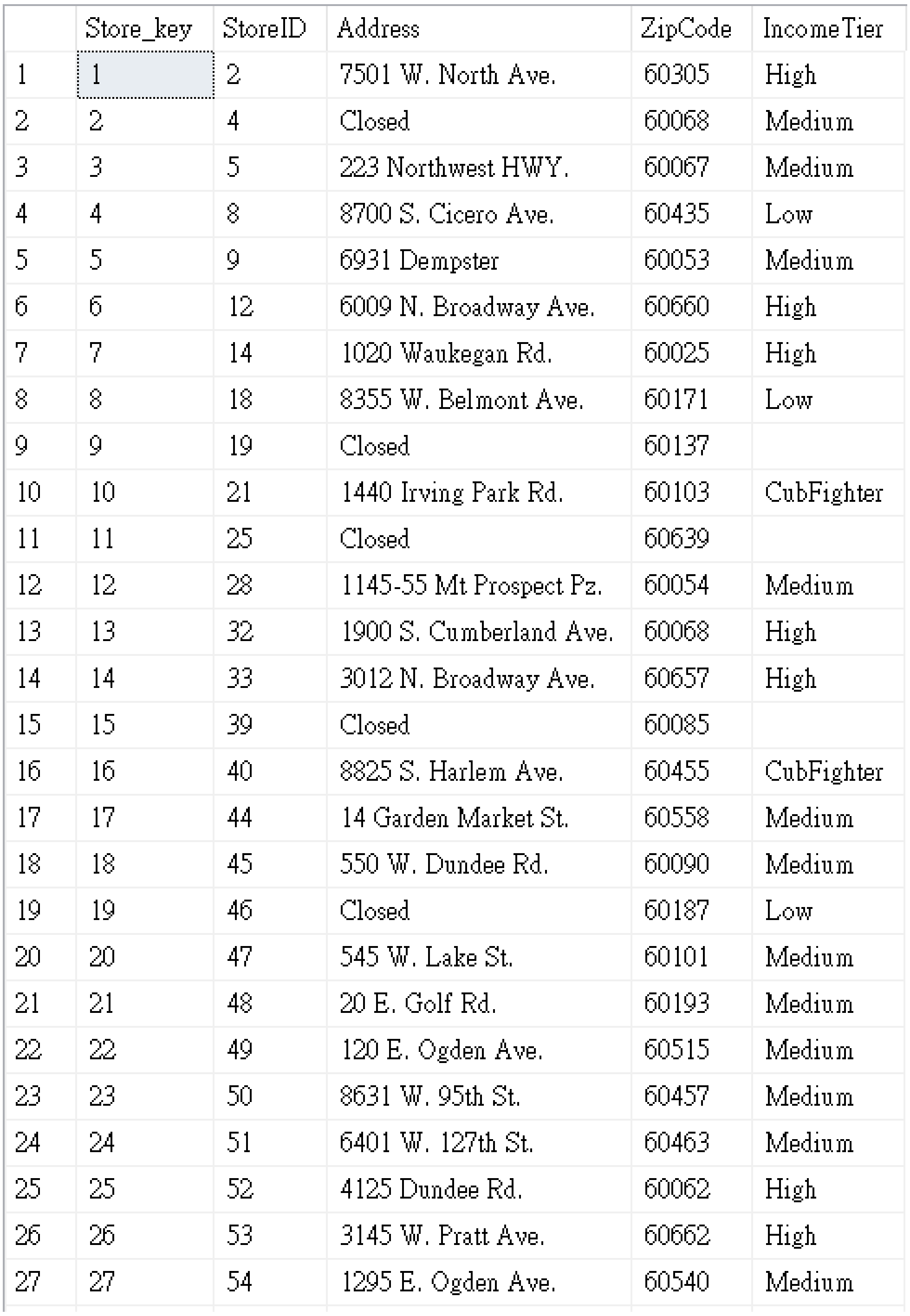
	Category_key	CategoryID	CategoryName
1	1	BER	Beer
2	2	CIG	Cigarette
3	3	DID	Dish Detergent
4	4	FRD	Frozen Dish
5	5	BER	Beer
6	6	CIG	Cigarette
7	7	DID	Dish Detergent
8	8	FRD	Frozen Dish
9	9	BER	Beer
10	10	CIG	Cigarette
11	11	DID	Dish Detergent
12	12	FRD	Frozen Dish
13	13	BER	Beer
14	14	CIG	Cigarette
15	15	DID	Dish Detergent
16	16	FRD	Frozen Dish
17	17	BER	Beer
18	18	CIG	Cigarette
19	19	DID	Dish Detergent
20	20	FRD	Frozen Dish
21	21	BER	Beer
22	22	CIG	Cigarette
23	23	DID	Dish Detergent
24	24	FRD	Frozen Dish
25	25	BER	Beer
26	26	CIG	Cigarette
27	27	DID	Dish Detergent
28	28	FRD	Frozen Dish

b) Table: [600_group5_datawarehouse].[dbo].[Dim_Product]



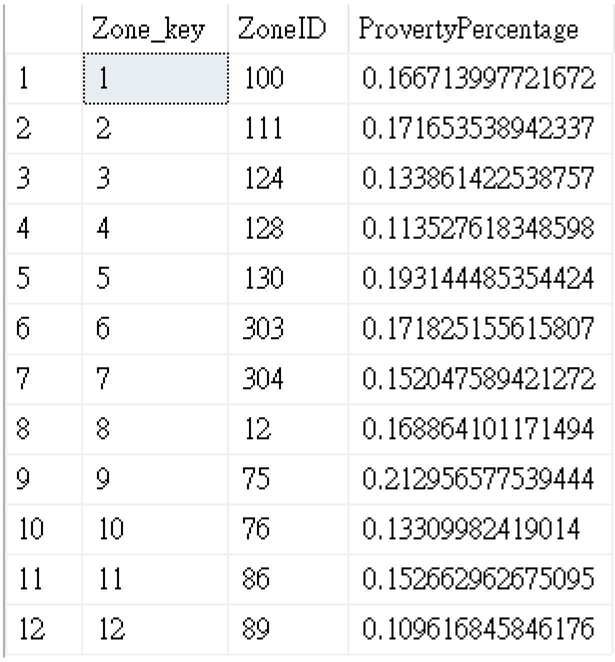
	Product_key	ProductID	ProductName	ProductDescription	CategoryID	CategoryName
1	1	03016	"CAFFEDRINE CAPLETS 1"	6	1192603016	ANA
2	2	62108	"SLEEPINAL SOFTGEL"	6	1192662108	ANA
3	3	01020	"NERVINE TABS"	1	1650001020	ANA
4	4	01022	"NERVINE SLEEP AID"	1	1650001022	ANA
5	5	04106	"ALKA-SELTZER GOLD"	1	1650004106	ANA
6	6	04108	"ALKA-SELTZER GOLD"	1	1650004108	ANA
7	7	04703	"ALKA MINTS"	1	1650004703	ANA
8	8	49030	"LEGATRIN PM"	1	2140649030	ANA
9	9	00493	"PERCOGESIC A/F ANALG"	1	2586600493	ANA
10	10	10493	"PERCOGESIC A/F ANALG"	1	2586610493	ANA
11	11	10501	"ALEVE TABLETS"	6	2586610501	ANA
12	12	10502	"ALEVE CAPLETS"	6	2586610502	ANA
13	13	10503	"ALEVE TABLETS"	6	2586610503	ANA
14	14	10504	"ALEVE CAPLETS"	6	2586610504	ANA
15	15	10505	"ALEVE TABLETS"	6	2586610505	ANA
16	16	10506	"ALEVE CAPLETS"	6	2586610506	ANA
17	17	59620	"SUNBEAM HEAT WRAP ...	1	3225259620	ANA
18	18	12732	"TC MOTION SICKNESS T"	12	3680012732	ANA
19	19	12740	"VALUE TIME ASPIRIN"	12	3680012740	ANA
20	20	12742	"VALUE TIME ACETA"	12	3680012742	ANA
21	21	12888	"TC IBUROFEN TABLETS$"	6	3680012888	ANA
22	22	12890	"TC X/STR PAIN RLF TA"	12	3680012890	ANA
23	23	12892	"TC X/STR N/A PAIN RE"	12	3680012892	ANA
24	24	12904	"TOPCARE IBUPROFEN T...	12	3680012904	ANA
25	25	14746	"TOPCARE IBUPROFEN T...	12	3680014746	ANA
26	26	19234	"TC SLEEPAID TABLETS"	12	3680019234	ANA
27	27	19972	"TC X/STR N/A GEL CAP"	12	3680019972	ANA
28	28	24827	"TC X/STR PAIN RLF TA"	12	3680024827	ANA

c) Table: [600_group5_datawarehouse].[dbo].[Dim_Store]



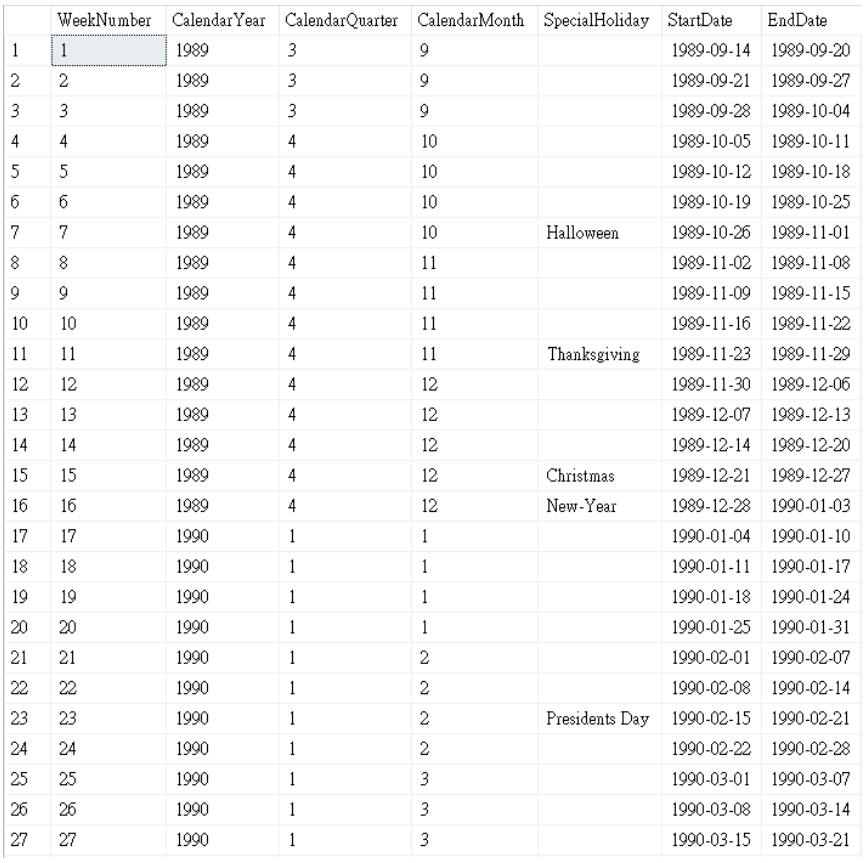
	Store_key	StoreID	Address	ZipCode	IncomeTier
1	1	2	7501 W. North Ave.	60305	High
2	2	4	Closed	60068	Medium
3	3	5	223 Northwest HWY.	60067	Medium
4	4	8	8700 S. Cicero Ave.	60435	Low
5	5	9	6931 Dempster	60053	Medium
6	6	12	6009 N. Broadway Ave.	60660	High
7	7	14	1020 Waukegan Rd.	60025	High
8	8	18	8355 W. Belmont Ave.	60171	Low
9	9	19	Closed	60137	
10	10	21	1440 Irving Park Rd.	60103	CubFighter
11	11	25	Closed	60639	
12	12	28	1145-55 Mt Prospect Pz.	60054	Medium
13	13	32	1900 S. Cumberland Ave.	60068	High
14	14	33	3012 N. Broadway Ave.	60657	High
15	15	39	Closed	60085	
16	16	40	8825 S. Harlem Ave.	60455	CubFighter
17	17	44	14 Garden Market St.	60558	Medium
18	18	45	550 W. Dundee Rd.	60090	Medium
19	19	46	Closed	60187	Low
20	20	47	545 W. Lake St.	60101	Medium
21	21	48	20 E. Golf Rd.	60193	Medium
22	22	49	120 E. Ogden Ave.	60515	Medium
23	23	50	8631 W. 95th St.	60457	Medium
24	24	51	6401 W. 127th St.	60463	Medium
25	25	52	4125 Dundee Rd.	60062	High
26	26	53	3145 W. Pratt Ave.	60662	High
27	27	54	1295 E. Ogden Ave.	60540	Medium

d) Table: [600_group5_datawarehouse].[dbo].[Dim_StoreDemograhpic]



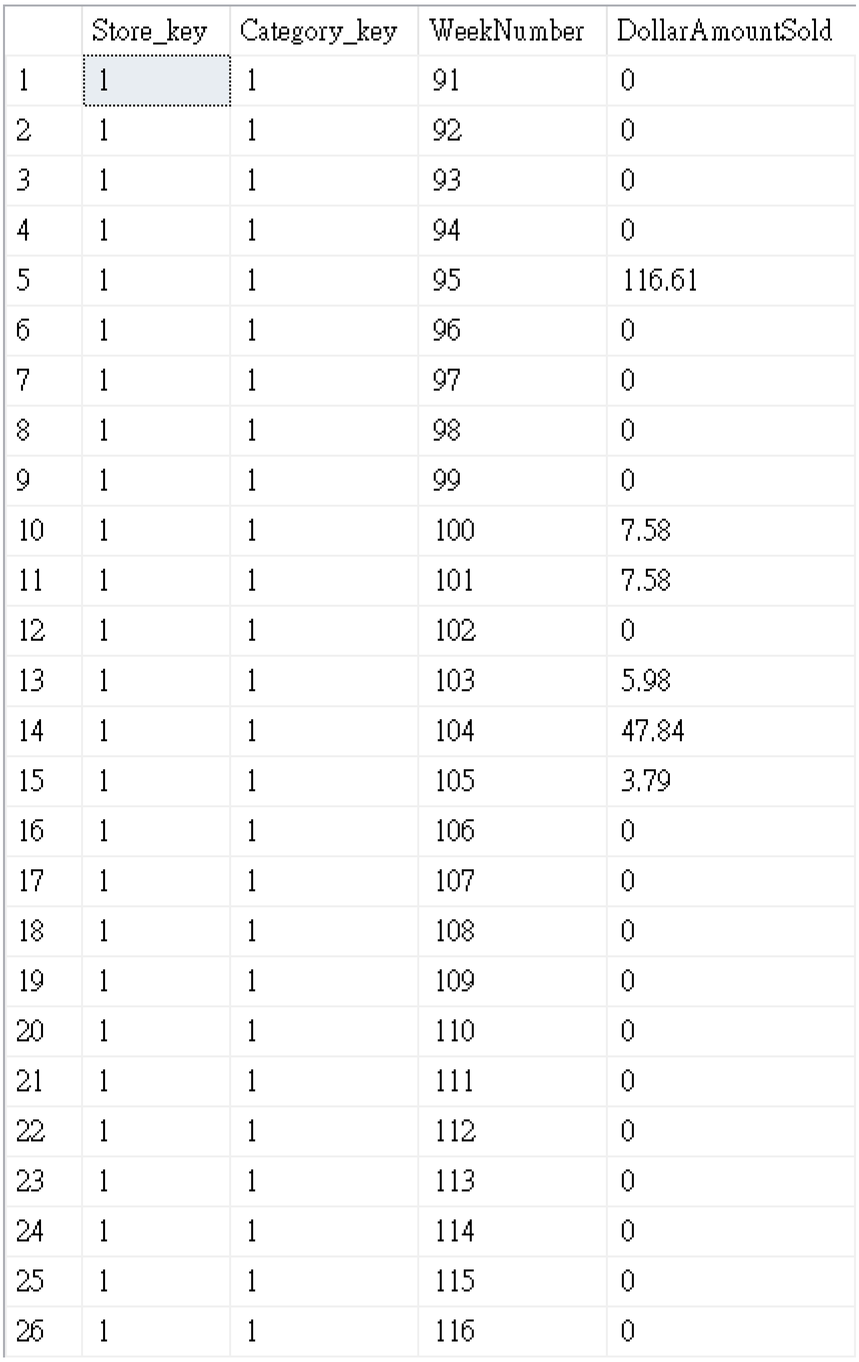
	Zone_key	ZoneID	ProvertyPercentage
1	1	100	0.166713997721672
2	2	111	0.171653538942337
3	3	124	0.133861422538757
4	4	128	0.113527618348598
5	5	130	0.193144485354424
6	6	303	0.171825155615807
7	7	304	0.152047589421272
8	8	12	0.168864101171494
9	9	75	0.212956577539444
10	10	76	0.13309982419014
11	11	86	0.152662962675095
12	12	89	0.109616845846176

e) Table: [600_group5_datawarehouse].[dbo].[Dim_Time]



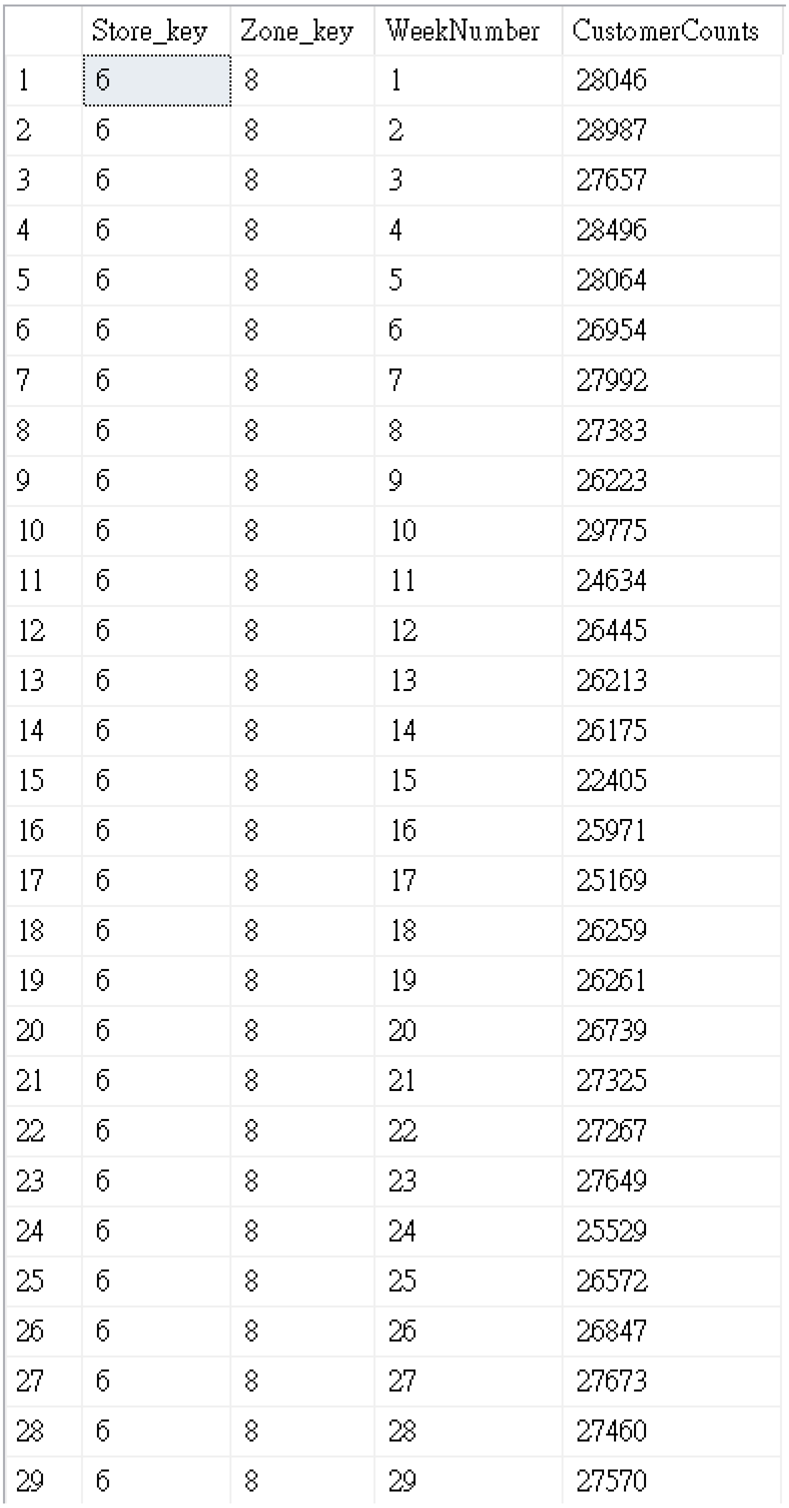
	WeekNumber	CalendarYear	CalendarQuarter	CalendarMonth	SpecialHoliday	StartDate	EndDate
1	1	1989	3	9		1989-09-14	1989-09-20
2	2	1989	3	9		1989-09-21	1989-09-27
3	3	1989	3	9		1989-09-28	1989-10-04
4	4	1989	4	10		1989-10-05	1989-10-11
5	5	1989	4	10		1989-10-12	1989-10-18
6	6	1989	4	10		1989-10-19	1989-10-25
7	7	1989	4	10	Halloween	1989-10-26	1989-11-01
8	8	1989	4	11		1989-11-02	1989-11-08
9	9	1989	4	11		1989-11-09	1989-11-15
10	10	1989	4	11		1989-11-16	1989-11-22
11	11	1989	4	11	Thanksgiving	1989-11-23	1989-11-29
12	12	1989	4	12		1989-11-30	1989-12-06
13	13	1989	4	12		1989-12-07	1989-12-13
14	14	1989	4	12		1989-12-14	1989-12-20
15	15	1989	4	12	Christmas	1989-12-21	1989-12-27
16	16	1989	4	12	New-Year	1989-12-28	1990-01-03
17	17	1990	1	1		1990-01-04	1990-01-10
18	18	1990	1	1		1990-01-11	1990-01-17
19	19	1990	1	1		1990-01-18	1990-01-24
20	20	1990	1	1		1990-01-25	1990-01-31
21	21	1990	1	2		1990-02-01	1990-02-07
22	22	1990	1	2		1990-02-08	1990-02-14
23	23	1990	1	2	Presidents Day	1990-02-15	1990-02-21
24	24	1990	1	2		1990-02-22	1990-02-28
25	25	1990	1	3		1990-03-01	1990-03-07
26	26	1990	1	3		1990-03-08	1990-03-14
27	27	1990	1	3		1990-03-15	1990-03-21

f) Table: [600_group5_datawarehouse].[dbo].[Category_Sales_Fact]



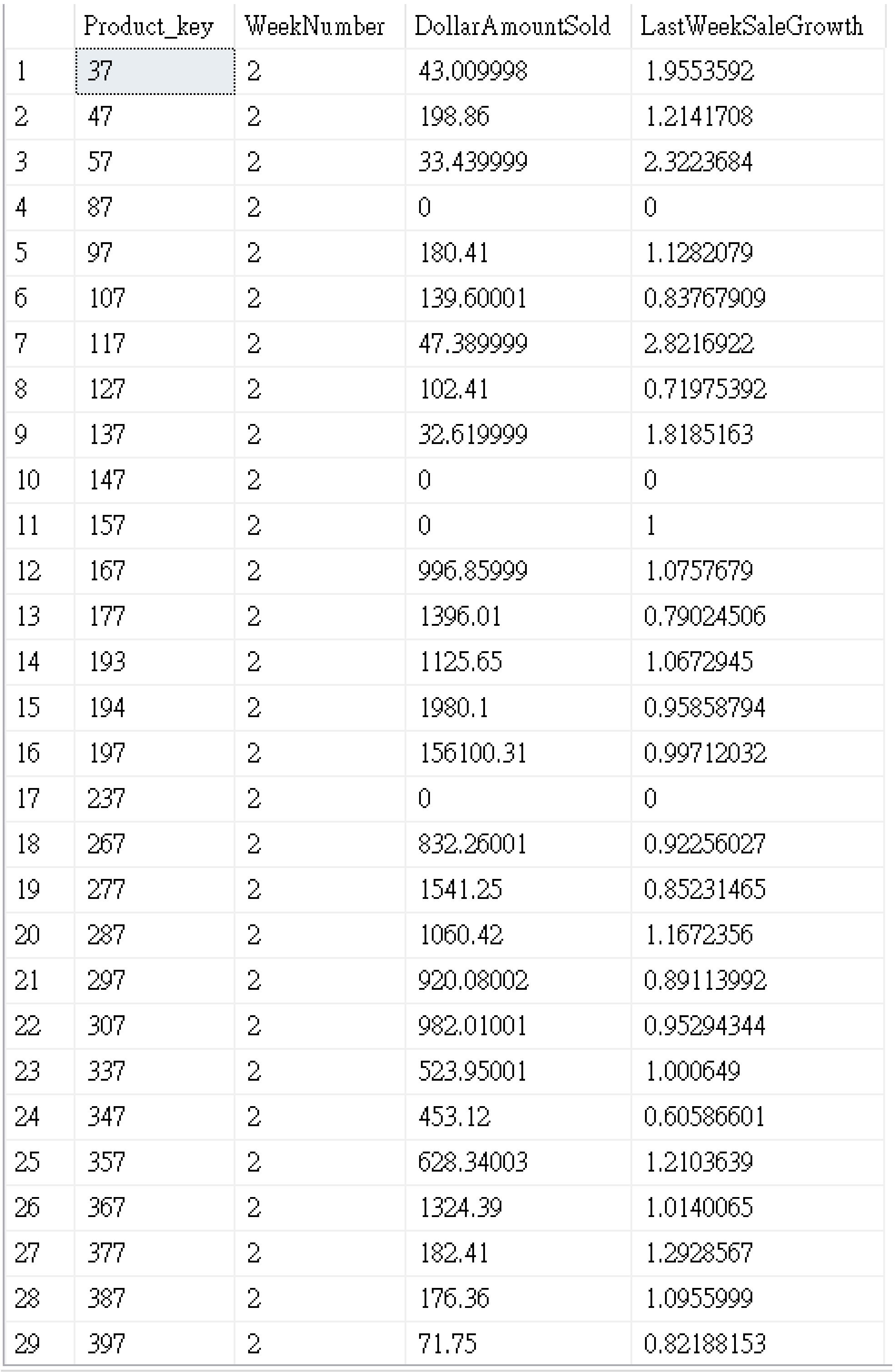
	Store_key	Category_key	WeekNumber	DollarAmountSold
1	1	1	91	0
2	1	1	92	0
3	1	1	93	0
4	1	1	94	0
5	1	1	95	116.61
6	1	1	96	0
7	1	1	97	0
8	1	1	98	0
9	1	1	99	0
10	1	1	100	7.58
11	1	1	101	7.58
12	1	1	102	0
13	1	1	103	5.98
14	1	1	104	47.84
15	1	1	105	3.79
16	1	1	106	0
17	1	1	107	0
18	1	1	108	0
19	1	1	109	0
20	1	1	110	0
21	1	1	111	0
22	1	1	112	0
23	1	1	113	0
24	1	1	114	0
25	1	1	115	0
26	1	1	116	0

g) Table: [600_group5_datawarehouse].[dbo].[Customer_Count_Fact]



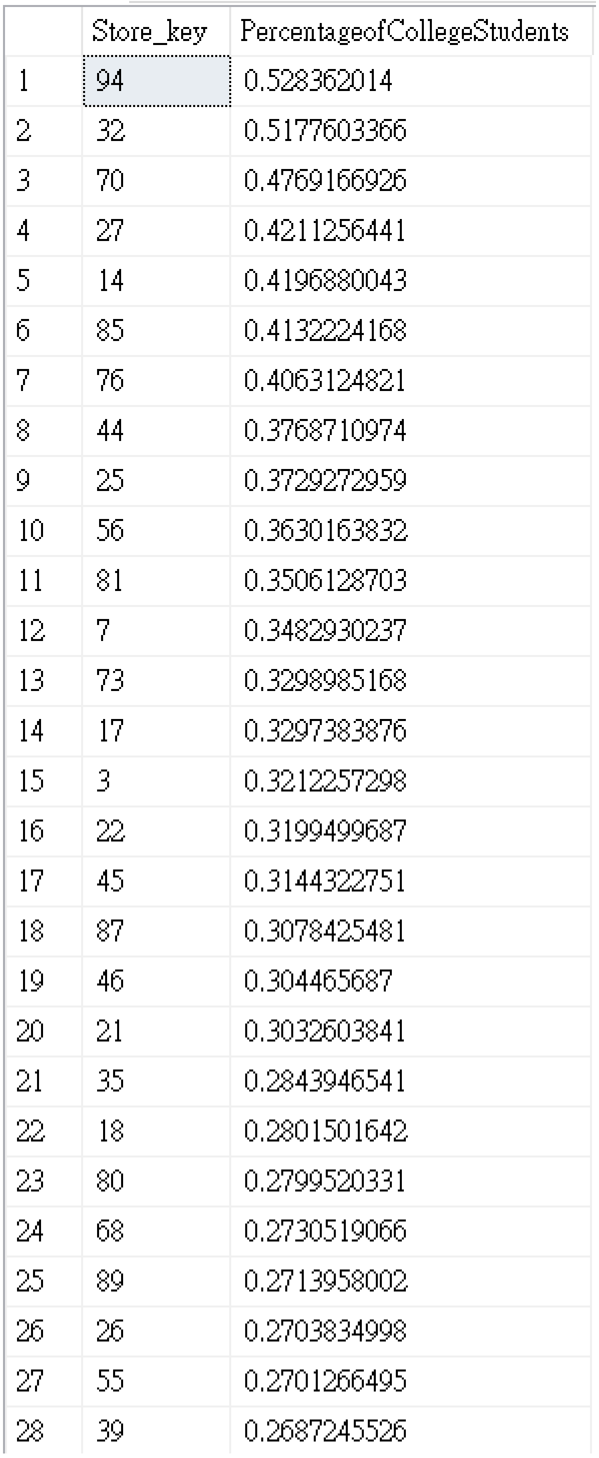
	Store_key	Zone_key	WeekNumber	CustomerCounts
1	6	8	1	28046
2	6	8	2	28987
3	6	8	3	27657
4	6	8	4	28496
5	6	8	5	28064
6	6	8	6	26954
7	6	8	7	27992
8	6	8	8	27383
9	6	8	9	26223
10	6	8	10	29775
11	6	8	11	24634
12	6	8	12	26445
13	6	8	13	26213
14	6	8	14	26175
15	6	8	15	22405
16	6	8	16	25971
17	6	8	17	25169
18	6	8	18	26259
19	6	8	19	26261
20	6	8	20	26739
21	6	8	21	27325
22	6	8	22	27267
23	6	8	23	27649
24	6	8	24	25529
25	6	8	25	26572
26	6	8	26	26847
27	6	8	27	27673
28	6	8	28	27460
29	6	8	29	27570

h) Table: [600_group5_datawarehouse].[dbo].[Product_Sales_Fact]



	Product_key	WeekNumber	DollarAmountSold	LastWeekSaleGrowth
1	37	2	43.009998	1.9553592
2	47	2	198.86	1.2141708
3	57	2	33.439999	2.3223684
4	87	2	0	0
5	97	2	180.41	1.1282079
6	107	2	139.60001	0.83767909
7	117	2	47.389999	2.8216922
8	127	2	102.41	0.71975392
9	137	2	32.619999	1.8185163
10	147	2	0	0
11	157	2	0	1
12	167	2	996.85999	1.0757679
13	177	2	1396.01	0.79024506
14	193	2	1125.65	1.0672945
15	194	2	1980.1	0.95858794
16	197	2	156100.31	0.99712032
17	237	2	0	0
18	267	2	832.26001	0.92256027
19	277	2	1541.25	0.85231465
20	287	2	1060.42	1.1672356
21	297	2	920.08002	0.89113992
22	307	2	982.01001	0.95294344
23	337	2	523.95001	1.000649
24	347	2	453.12	0.60586601
25	357	2	628.34003	1.2103639
26	367	2	1324.39	1.0140065
27	377	2	182.41	1.2928567
28	387	2	176.36	1.0955999
29	397	2	71.75	0.82188153

i) Table: [600_group5_datawarehouse].[dbo].[Store_Demographics_Fact]



	Store_key	PercentageofCollegeStudents
1	94	0.528362014
2	32	0.5177603366
3	70	0.4769166926
4	27	0.4211256441
5	14	0.4196880043
6	85	0.4132224168
7	76	0.4063124821
8	44	0.3768710974
9	25	0.3729272959
10	56	0.3630163832
11	81	0.3506128703
12	7	0.3482930237
13	73	0.3298985168
14	17	0.3297383876
15	3	0.3212257298
16	22	0.3199499687
17	45	0.3144322751
18	87	0.3078425481
19	46	0.304465687
20	21	0.3032603841
21	35	0.2843946541
22	18	0.2801501642
23	80	0.2799520331
24	68	0.2730519066
25	89	0.2713958002
26	26	0.2703834998
27	55	0.2701266495
28	39	0.2687245526

Data Granularity at the Independent Data Mart Level

Data granularity is the level of details that is included in the data mart based on the needs of the user. If the users don’t need to go deeper into the details then summarized data could be used such as monthly or yearly trends. On the other hand, if the users wish to look into more details then we ought to keep a deeper level of data such as weekly or daily trends.

Target Table	Target Column	Target Data Type
[600_group5_datawarehouse].[dbo].[Dim_Time]	WeekNumber	int
	CalendarYear	int
	CalendarQuarter	int
	CalendarMonth	int
	SpecialHoliday	varchar
	StartDate	date
	EndDate	date

If we look at the time dimension, the level of detail we used is year, quarter and month as well as special holiday. So, we provided a wide range of levels so that users can access any level of detail in the data as needed. For our business questions, the granularity of our data mart is chosen as store, week, product, and category level.

List of all temporary tables that were removed from the staging area

- **[600_group5_staging_area].[dbo].[Customer Count]**
- **[600_group5_staging_area].[dbo].[Movement]**

Special things done in our ETL Task

There are many movement files in different folders. We used C# code to traverse all the folders and capture the folder name. We use the string split method and the substring method to calculate the category name and saved it as a new column. We finally then used Input-Output steam to combine all CSV files together and save it as a single CSV for future extraction.

When we calculated the growth rate, chances are that the previous week has 0 sale records. In order to avoid divided by 0 error from SQL server, we used IF ELSE statement to define the growth rate. If both this week and last week are 0, then the growth rate is 0, if last week is 0 and this week is greater than 0, then the growth rate will be set to 100%. otherwise growth rate will be set to this week-last week/last week.

BI REPORTING PLAN

Target reports that satisfy the business questions and why

BQ 1: Which category has the highest selling for each store over a span of one week?

Tools Used: SSRS+SSAS

Approach Taken: We built an SSRS report over the SSAS cube which we created first for answering this question. We are calculating the highest selling category in a week. We are dealing with only 4 categories in order to maintain the speed of the data server. They are beer, frozen dinner, detergent and cigarettes. The Data source that is the Sec 600_Group5 is selected and Category_Sales_Fact Table, Dim_Time, Dim_Category and Dim_Store tables are selected from the data mart. The tables listed above are selected in Data Source View. A mapping is created between the selected tables. The cube is created by selecting Category_Sales_Fact as the measure group tables and Dim_Category, Dim_Time and Dim_Store as dimensions. Then we specified the hierarchies, defined the server and deployed the cube on infodata.tamu.edu.

In the SSRS part, we built the report and chart to answer our question. We first select the data source type as SSAS cube and then select the database in which the cube is stored. We then build the query using the query designer which has the same functions as it was while working on SSAS cube. The Category_Sales_Fact is in the measures group from which the highest selling category of a particular week is selected while week is selected from Dim_Time and one of the 4 categories is selected from Dim_Category. We use the percentage change formula to calculate the weekly sales. IF last week has zero sales the increase in sales is 100% . In the report wizard, the increase in sales is grouped by week number and we show the top 3 categories in the sales category name.

The server is defined as http://infodata16.mbs.tamu.edu/ReportServer/Pages/ReportViewer.aspx?%2f600Group4ZhongzhuZhou%**2fBU1+Category+Sales&rs:Command=Render**

The deployment folder is set as **600Group4ZhongzhuZhou**

BQ 2: Which store sees a higher number of footfalls from people below the poverty line?

Tools Used: SSRS Alone

Approach Taken: The first step was to select the data source type as Microsoft SQL Server and the database was selected as 600_group5_datawarehouse. The query was built using the query designer. The tables namely, Customer_Count_Fact, Dim_StoreDemographic, Dim_Time, Dim_Store were selected. Week numbers were sorted in ascending order and customer count in descending order to get the highest customer count. We grouped the number of footfalls by week number so that we can calculate the number of customers each week. Our report shows the StoreID, Address and customer count details which are then deployed on the server.

The target server is http://infodata16.mbs.tamu.edu/ReportServer?%2f600-Group5-report&rs:Command=ListChildren

Target folder name **600-Group5-report/Report4**

Log In Name: ch6606
Password: Mays6606

BQ 3: What are the products that have shown slow or static growth?

Tools Used: SSAS Alone

Approach Taken: First of all, we made a connection to the data warehouse in SSAS. We have used SSAS alone to build the report for this question. The Data source that is the 600_Group 5_Datawarehouse is selected and ProductSalesFact, Dim_Time and Dim_Product tables are selected from the data mart. The above tables listed are selected in data source view. Then, we modified the data source view by adding a new named query. The new query aggregates sales records for different categories for each week. We created a mapping between the selected tables. The cube is created by selecting ProductSalesFAct as the measure group tables and Dim_Time and Dim_Product as Dimensions. After this, the hierarchies are specified if there are any for each dimension. When the cube is created, we deploy it to the analysis server. Finally, managers could go to the SQL analysis server and execute query in the cube

The cube is listed under **Multidimensional-600-group4Z** cube in SQL analysis sever.

BQ 4: Which product sales record is the most volatile over a span of 40 weeks?

Tools Used: Tableau alone

Approach Taken: We first made connection to SSAS via Tableau data source tab. In the SSAS cube, there is one fact table and two dimensions. The sales growth fact table is connected with the time dimension and product dimension. Then, we created a new worksheet in Tableau and plotted scatters based on sales growth for different products over time. We applied products' names to the color mark to show different products. We also added a filter to the worksheet to allow users to select products they are interested in. Finally, we published the worksheet to Tableau public servers and embedded it in the html page.

The reports will be visible here:
https://public.tableau.com/views/DominickDemo/Sheet1?:language=en&:display_count=y&:origin=viz_share_link

BQ 5: Which store is in a district with the most college students?

Tools Used: SSRS Alone

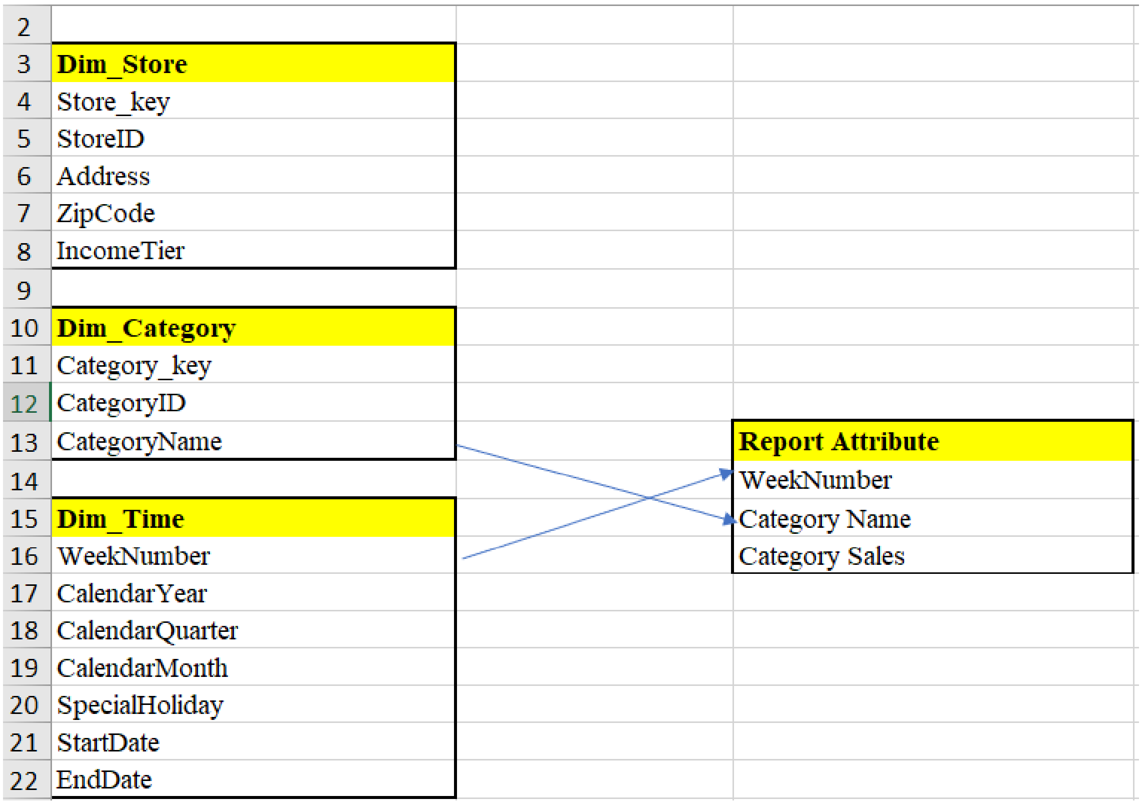
Approach Taken: We first select the data source type as Microsoft SQL server and then select the database as 600_Group 5_warehouse. We then build the query using the query designer. The tables Dim_Store and Store_Demographics_Fact are selected.

The columns Store_key and PercentageOfCollegeStudent in Store_Demographics_Fact are selected. Besides, the column ZipCode in Dim_Store is selected. We sorted the PercentageOfCollegeStudent column in descending order to show the store which has the highest percentage of college students. The report is then deployed on the server.

The target server is set to **infodata16.tamu.edu/ReportServer**
The target folder name is **BQ10-Report**
Log In Name: ch6606
Password: Mays6606

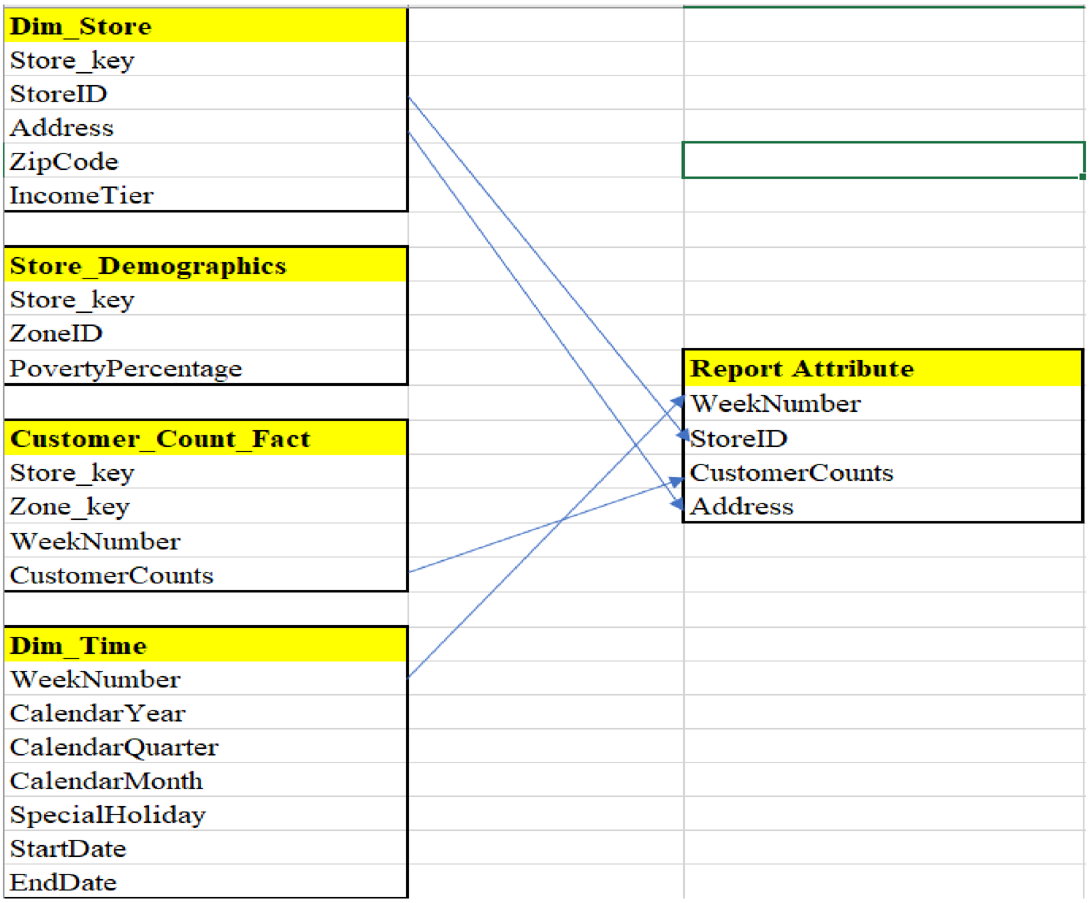
Mappings from Independent Data Marts to Individual Report Attributes

BQ1: Which category has the highest selling for each store over a span of one week?



2			
3	**Dim_Store**		
4	Store_key		
5	StoreID		
6	Address		
7	ZipCode		
8	IncomeTier		
9			
10	**Dim_Category**		
11	Category_key		
12	CategoryID		
13	CategoryName		**Report Attribute**
14			WeekNumber
15	**Dim_Time**		Category Name
16	WeekNumber		Category Sales
17	CalendarYear		
18	CalendarQuarter		
19	CalendarMonth		
20	SpecialHoliday		
21	StartDate		
22	EndDate		

BQ 2: Which store sees a higher number of footfalls from people below the poverty line?



Dim_Store
Store_key
StoreID
Address
ZipCode
IncomeTier

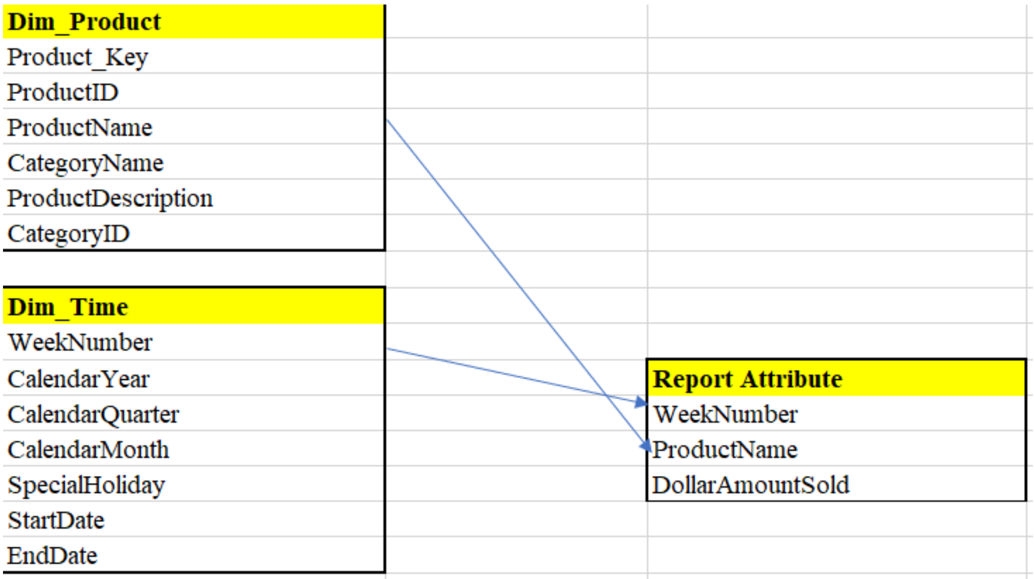
Store_Demographics
Store_key
ZoneID
PovertyPercentage

Customer_Count_Fact
Store_key
Zone_key
WeekNumber
CustomerCounts

Dim_Time
WeekNumber
CalendarYear
CalendarQuarter
CalendarMonth
SpecialHoliday
StartDate
EndDate

Report Attribute
WeekNumber
StoreID
CustomerCounts
Address

BQ 3: What are the products that have shown slow or static growth?

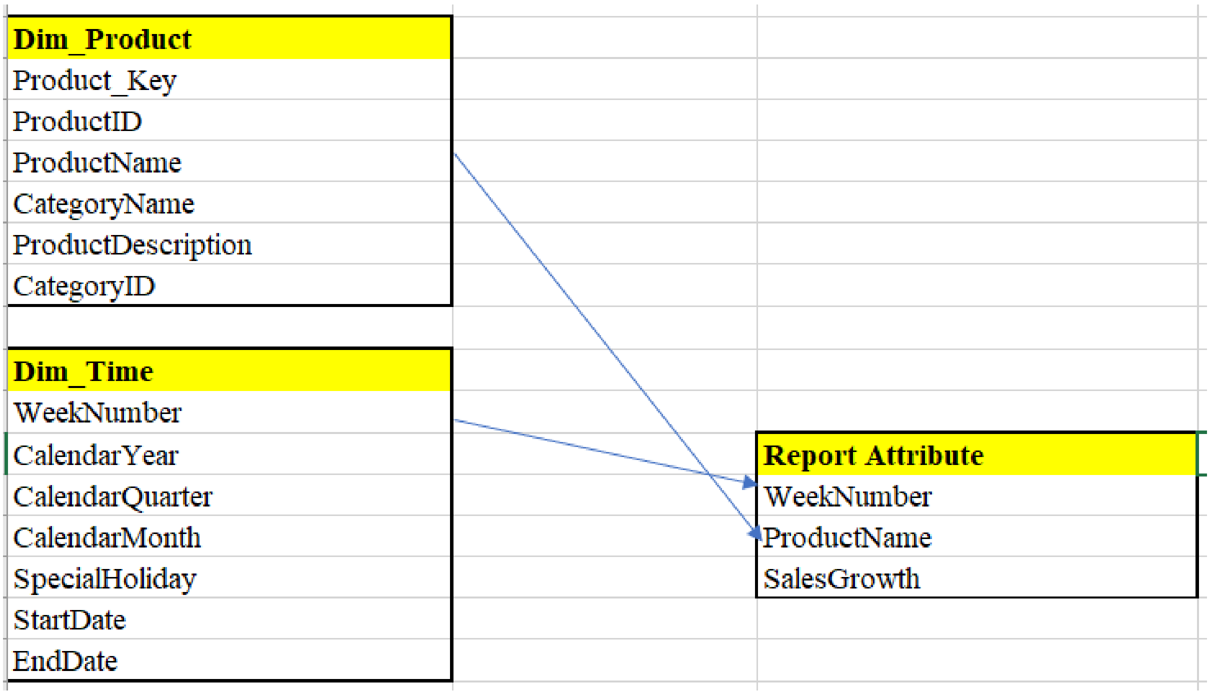


Dim_Product
Product_Key
ProductID
ProductName
CategoryName
ProductDescription
CategoryID

Dim_Time
WeekNumber
CalendarYear
CalendarQuarter
CalendarMonth
SpecialHoliday
StartDate
EndDate

Report Attribute
WeekNumber
ProductName
DollarAmountSold

BQ 4: Which product sales record is the most volatile over a span of 40 weeks?

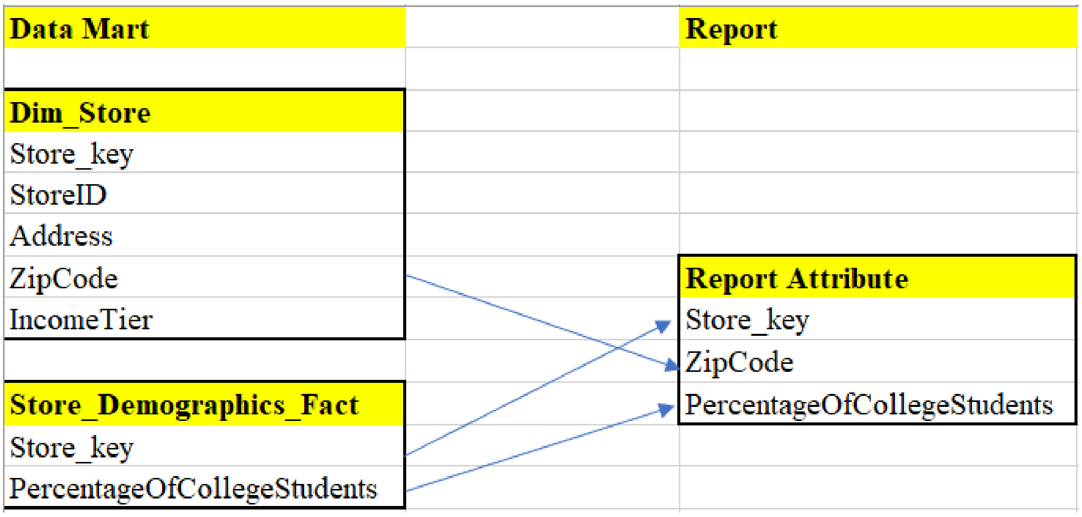


Dim_Product
Product_Key
ProductID
ProductName
CategoryName
ProductDescription
CategoryID

Dim_Time
WeekNumber
CalendarYear
CalendarQuarter
CalendarMonth
SpecialHoliday
StartDate
EndDate

Report Attribute
WeekNumber
ProductName
SalesGrowth

BQ 5: Which store is in a district with the most college students?



Data Mart

Dim_Store
Store_key
StoreID
Address
ZipCode
IncomeTier

Store_Demographics_Fact
Store_key
PercentageOfCollegeStudents

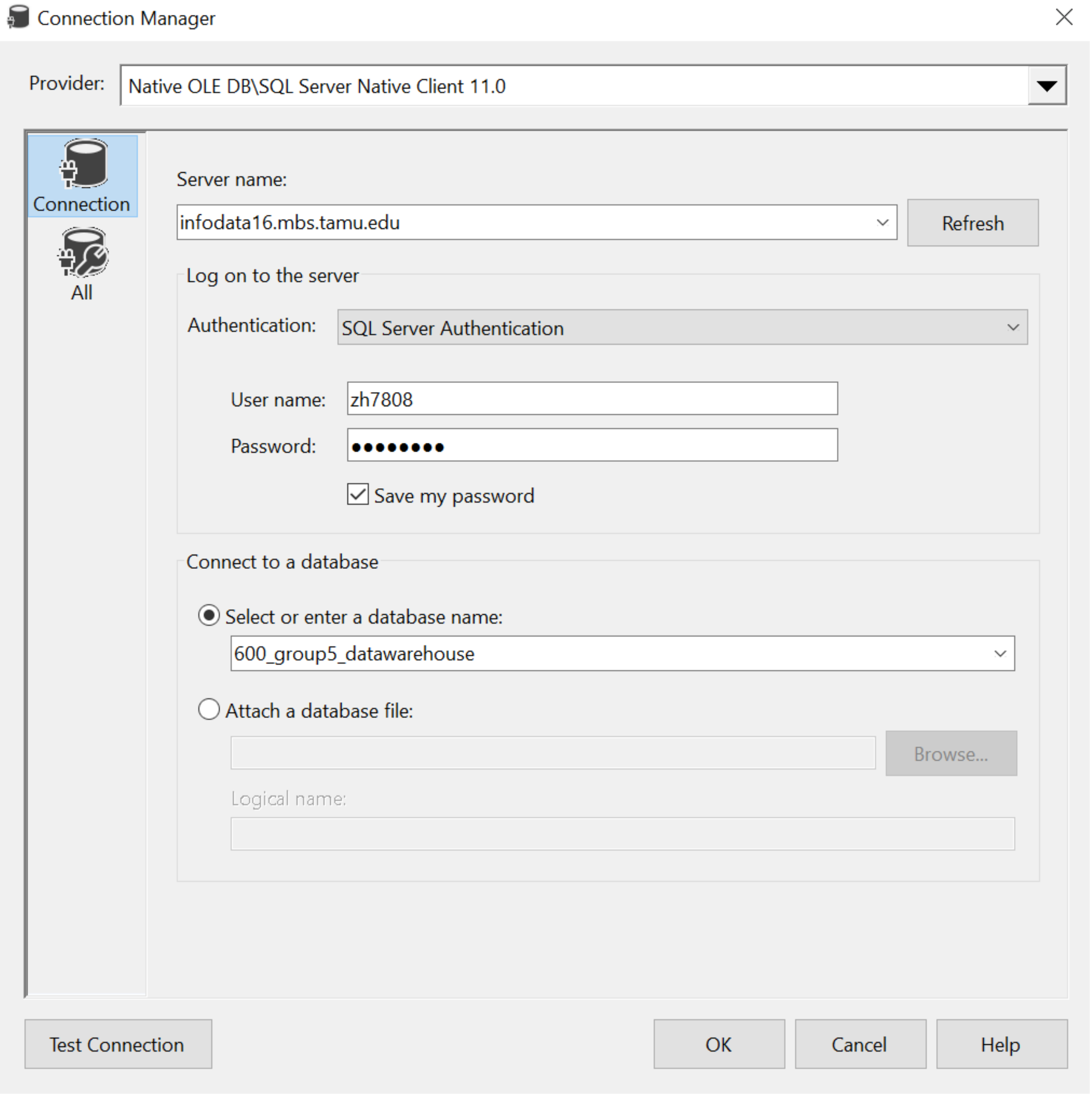
Report

Report Attribute
Store_key
ZipCode
PercentageOfCollegeStudents

REPORTS

Reports from Cubes using SSRS and SSAS

BQ1: Which category has the highest selling for each store over a span of one week?



Connection Manager

Provider: Native OLE DB\SQL Server Native Client 11.0

Server name: infodata16.mbs.tamu.edu

Log on to the server

Authentication: SQL Server Authentication

User name: zh7808

Password: ••••••••

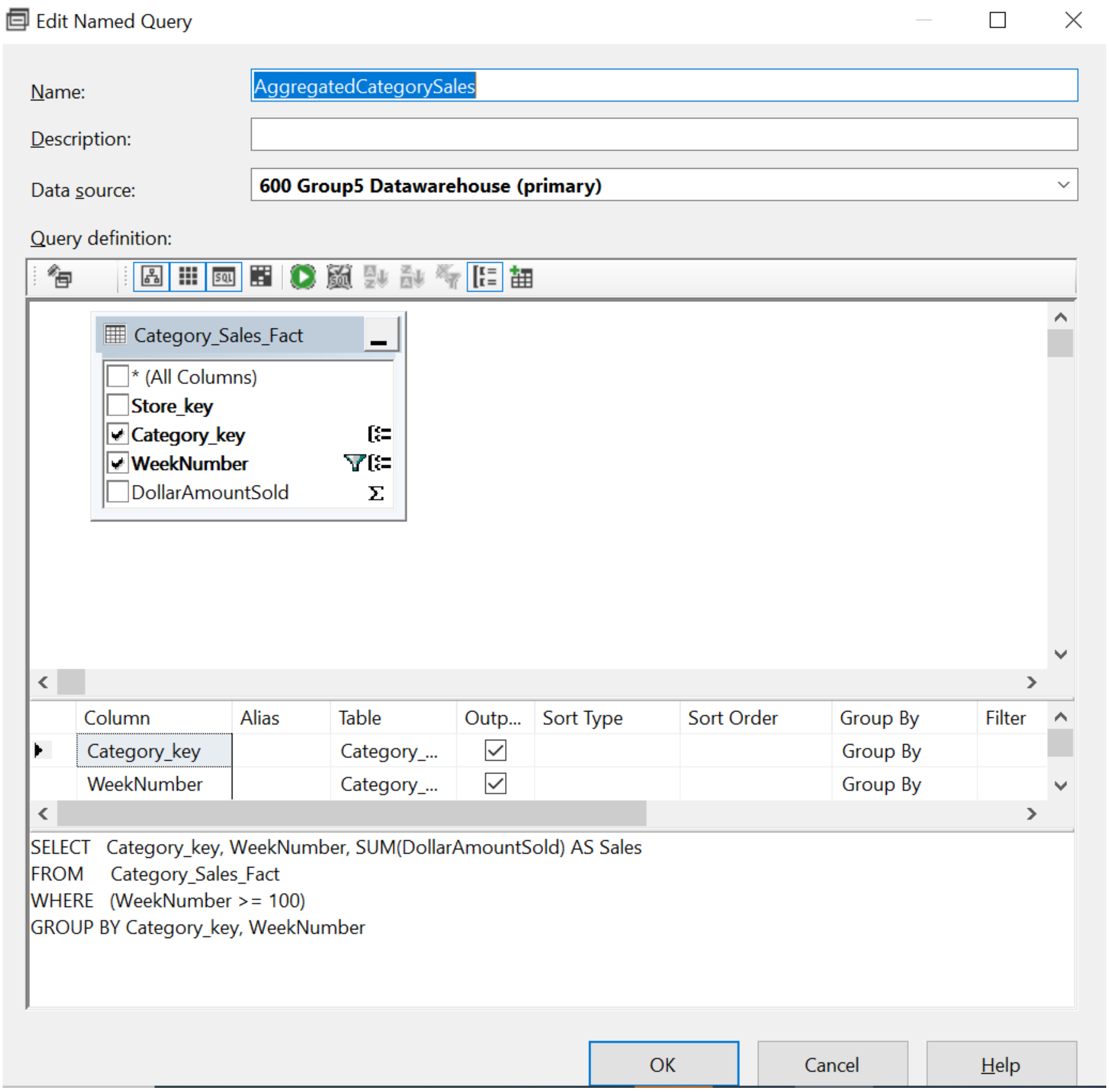
☑ Save my password

Connect to a database

● Select or enter a database name: 600_group5_datawarehouse

○ Attach a database file:

Logical name:



Name: AggregatedCategorySales

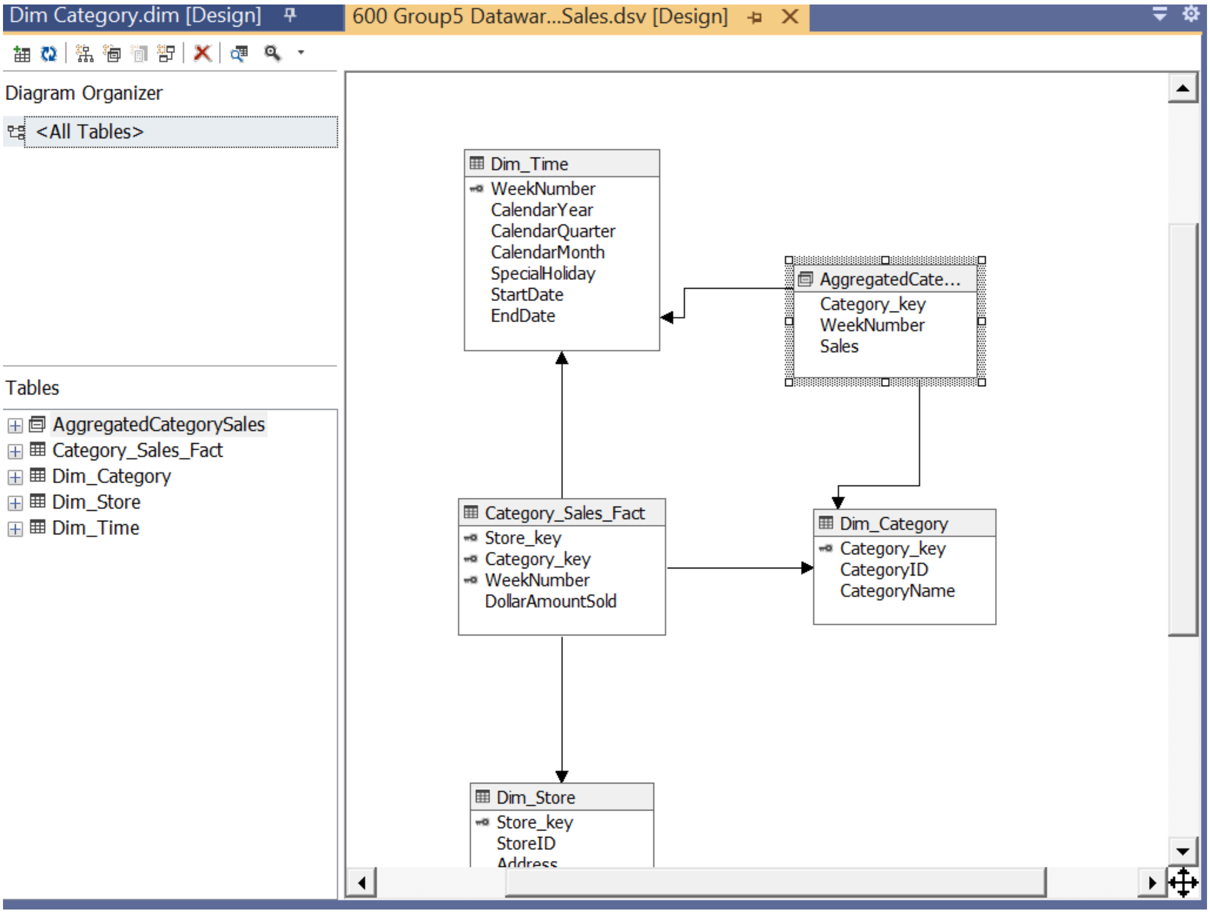
Description:

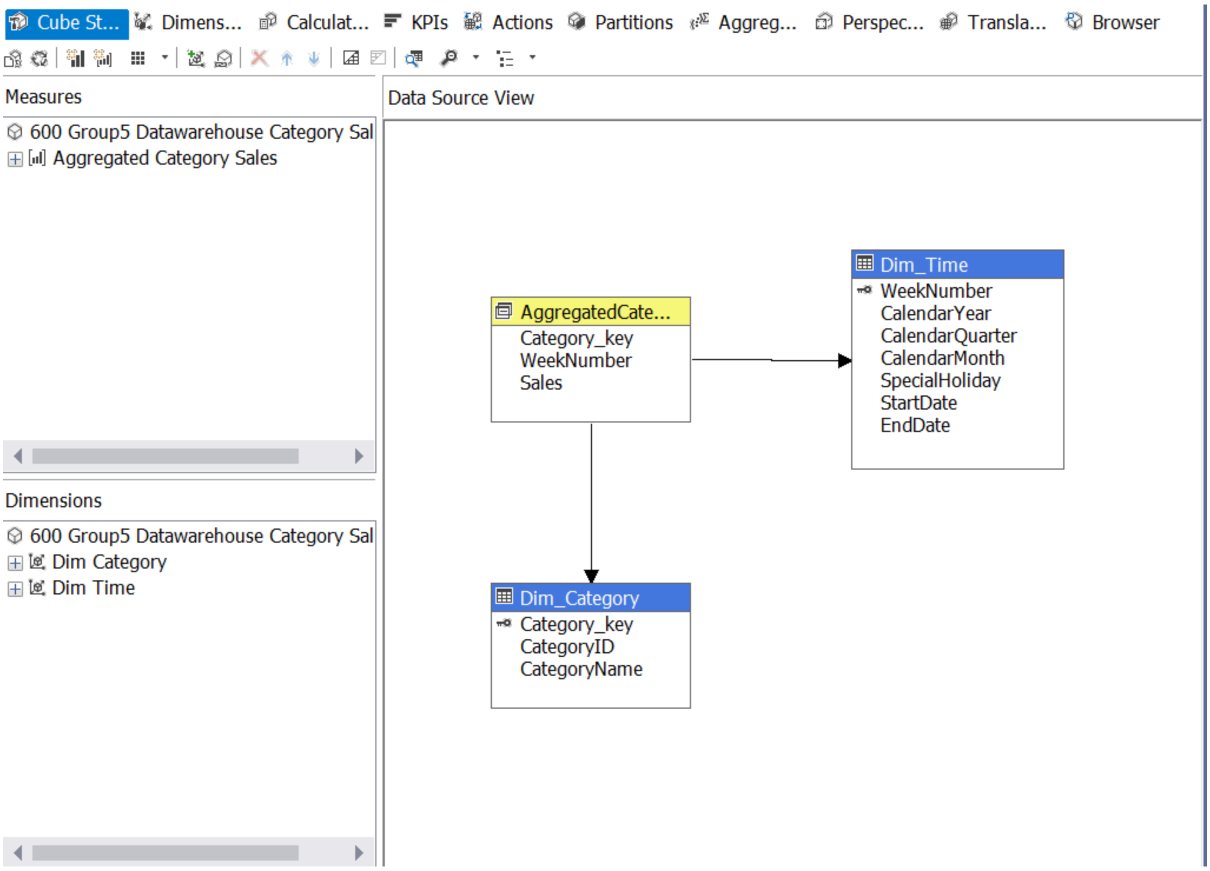
Data source: 600 Group5 Datawarehouse (primary)

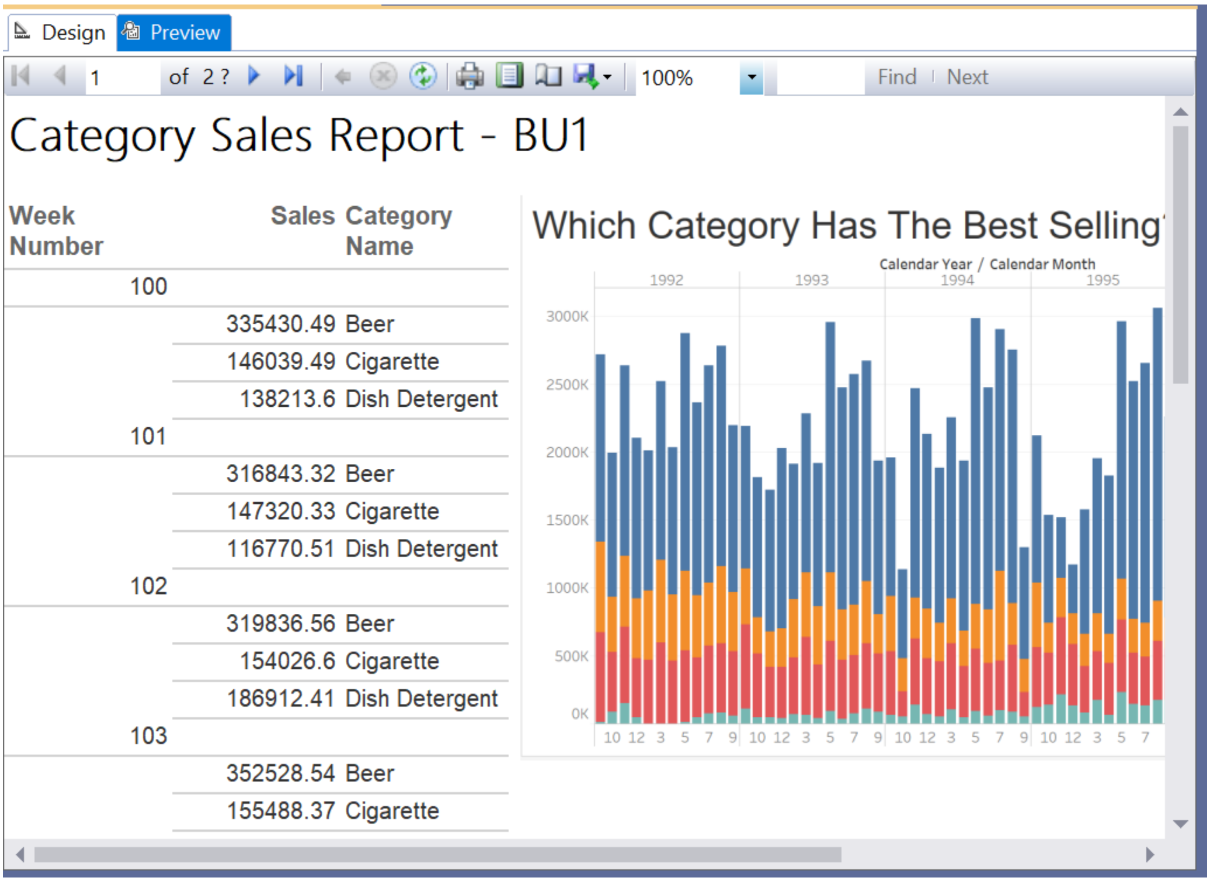
Query definition:

Column	Alias	Table	Outp...	Sort Type	Sort Order	Group By	Filter
Category_key		Category_...	☑			Group By	
WeekNumber		Category_...	☑			Group By	

```
SELECT   Category_key, WeekNumber, SUM(DollarAmountSold) AS Sales
FROM     Category_Sales_Fact
WHERE    (WeekNumber >= 100)
GROUP BY Category_key, WeekNumber
```


Dim Category.dim [Design]
600 Group5 Datawar...Sales.dsv [Design]
Diagram Organizer
<All Tables>
Tables
AggregatedCategorySales
Category_Sales_Fact
Dim_Category
Dim_Store
Dim_Time
Dim_Time
WeekNumber
CalendarYear
CalendarQuarter
CalendarMonth
SpecialHoliday
StartDate
EndDate
AggregatedCate...
Category_key
WeekNumber
Sales
Category_Sales_Fact
Store_key
Category_key
WeekNumber
DollarAmountSold
Dim_Category
Category_key
CategoryID
CategoryName
Dim_Store
Store_key
StoreID
Address

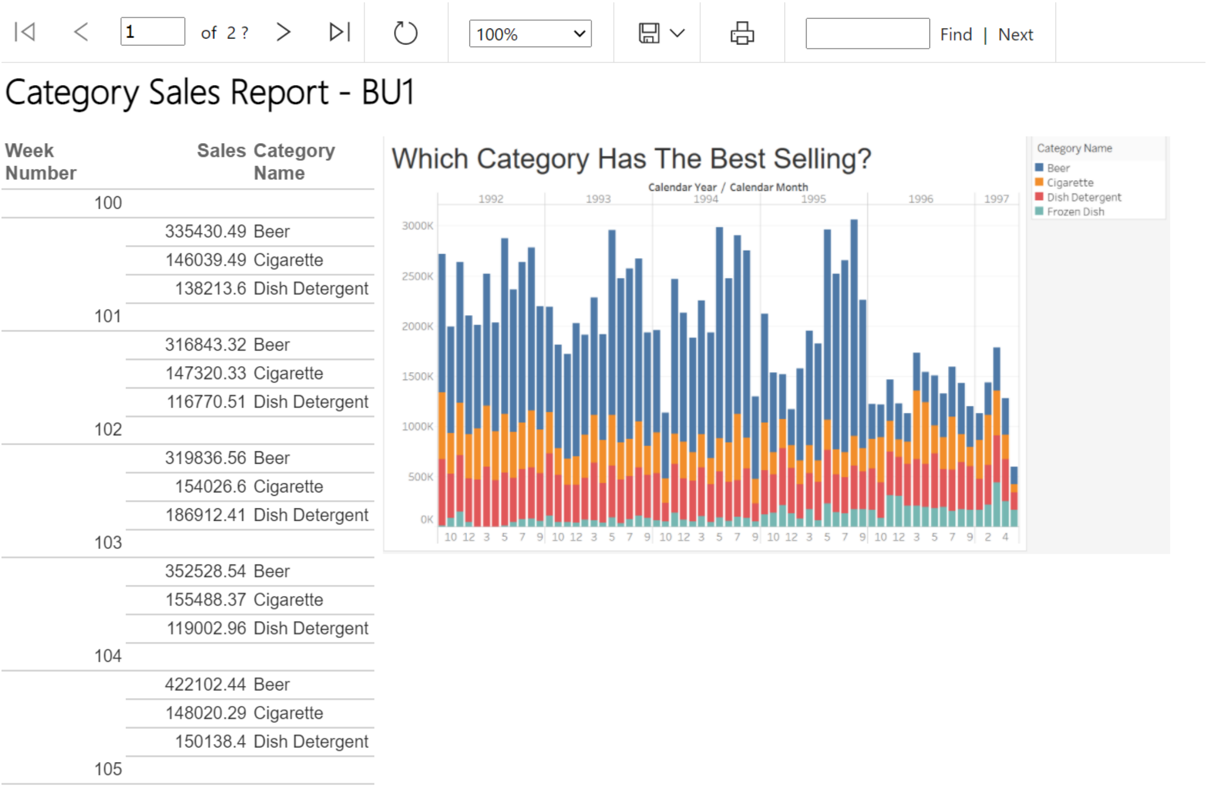




Category Sales Report - BU1

Week Number	Sales	Category Name
100		
	335430.49	Beer
	146039.49	Cigarette
	138213.6	Dish Detergent
101		
	316843.32	Beer
	147320.33	Cigarette
	116770.51	Dish Detergent
102		
	319836.56	Beer
	154026.6	Cigarette
	186912.41	Dish Detergent
103		
	352528.54	Beer
	155488.37	Cigarette

Which Category Has The Best Selling

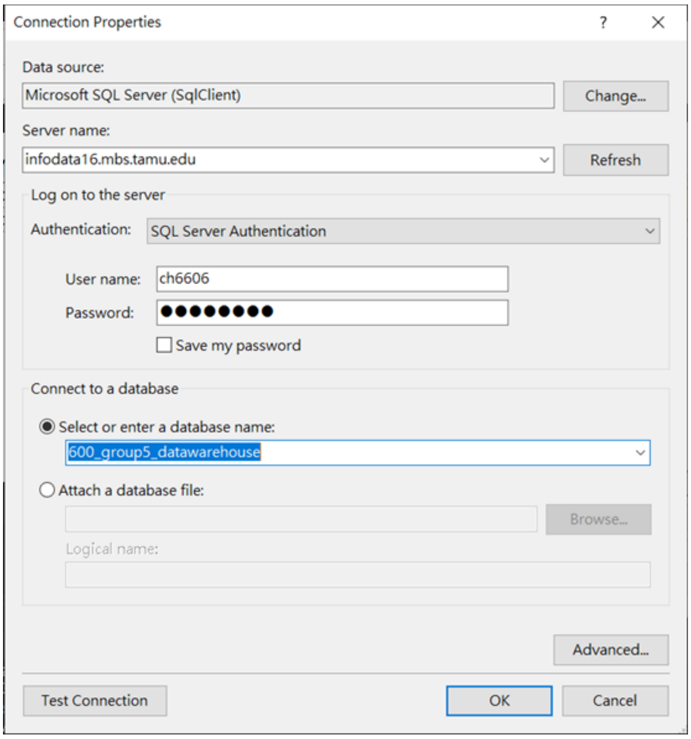


Category Sales Report - BU1

Week Number	Sales	Category Name
100		
	335430.49	Beer
	146039.49	Cigarette
	138213.6	Dish Detergent
101		
	316843.32	Beer
	147320.33	Cigarette
	116770.51	Dish Detergent
102		
	319836.56	Beer
	154026.6	Cigarette
	186912.41	Dish Detergent
103		
	352528.54	Beer
	155488.37	Cigarette
	119002.96	Dish Detergent
104		
	422102.44	Beer
	148020.29	Cigarette
	150138.4	Dish Detergent
105		

Reports from Independent Data Marts using SSRS

BQ2: Which store sees a higher number of footfalls from people below the poverty line?



Connection Properties

Data source: Microsoft SQL Server (SqlClient)

Server name: infodata16.mbs.tamu.edu

Log on to the server

Authentication: SQL Server Authentication

User name: ch6606

Password: ●●●●●●●●

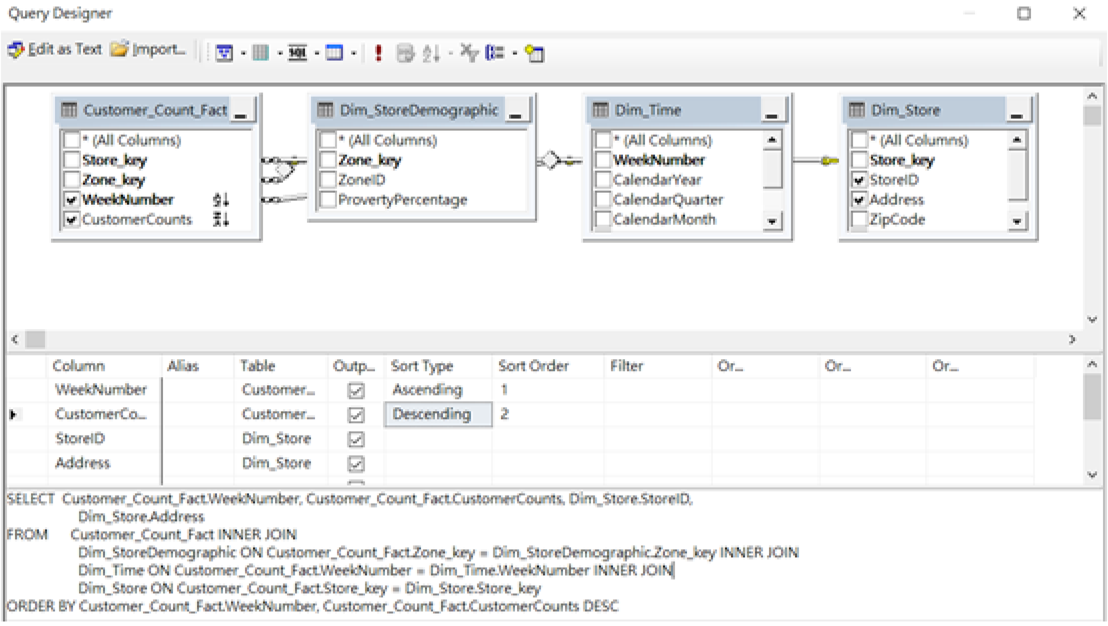
□ Save my password

Connect to a database

● Select or enter a database name: 600_group5_datawarehouse

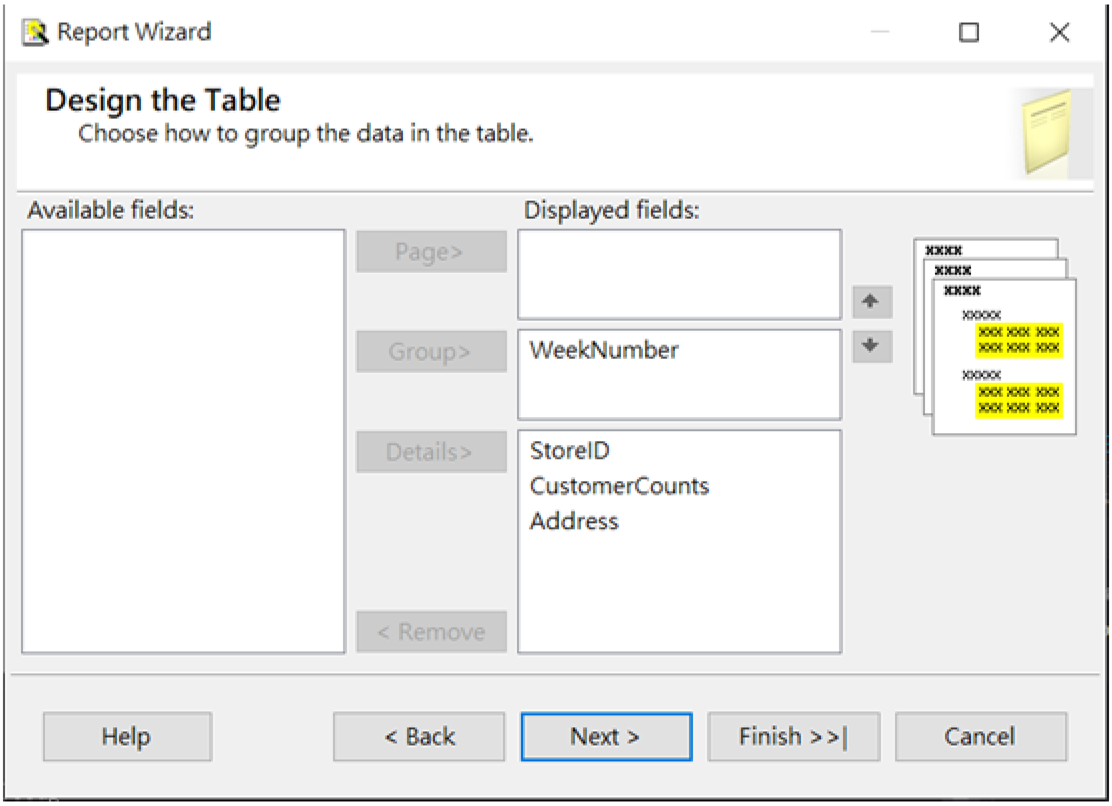
○ Attach a database file:

Logical name:



Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...	Or...
WeekNumber		Customer...	☑	Ascending	1				
CustomerCo...		Customer...	☑	Descending	2				
StoreID		Dim_Store	☑						
Address		Dim_Store	☑						

```
SELECT  Customer_Count_Fact.WeekNumber, Customer_Count_Fact.CustomerCounts, Dim_Store.StoreID,
         Dim_Store.Address
FROM    Customer_Count_Fact INNER JOIN
         Dim_StoreDemographic ON Customer_Count_Fact.Zone_key = Dim_StoreDemographic.Zone_key INNER JOIN
         Dim_Time ON Customer_Count_Fact.WeekNumber = Dim_Time.WeekNumber INNER JOIN
         Dim_Store ON Customer_Count_Fact.Store_key = Dim_Store.Store_key
ORDER BY Customer_Count_Fact.WeekNumber, Customer_Count_Fact.CustomerCounts DESC
```



Report Wizard

Design the Table

Choose how to group the data in the table.

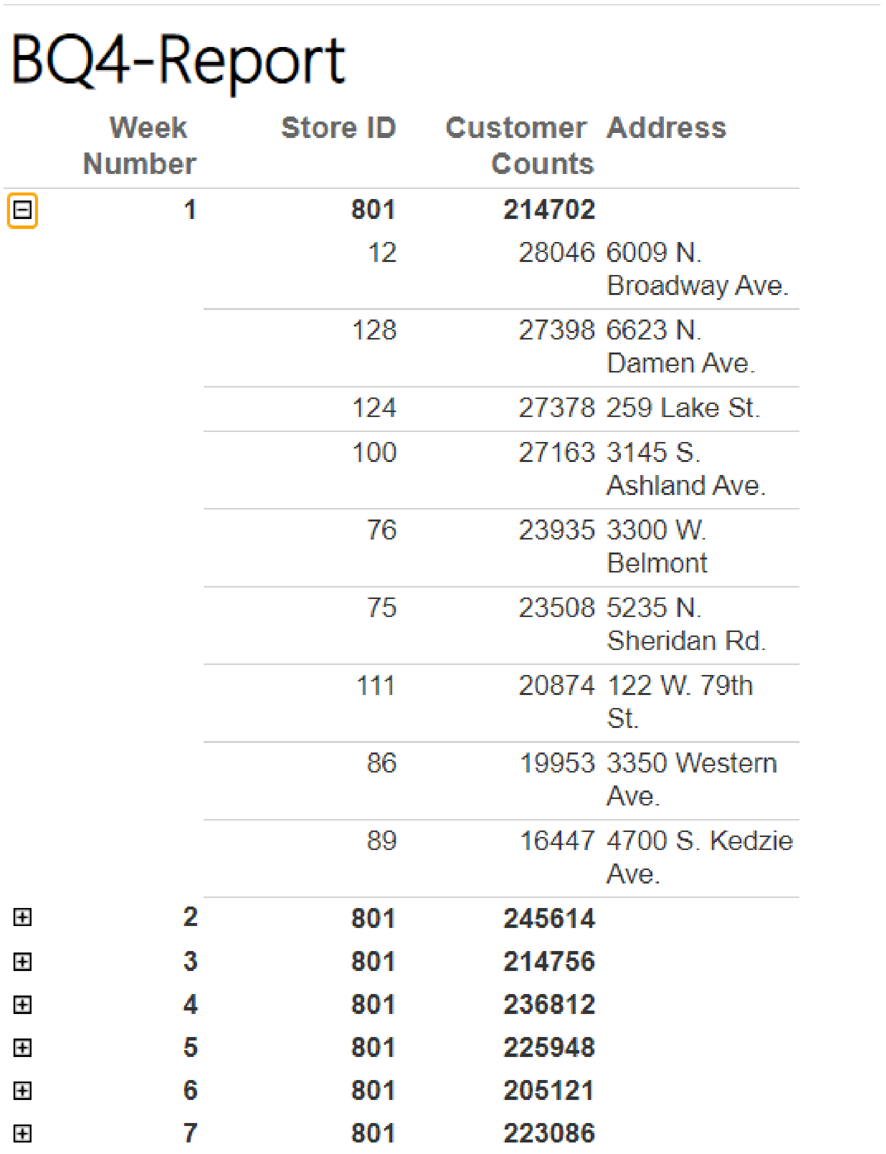
Available fields:

Displayed fields:

Page: (none)

Group: WeekNumber

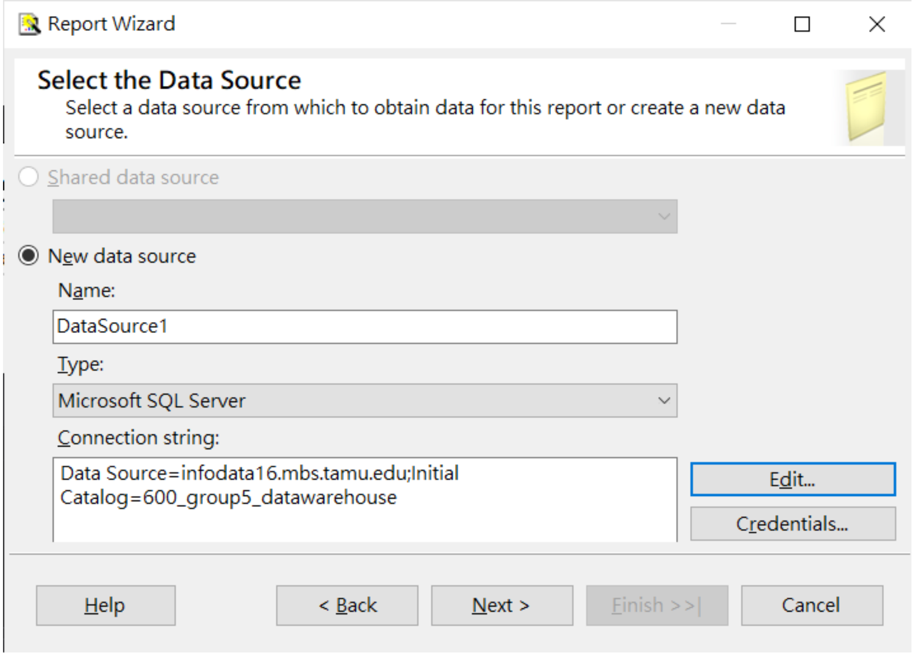
Details: StoreID, CustomerCounts, Address



BQ4-Report

	Week Number	Store ID	Customer Counts	Address
⊟	**1**	**801**	**214702**	
		12	28046	6009 N. Broadway Ave.
		128	27398	6623 N. Damen Ave.
		124	27378	259 Lake St.
		100	27163	3145 S. Ashland Ave.
		76	23935	3300 W. Belmont
		75	23508	5235 N. Sheridan Rd.
		111	20874	122 W. 79th St.
		86	19953	3350 Western Ave.
		89	16447	4700 S. Kedzie Ave.
⊞	**2**	**801**	**245614**	
⊞	**3**	**801**	**214756**	
⊞	**4**	**801**	**236812**	
⊞	**5**	**801**	**225948**	
⊞	**6**	**801**	**205121**	
⊞	**7**	**801**	**223086**	

BQ4: Which store is in a district with the most college students?



Select the Data Source

Select a data source from which to obtain data for this report or create a new data source.

○ Shared data source

● New data source

Name:

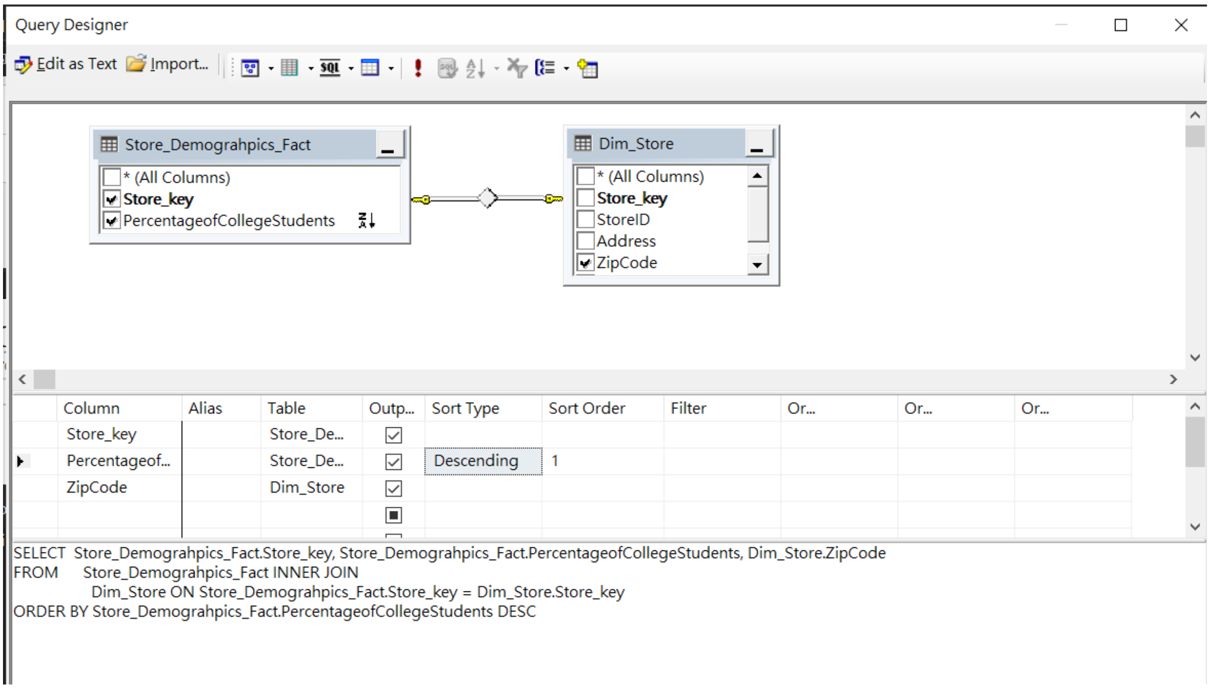
DataSource1

Type:

Microsoft SQL Server

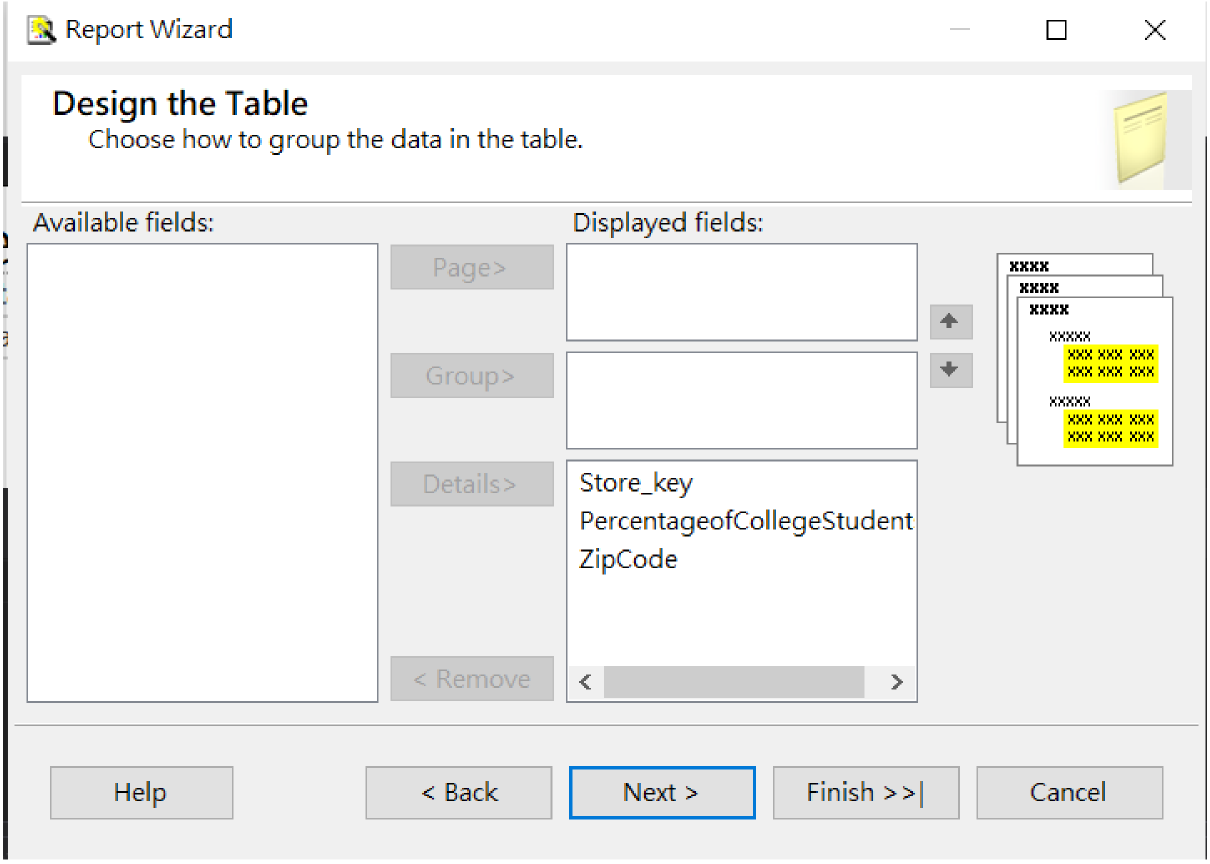
Connection string:

Data Source=infodata16.mbs.tamu.edu;Initial Catalog=600_group5_datawarehouse



Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...	Or...
Store_key		Store_De...	☑						
Percentageof...		Store_De...	☑	Descending	1				
ZipCode		Dim_Store	☑						
			▣						

```
SELECT  Store_Demograhpics_Fact.Store_key, Store_Demograhpics_Fact.PercentageofCollegeStudents, Dim_Store.ZipCode
FROM     Store_Demograhpics_Fact INNER JOIN
            Dim_Store ON Store_Demograhpics_Fact.Store_key = Dim_Store.Store_key
ORDER BY Store_Demograhpics_Fact.PercentageofCollegeStudents DESC
```



Report Wizard

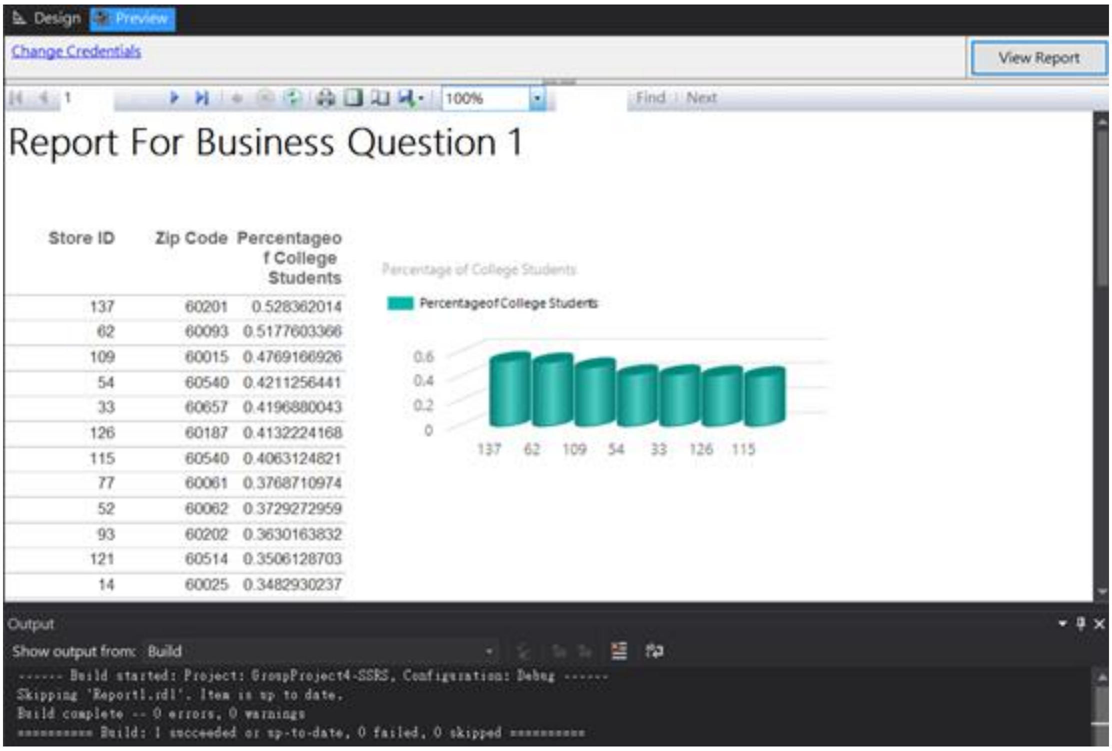
Design the Table

Choose how to group the data in the table.

Available fields:

Displayed fields:

Store_key
PercentageofCollegeStudent
ZipCode

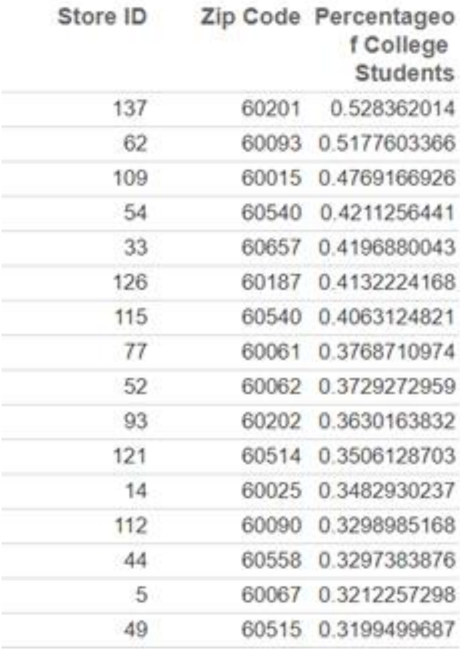


Report For Business Question 1

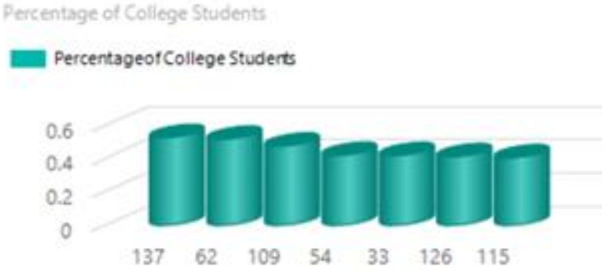
Store ID	Zip Code	Percentageo f College Students
137	60201	0.528362014
62	60093	0.5177603366
109	60015	0.4769166926
54	60540	0.4211256441
33	60657	0.4196880043
126	60187	0.4132224168
115	60540	0.4063124821
77	60061	0.3768710974
52	60062	0.3729272959
93	60202	0.3630163832
121	60514	0.3506128703
14	60025	0.3482930237

```
------ Build started: Project: GroupProject4-SSRS, Configuration: Debug ------
Skipping 'Report1.rdl'. Item is up to date.
Build complete -- 0 errors, 0 warnings
========== Build: 1 succeeded or up-to-date, 0 failed, 0 skipped ==========
```

Report For Business Question 10

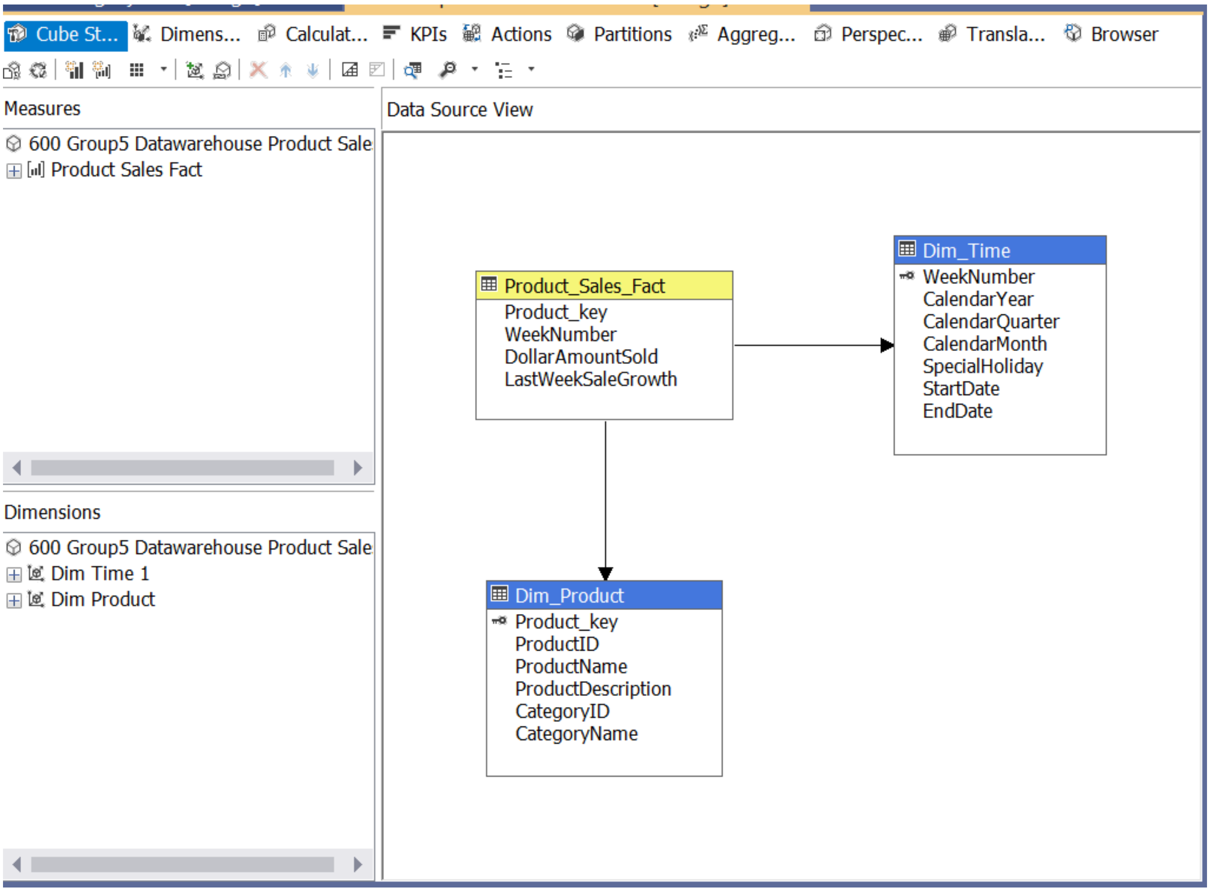


Store ID	Zip Code	Percentageof College Students
137	60201	0.528362014
62	60093	0.5177603366
109	60015	0.4769166926
54	60540	0.4211256441
33	60657	0.4196880043
126	60187	0.4132224168
115	60540	0.4063124821
77	60061	0.3768710974
52	60062	0.3729272959
93	60202	0.3630163832
121	60514	0.3506128703
14	60025	0.3482930237
112	60090	0.3298985168
44	60558	0.3297383876
5	60067	0.3212257298
49	60515	0.3199499687



Cubes from SSAS

BQ3: What are the products that have shown slow or static growth?



Measures

- 600 Group5 Datawarehouse Product Sale
- Product Sales Fact

Dimensions

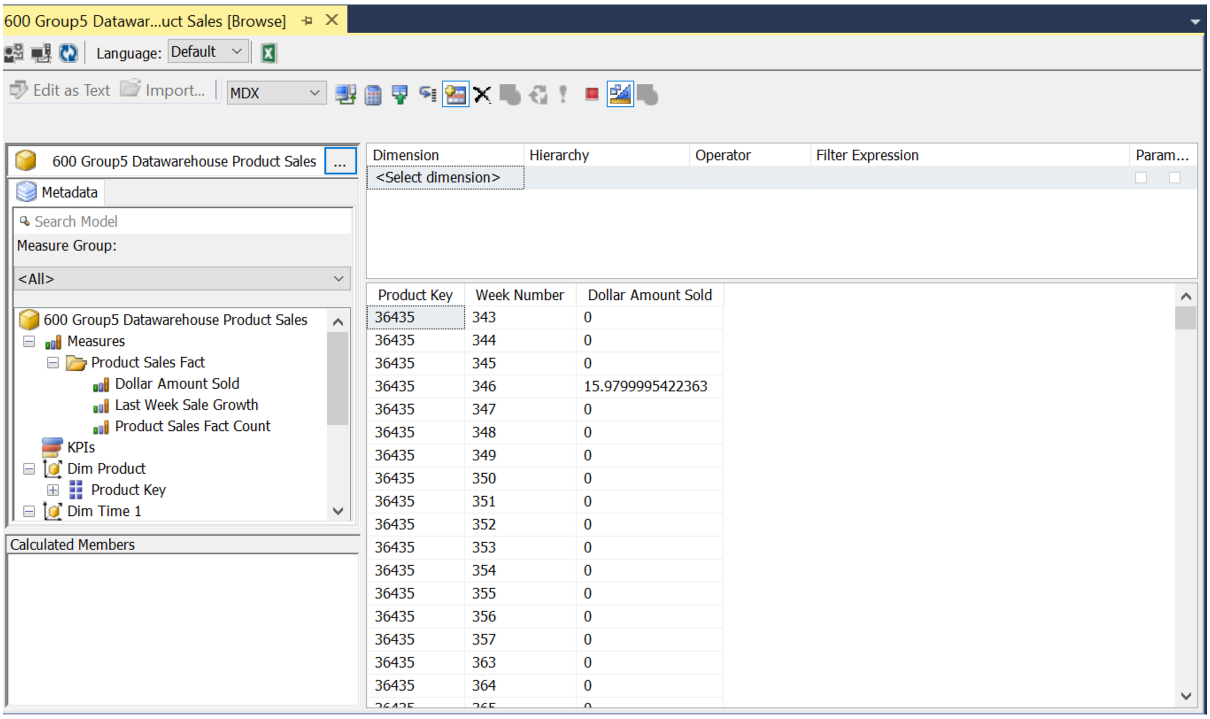
- 600 Group5 Datawarehouse Product Sale
- Dim Time 1
- Dim Product

Data Source View

Product_Sales_Fact: Product_key, WeekNumber, DollarAmountSold, LastWeekSaleGrowth

Dim_Time: WeekNumber, CalendarYear, CalendarQuarter, CalendarMonth, SpecialHoliday, StartDate, EndDate

Dim_Product: Product_key, ProductID, ProductName, ProductDescription, CategoryID, CategoryName



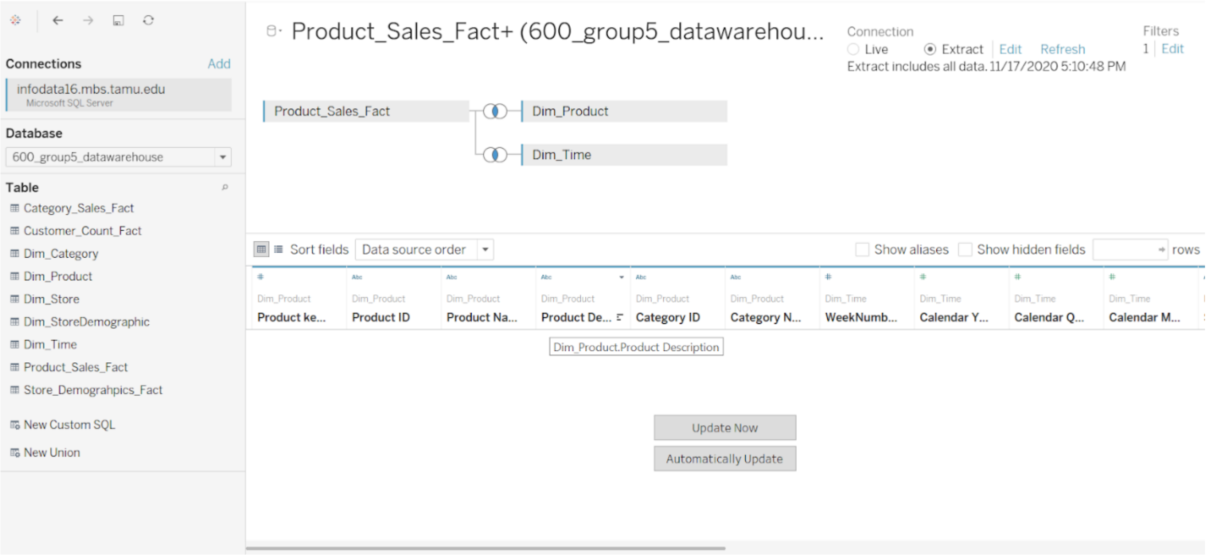
600 Group5 Datawarehouse Product Sales

- 600 Group5 Datawarehouse Product Sales
  - Measures
    - Product Sales Fact
      - Dollar Amount Sold
      - Last Week Sale Growth
      - Product Sales Fact Count
  - KPIs
  - Dim Product
    - Product Key
  - Dim Time 1

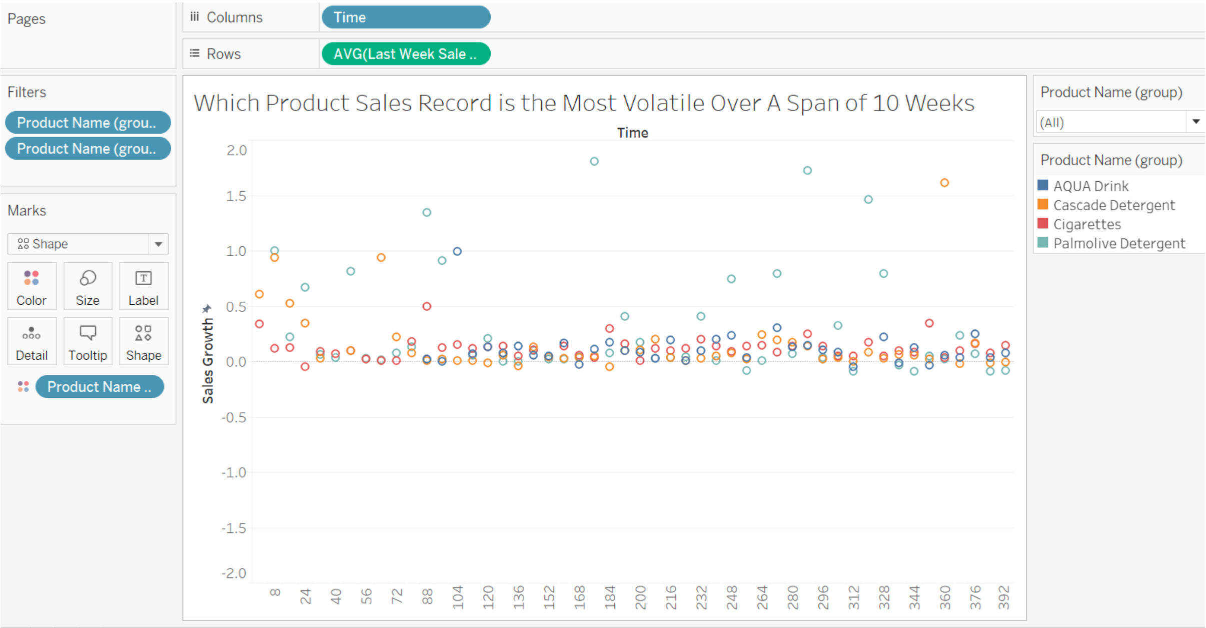
Product Key	Week Number	Dollar Amount Sold
36435	343	0
36435	344	0
36435	345	0
36435	346	15.9799995422363
36435	347	0
36435	348	0
36435	349	0
36435	350	0
36435	351	0
36435	352	0
36435	353	0
36435	354	0
36435	355	0
36435	356	0
36435	357	0
36435	363	0
36435	364	0

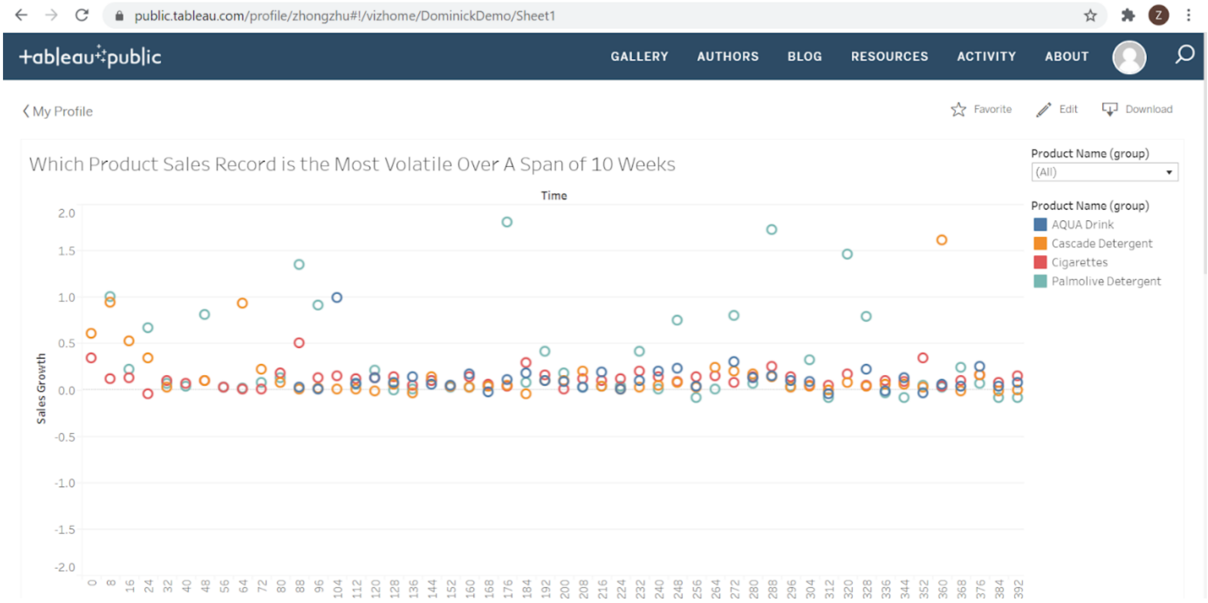
Reports using Tableau

BQ4: Which product sales record is the most volatile over a span of 40 weeks?



Extract includes all data. 11/17/2020 5:10:48 PM





Which Product Sales Record is the Most Volatile Over A Span of 10 Weeks

Product Name (group): (All)

Product Name (group)
- 🟦 AQUA Drink
- 🟧 Cascade Detergent
- 🟥 Cigarettes
- 🟩 Palmolive Detergent

REFERENCES

1. https://en.wikipedia.org/wiki/Dominick's
2. Rossi, Peter E., Robert E. McCulloch, and Greg M. Allenby, "The Value of Purchase History Data in Target Marketing," Marketing Science, 15 (1996): 321-340.
3. Dominicks-Manual-and-Codebook_KiltsCenter2013.pdf
4. “2018 retail, wholesale and distribution industry trends outlook.”
   Deloitte, https://www2.deloitte.com/us/en/pages/consumer-business/articles/retail-distribution industry-outlook.html. Accessed 28 February 2018.
5. Levy, Daniel, Dongwon Lee, Haipeng (Allan) Chen, Robert J. Kauffman, and Mark Bergen. “Price Points and Price Rigidity.” The Review of Economics and Statistics 93.4 (2011):1417-1431
6. Pauwels, Keon, "How Retailer and Competitor Decisions Drive the Long-term Effectiveness of Manufacturer Promotions for Fast Moving Consumer Goods," Journal of Retailing, Vol. 83 (March 2007): 297-308
7. Kamakura, Wagner A. and Wooseong Kang, "Chain-wide and Store-level Analysis for Cross category Management," Journal of Retailing, 83 (February 2007): 159-170.
8. Pofahl, Geoffrey M., Oral Capps Jr., and H. Alan Love, "Retail Zone Pricing and Simulated Price Effects of Upstream Mergers," International Journal of the Economics of Business, 13 (July 2006): 195215.
9. http://www.chicagobusiness.com/article/20131014/OPINION/131019936/why-dominicks-sputteredout.