CASE STUDY PROJECT

PHASE 2

TEAM 15

Team Members

Naveen Varma Patsamatla

Rishitha Ganagoni

Rhoda Alawiye

Phase 2 of the case study project comprised of the steps we followed to create a functional data warehouse design that generates the FACT and DIMENSION tables. We made use of ETL process to Extract/Transform/Load the required information into our data warehouse.

We performed the following data cleaning methods to address the variety of flaws that were in the original dataset with help of SSIS.

Cleaning Process:

- First, we Identified there are Duplicate records in the provided dataset and used **sort** function in ETL (SSIS) to remove those records and we ended up getting **74,111** records.
- The StoreState column consists of not only states but also some territories and one district. So, we separated the districts and territories and created two **new columns named "StoreDistrict" & "StoreTerritory"** in the "Drop_Duplicates" table in the ETL Package by using Copy Column function. The reason behind doing this to in future it is easy to identify the how many states, districts and territories does the company has stores.
- Updated the datatype of **"StoreZipcode"** to **integer** from **String** because some zip codes consist of 4 characters. Unless it is an integer it is hard to find the match. For example, let's say there is a zip code 07075 but we have that it as 7075 as a string in our database so it gives us a zero match when we try to fetch the records related to 07075. So, to avoid this complication we changed the datatype to integer.
- Changed the word **"Sweet Team-Large"** to **"Sweet Tea-Large"** in MenuName Column.
- There is a repetition of sandwich word as sandwiches in **MenuName** column, so we changed **"Sandwiches"** to **"Sandwich"**.
- There is a repetition of Soups & Salads word as Soup & Salad in **MenuName** column, so we changed **"Soup & Salad"** to **"Soups & Salads"**.
- There is a repetition of Side word as Sides in **MenuType** column, so we changed **"Sides"** to **"Side"**.

- Changed one of the **“StoreAddress”** row from **Lower case to Upper case** in one of the Derived Columns for better view to the user.
- Altered some rows of **ProductDesc from 0 to “No Description Available”** for Drinks Category.
- Replaced the special character **“?”** with **space** in **ProductDesc** in one of the Derived Columns.
- Updated some of the rows in “Drop_Duplicates” table as **“Not a District”** where **StoreDistrict** is not “District of Columbia”
- Updated some of the rows in “Drop_Duplicates” table as **“Not a State”** where **StoreSate** is equal to “District of Columbia” & “Guam” & “Virgin Islands of the U.S.” & “Puerto rico”
- Updated some of the rows in “Drop_Duplicates” table as “**Not a Territory**” where **StoreTerritory** is not “Guam” & “Virgin Islands of the U.S.” & “Puerto rico”
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 2000 && MenuCategory == ”Entrees”) and it is named as Split by 2000. And applied Derived Column to replace the **“Entrees”** with the **“Breakfast”** because, **there are two different MenuCategory named Entrees & Breakfast** to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 2200 && MenuCategory == ”Entrees”) and it is named as Split by 2200. And applied Derived Column to replace the **“Entrees”** with the **“Breakfast”** because**, there are two different MenuCategory named Entrees & Breakfast also replaced some of the rows in MenuSubCategory from “0” to “Chicken”** to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 2400 && MenuCategory ==” Entrees”) and it is named as Split by 2400. And applied Derived Column to replace the **“Entrees”** with the **“Breakfast”** because, **there are two different MenuCategory named Entrees & Breakfast,** replaced few rows of **MenuSubCategory from “0” to “Veggie”** and few rows **in MenuType from “Soup & Salad” to “Breakfast”** to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 3002 && MenuSubCategory == ”0”) and it is named as Split by 3002. And applied Derived

Column to **replace the “0” with the “Veggie”** because, there are two different MenuSubCategory named 0 & Veggie because to avoid duplication.

- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 3300 && MenuSubCategory == ”Sandwich”) and it is named as Split by 3300. And applied Derived Column to **replace the “Sandwich” with the “Fish”** because, there are two different MenuSubCategory named Sandwich & Fish because to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 3600 && MenuSubCategory ==” Chicken”) and it is named as Split by 3600. And applied Derived Column to **replace the “Chicken” with the “Beef”** because, there are two different MenuSubCategory named Chicken & Beef because to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 3601 && MenuSubCategory == ”Salad”) and it is named as Split by 3601. And applied Derived Column to **replace the “Salad” with the “Soups & Salads”** because, there are different MenuSubCategory named Salad & ‘Soups & Salads’ because to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 3501 && MenuSubCategory == ”0”) and it is named as Split by 3501. Used a Derived Column to **replace the “0” with the “Kids”** because, there are different MenuSubCategory named 0 & Kids because to avoid duplication.
- Used a Conditional Split in ETL to split on a condition i.e., (ProductNbr == 6002 && MenuSubCategory == ”0”) and it is named as Split by 6002. And applied Derived Column to **replace the “0” with the “Beef”** because, there are different MenuSubCategory named 0 & Beef because to avoid duplication.
- Removed an **extra space in “Fried Fish Sandwich”** which is present in **MenuName.**
- Changed the **Null in BuildingType to “Not Known”** in one of the Derived Columns.

The next thing after finally achieving a clean dataset was to create our FACT and DIMENSION tables.

Datawarehouse Design:

- We have created 5 Dimension tables and 1 Fact Table. We have segregated the attributed to get clear idea for the user about the data and that can be used for the future purpose also.
- The Dimension and Fact Tables are:

Dimension Tables	**Fact Table**
➢ DIM_LOCATION	FACT_SALES
➢ DIM_MENU	
➢ DIM_STORE	
➢ DIM_PRODUCT	
➢ DIM_DATE	

Attributes in each Dimension Table and Fact Table

- **DIM_LOCATION**
    - Location_ID
    - StoreAddress
    - StoreCity
    - StoreDistrict
    - StoreState
    - StoreTerritory
    - StoreZipcode

- **DIM_MENU**
    - Menu_ID
    - MenuName
    - MenuCategory
    - MenuSubCategory
    - MenuType

- **MENU_STORE**
    - Store_ID
    - StoreNbr
    - StoreCapacity
    - StoreStatus
    - BuildingType
    - NbrDriveThruLanes
    - NbrParkingSpaces

- **DIM_PRODUCT**
    - ProductNbr
    - ProductDesc
    - MenuName

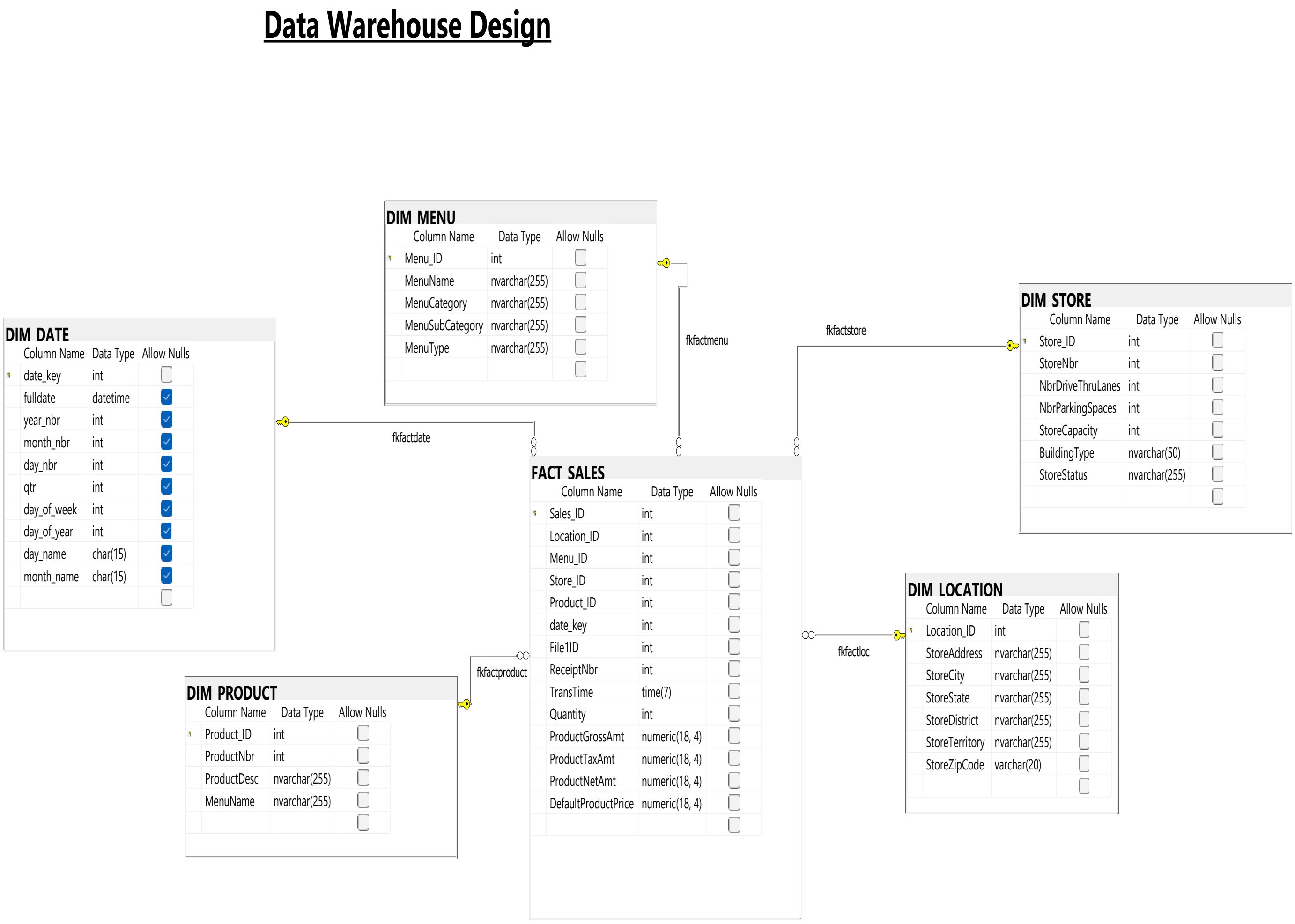
- **DIM_DATE**
    - date_key
    - fulldate
    - year_nbr
    - month_nbr
    - day_nbr
    - qtr
    - day_of_week
    - day_of_year
    - day_name
    - month_name

- **FACT_SALES**
    - Sales_ID
    - Location_ID
    - Menu_ID

    - Store_ID
    - Product_ID
    - date_key
    - File1ID
    - ReceiptNbr
    - TransDate
    - TransTime
    - ProductGrossAmt
    - Quantity
    - ProductNetAmt
    - ProductTaxAmt
    - DefaultProductPrice

- For all the attributes in the Dimension and Fact Tables we have changed to not null.
- Loaded the cleaned rows present in DROP_DUPLICATES table into each Dimension Table.
- Used Lookup to give data to the foreign keys present in the fact table considering DROP_DUPLICATES as my input table and Dimension tables as my reference table.

Datawarehouse Design Diagram:



Data Warehouse Design

DIM MENU

Column Name	Data Type	Allow Nulls
Menu_ID	int	□
MenuName	nvarchar(255)	□
MenuCategory	nvarchar(255)	□
MenuSubCategory	nvarchar(255)	□
MenuType	nvarchar(255)	□

DIM DATE

Column Name	Data Type	Allow Nulls
date_key	int	□
fulldate	datetime	☑
year_nbr	int	☑
month_nbr	int	☑
day_nbr	int	☑
qtr	int	☑
day_of_week	int	☑
day_of_year	int	☑
day_name	char(15)	☑
month_name	char(15)	☑

DIM STORE

Column Name	Data Type	Allow Nulls
Store_ID	int	□
StoreNbr	int	□
NbrDriveThruLanes	int	□
NbrParkingSpaces	int	□
StoreCapacity	int	□
BuildingType	nvarchar(50)	□
StoreStatus	nvarchar(255)	□

FACT SALES

Column Name	Data Type	Allow Nulls
Sales_ID	int	□
Location_ID	int	□
Menu_ID	int	□
Store_ID	int	□
Product_ID	int	□
date_key	int	□
File1ID	int	□
ReceiptNbr	int	□
TransTime	time(7)	□
Quantity	int	□
ProductGrossAmt	numeric(18, 4)	□
ProductTaxAmt	numeric(18, 4)	□
ProductNetAmt	numeric(18, 4)	□
DefaultProductPrice	numeric(18, 4)	□

DIM LOCATION

Column Name	Data Type	Allow Nulls
Location_ID	int	□
StoreAddress	nvarchar(255)	□
StoreCity	nvarchar(255)	□
StoreState	nvarchar(255)	□
StoreDistrict	nvarchar(255)	□
StoreTerritory	nvarchar(255)	□
StoreZipCode	varchar(20)	□

DIM PRODUCT

Column Name	Data Type	Allow Nulls
Product_ID	int	□
ProductNbr	int	□
ProductDesc	nvarchar(255)	□
MenuName	nvarchar(255)	□

Queries:

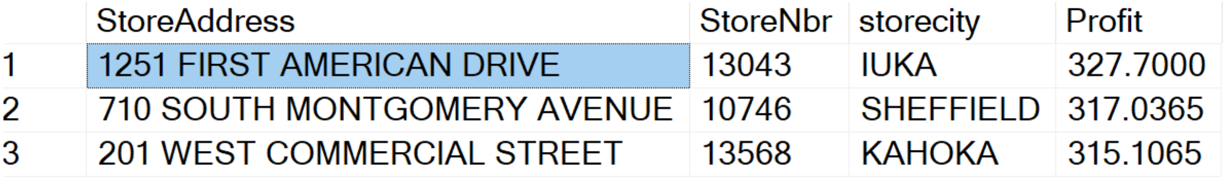
1. **What are the top 3 locations with respect to profit annually, monthly, weekly and daily?**

Profit Annually

```
CREATE VIEW QUERY1A AS
select top 3 StoreAddress,StoreNbr,storecity,sum(ProductNetAmt) as Profit from
FACT_SALES join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where year_nbr = 2022
group by StoreAddress,StoreNbr,StoreCity
order by profit desc
GO
```

Input given to the Query for Testing : year_nbr = 2022

Output:



	StoreAddress	StoreNbr	storecity	Profit
1	1251 FIRST AMERICAN DRIVE	13043	IUKA	327.7000
2	710 SOUTH MONTGOMERY AVENUE	10746	SHEFFIELD	317.0365
3	201 WEST COMMERCIAL STREET	13568	KAHOKA	315.1065

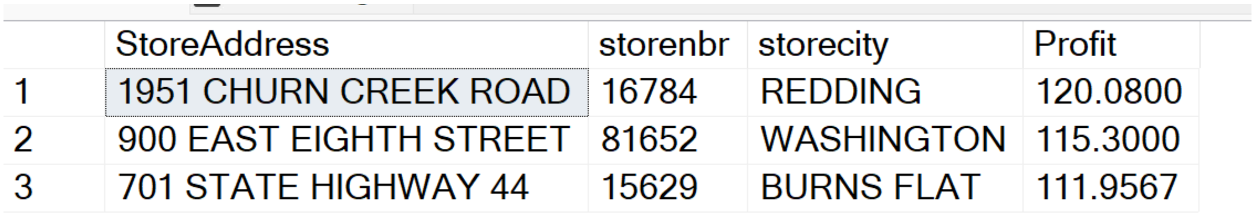
Profit Monthly

```
CREATE VIEW QUERY1B AS
select top 3 StoreAddress,storenbr,storecity,sum(ProductNetAmt) as Profit from
FACT_SALES join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where year_nbr = 2022 and month_nbr =4
```

```
group by StoreAddress,StoreNbr,StoreCity
order by profit desc
GO
```

Input given to the Query for Testing : year_nbr = 2022 and month_nbr =4

Output:



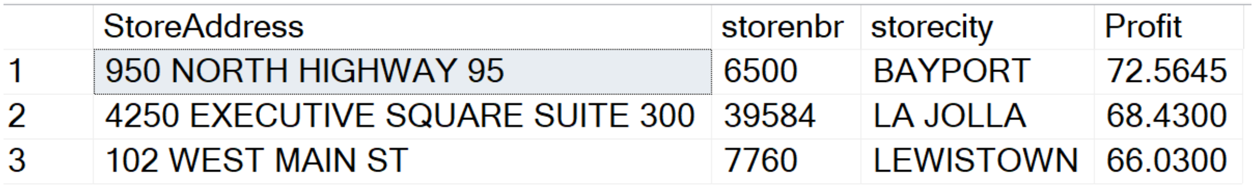
	StoreAddress	storenbr	storecity	Profit
1	1951 CHURN CREEK ROAD	16784	REDDING	120.0800
2	900 EAST EIGHTH STREET	81652	WASHINGTON	115.3000
3	701 STATE HIGHWAY 44	15629	BURNS FLAT	111.9567

Profit Week

```
CREATE VIEW QUERY1C AS
select top 3 StoreAddress,storenbr,storecity,sum(ProductNetAmt) as Profit from
FACT_SALES join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                    join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                    join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                    join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where CEILING(DAY(fulldate)/7.0) = 1 and  month_nbr = 1 and year_nbr = 2022
group by StoreAddress,storenbr,storecity
order by profit desc
GO
```

Input given to the Query for Testing : Week of the month = 1 and month_nbr = 1 and year_nbr = 2022

Output:



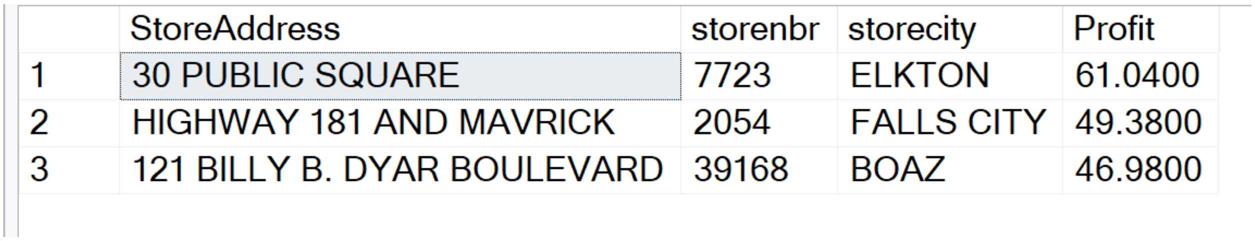
	StoreAddress	storenbr	storecity	Profit
1	950 NORTH HIGHWAY 95	6500	BAYPORT	72.5645
2	4250 EXECUTIVE SQUARE SUITE 300	39584	LA JOLLA	68.4300
3	102 WEST MAIN ST	7760	LEWISTOWN	66.0300

Profit Daily

```
CREATE VIEW QUERY1D AS
select top 3 StoreAddress,storenbr,storecity,sum(ProductNetAmt) as Profit from
FACT_SALES join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where fulldate = '2022-03-31'
group by StoreAddress,storenbr,StoreCity
order by profit desc
GO
```

Input given to the Query for Testing : fulldate = '2022-03-31'

Output:



	StoreAddress	storenbr	storecity	Profit
1	30 PUBLIC SQUARE	7723	ELKTON	61.0400
2	HIGHWAY 181 AND MAVRICK	2054	FALLS CITY	49.3800
3	121 BILLY B. DYAR BOULEVARD	39168	BOAZ	46.9800

2. How many customers does each location serve annually, monthly, weekly and daily?

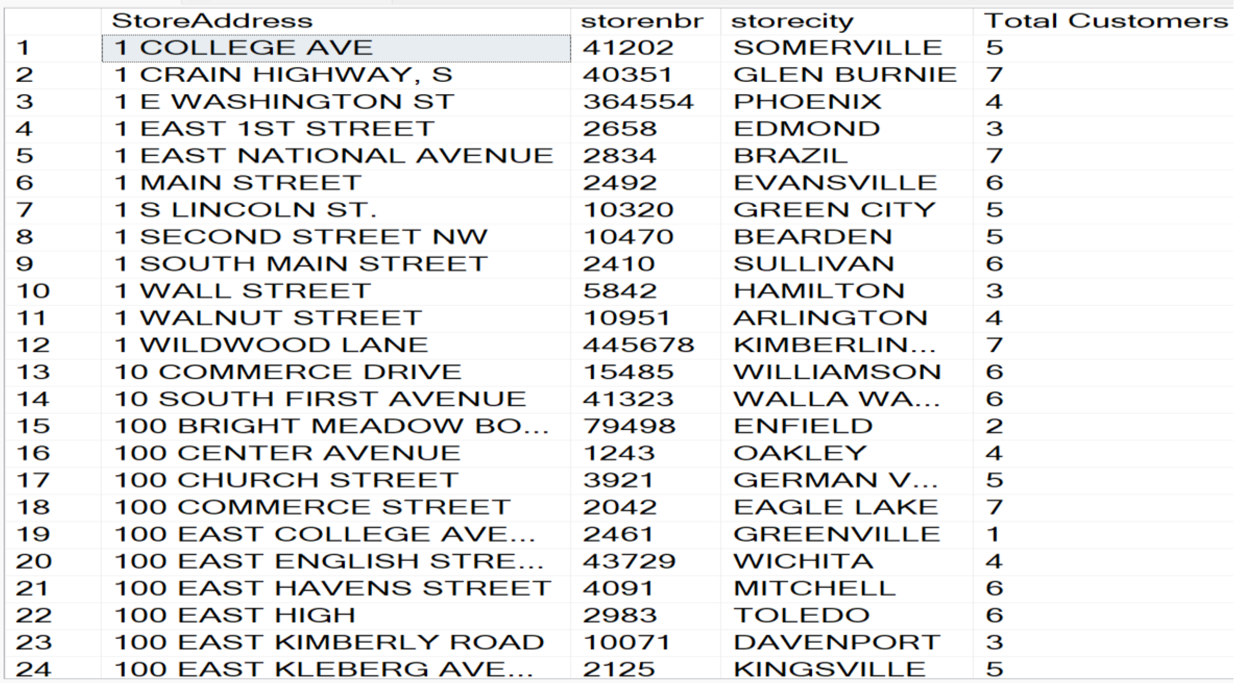
Customers Annually

```
CREATE VIEW QUERY2A AS
select StoreAddress,storenbr,storecity,count(distinct ReceiptNbr) as 'Total Customers' from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where year_nbr = 2022
```

```
group by StoreAddress,storenbr,storecity
GO
```

Input given to the query for testing: year_nbr = 2022

Output:



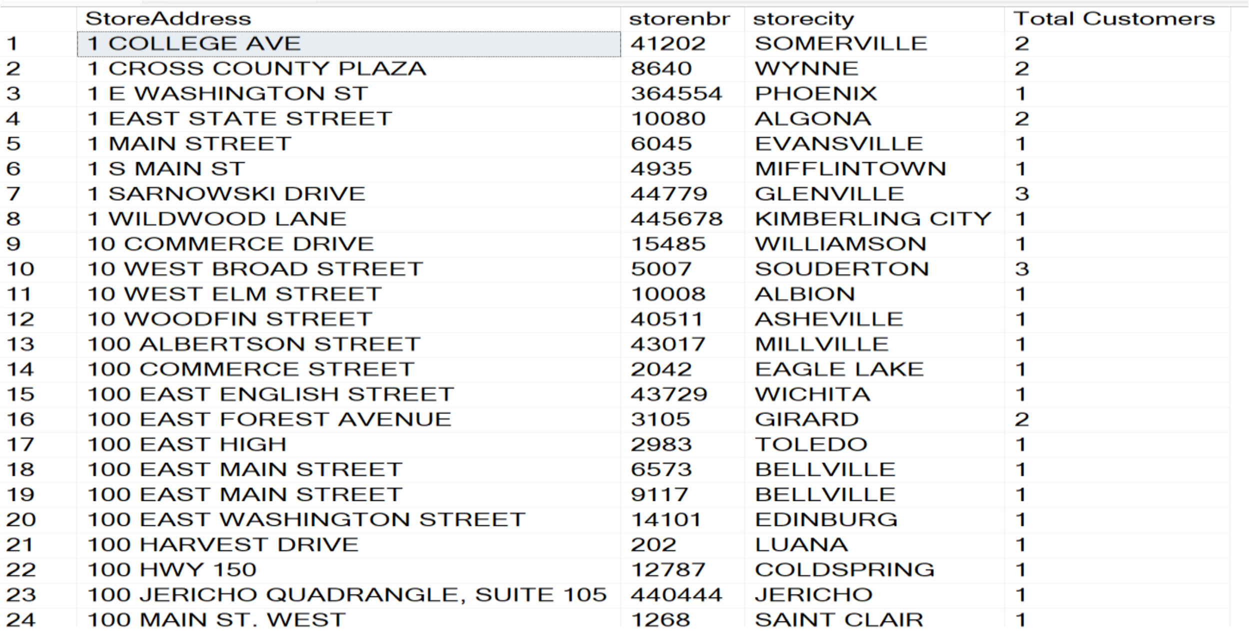
	StoreAddress	storenbr	storecity	Total Customers
1	1 COLLEGE AVE	41202	SOMERVILLE	5
2	1 CRAIN HIGHWAY, S	40351	GLEN BURNIE	7
3	1 E WASHINGTON ST	364554	PHOENIX	4
4	1 EAST 1ST STREET	2658	EDMOND	3
5	1 EAST NATIONAL AVENUE	2834	BRAZIL	7
6	1 MAIN STREET	2492	EVANSVILLE	6
7	1 S LINCOLN ST.	10320	GREEN CITY	5
8	1 SECOND STREET NW	10470	BEARDEN	5
9	1 SOUTH MAIN STREET	2410	SULLIVAN	6
10	1 WALL STREET	5842	HAMILTON	3
11	1 WALNUT STREET	10951	ARLINGTON	4
12	1 WILDWOOD LANE	445678	KIMBERLIN...	7
13	10 COMMERCE DRIVE	15485	WILLIAMSON	6
14	10 SOUTH FIRST AVENUE	41323	WALLA WA...	6
15	100 BRIGHT MEADOW BO...	79498	ENFIELD	2
16	100 CENTER AVENUE	1243	OAKLEY	4
17	100 CHURCH STREET	3921	GERMAN V...	5
18	100 COMMERCE STREET	2042	EAGLE LAKE	7
19	100 EAST COLLEGE AVE...	2461	GREENVILLE	1
20	100 EAST ENGLISH STRE...	43729	WICHITA	4
21	100 EAST HAVENS STREET	4091	MITCHELL	6
22	100 EAST HIGH	2983	TOLEDO	6
23	100 EAST KIMBERLY ROAD	10071	DAVENPORT	3
24	100 EAST KLEBERG AVE...	2125	KINGSVILLE	5

Customers Monthly

```
CREATE VIEW QUERY2B AS
select StoreAddress,storenbr,storecity,count(distinct ReceiptNbr) as 'Total Customers'from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where year_nbr = 2022 and month_nbr =1
group by StoreAddress,storenbr,storecity
GO
```

Input given to the query for testing: year_nbr = 2022 and month_nbr =1

Ouput:



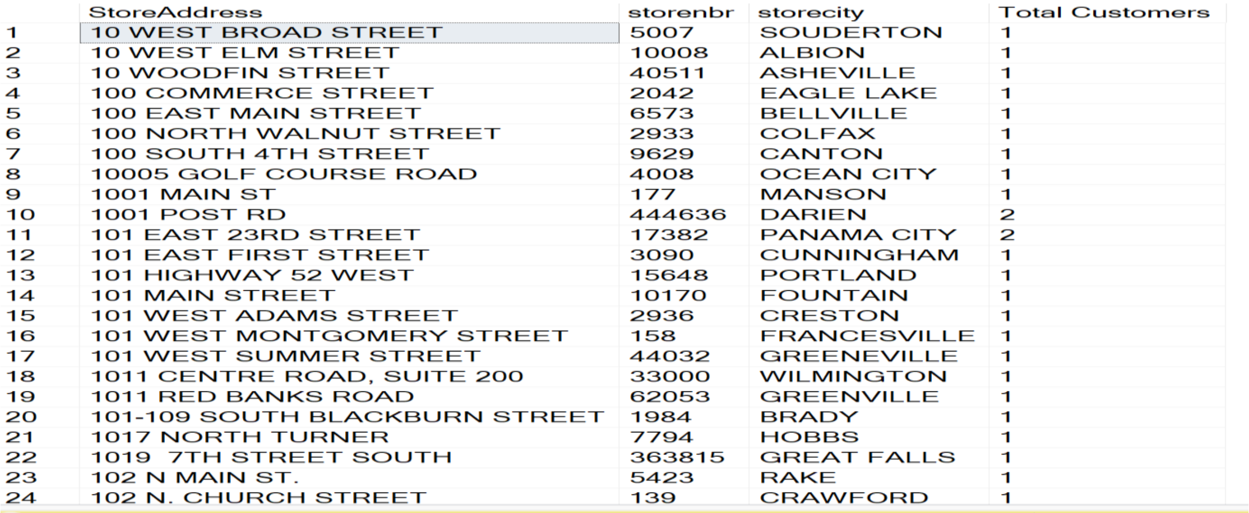
	StoreAddress	storenbr	storecity	Total Customers
1	1 COLLEGE AVE	41202	SOMERVILLE	2
2	1 CROSS COUNTY PLAZA	8640	WYNNE	2
3	1 E WASHINGTON ST	364554	PHOENIX	1
4	1 EAST STATE STREET	10080	ALGONA	2
5	1 MAIN STREET	6045	EVANSVILLE	1
6	1 S MAIN ST	4935	MIFFLINTOWN	1
7	1 SARNOWSKI DRIVE	44779	GLENVILLE	3
8	1 WILDWOOD LANE	445678	KIMBERLING CITY	1
9	10 COMMERCE DRIVE	15485	WILLIAMSON	1
10	10 WEST BROAD STREET	5007	SOUDERTON	3
11	10 WEST ELM STREET	10008	ALBION	1
12	10 WOODFIN STREET	40511	ASHEVILLE	1
13	100 ALBERTSON STREET	43017	MILLVILLE	1
14	100 COMMERCE STREET	2042	EAGLE LAKE	1
15	100 EAST ENGLISH STREET	43729	WICHITA	1
16	100 EAST FOREST AVENUE	3105	GIRARD	2
17	100 EAST HIGH	2983	TOLEDO	1
18	100 EAST MAIN STREET	6573	BELLVILLE	1
19	100 EAST MAIN STREET	9117	BELLVILLE	1
20	100 EAST WASHINGTON STREET	14101	EDINBURG	1
21	100 HARVEST DRIVE	202	LUANA	1
22	100 HWY 150	12787	COLDSPRING	1
23	100 JERICHO QUADRANGLE, SUITE 105	440444	JERICHO	1
24	100 MAIN ST. WEST	1268	SAINT CLAIR	1

Customers Weekly

```
CREATE VIEW QUERY2C AS
select StoreAddress,storenbr,storecity,count(distinct ReceiptNbr) as 'Total Customers' from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where CEILING(DAY(fulldate)/7.0) = 1 and  month_nbr = 1 and year_nbr = 2022
group by StoreAddress,storenbr,storecity
GO
```

Input given to the query for testing: Week of the month = 1 and month_nbr = 1 and year_nbr = 2022

Output:



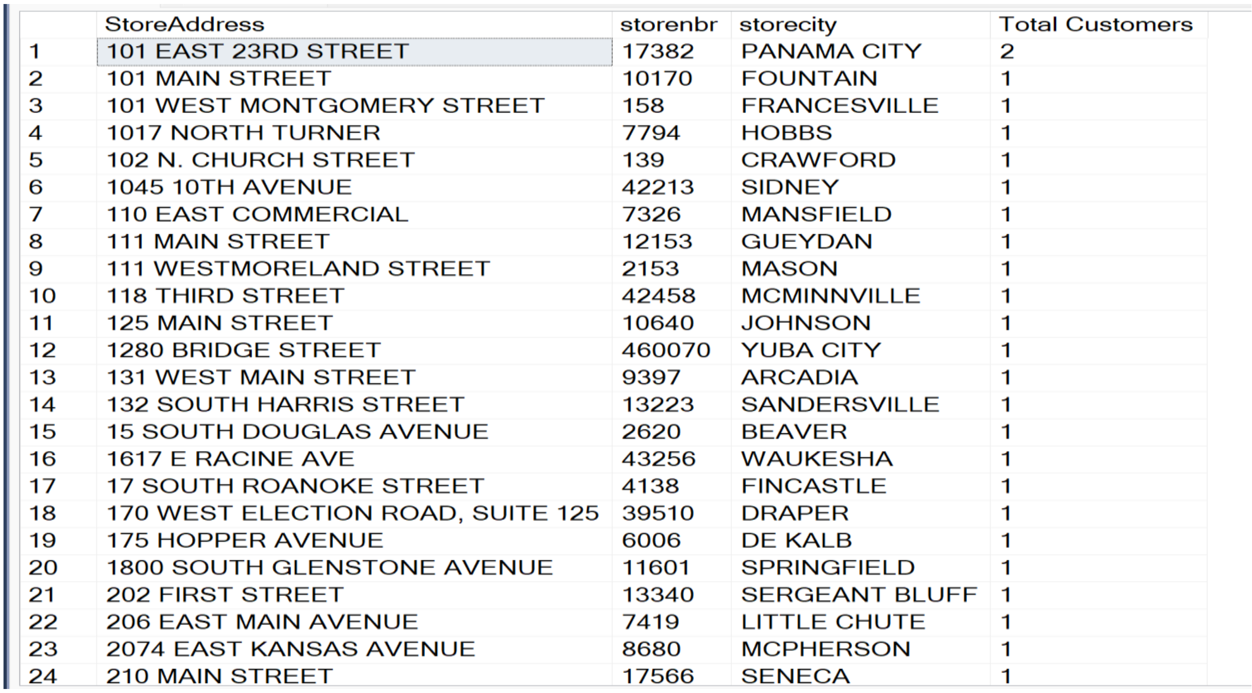
	StoreAddress	storenbr	storecity	Total Customers
1	10 WEST BROAD STREET	5007	SOUDERTON	1
2	10 WEST ELM STREET	10008	ALBION	1
3	10 WOODFIN STREET	40511	ASHEVILLE	1
4	100 COMMERCE STREET	2042	EAGLE LAKE	1
5	100 EAST MAIN STREET	6573	BELLVILLE	1
6	100 NORTH WALNUT STREET	2933	COLFAX	1
7	100 SOUTH 4TH STREET	9629	CANTON	1
8	10005 GOLF COURSE ROAD	4008	OCEAN CITY	1
9	1001 MAIN ST	177	MANSON	1
10	1001 POST RD	444636	DARIEN	2
11	101 EAST 23RD STREET	17382	PANAMA CITY	2
12	101 EAST FIRST STREET	3090	CUNNINGHAM	1
13	101 HIGHWAY 52 WEST	15648	PORTLAND	1
14	101 MAIN STREET	10170	FOUNTAIN	1
15	101 WEST ADAMS STREET	2936	CRESTON	1
16	101 WEST MONTGOMERY STREET	158	FRANCESVILLE	1
17	101 WEST SUMMER STREET	44032	GREENEVILLE	1
18	1011 CENTRE ROAD, SUITE 200	33000	WILMINGTON	1
19	1011 RED BANKS ROAD	62053	GREENVILLE	1
20	101-109 SOUTH BLACKBURN STREET	1984	BRADY	1
21	1017 NORTH TURNER	7794	HOBBS	1
22	1019 7TH STREET SOUTH	363815	GREAT FALLS	1
23	102 N MAIN ST.	5423	RAKE	1
24	102 N. CHURCH STREET	139	CRAWFORD	1

Customers daily

```
CREATE VIEW QUERY2D AS
select StoreAddress,storenbr,storecity,count(distinct ReceiptNbr) as 'Total Customers' from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                    join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                    join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where fulldate = '2022-01-07'
group by StoreAddress,storenbr,storecity
GO
```

Input given to the query for testing: fulldate = '2022-01-07'

Output:



	StoreAddress	storenbr	storecity	Total Customers
1	101 EAST 23RD STREET	17382	PANAMA CITY	2
2	101 MAIN STREET	10170	FOUNTAIN	1
3	101 WEST MONTGOMERY STREET	158	FRANCESVILLE	1
4	1017 NORTH TURNER	7794	HOBBS	1
5	102 N. CHURCH STREET	139	CRAWFORD	1
6	1045 10TH AVENUE	42213	SIDNEY	1
7	110 EAST COMMERCIAL	7326	MANSFIELD	1
8	111 MAIN STREET	12153	GUEYDAN	1
9	111 WESTMORELAND STREET	2153	MASON	1
10	118 THIRD STREET	42458	MCMINNVILLE	1
11	125 MAIN STREET	10640	JOHNSON	1
12	1280 BRIDGE STREET	460070	YUBA CITY	1
13	131 WEST MAIN STREET	9397	ARCADIA	1
14	132 SOUTH HARRIS STREET	13223	SANDERSVILLE	1
15	15 SOUTH DOUGLAS AVENUE	2620	BEAVER	1
16	1617 E RACINE AVE	43256	WAUKESHA	1
17	17 SOUTH ROANOKE STREET	4138	FINCASTLE	1
18	170 WEST ELECTION ROAD, SUITE 125	39510	DRAPER	1
19	175 HOPPER AVENUE	6006	DE KALB	1
20	1800 SOUTH GLENSTONE AVENUE	11601	SPRINGFIELD	1
21	202 FIRST STREET	13340	SERGEANT BLUFF	1
22	206 EAST MAIN AVENUE	7419	LITTLE CHUTE	1
23	2074 EAST KANSAS AVENUE	8680	MCPHERSON	1
24	210 MAIN STREET	17566	SENECA	1

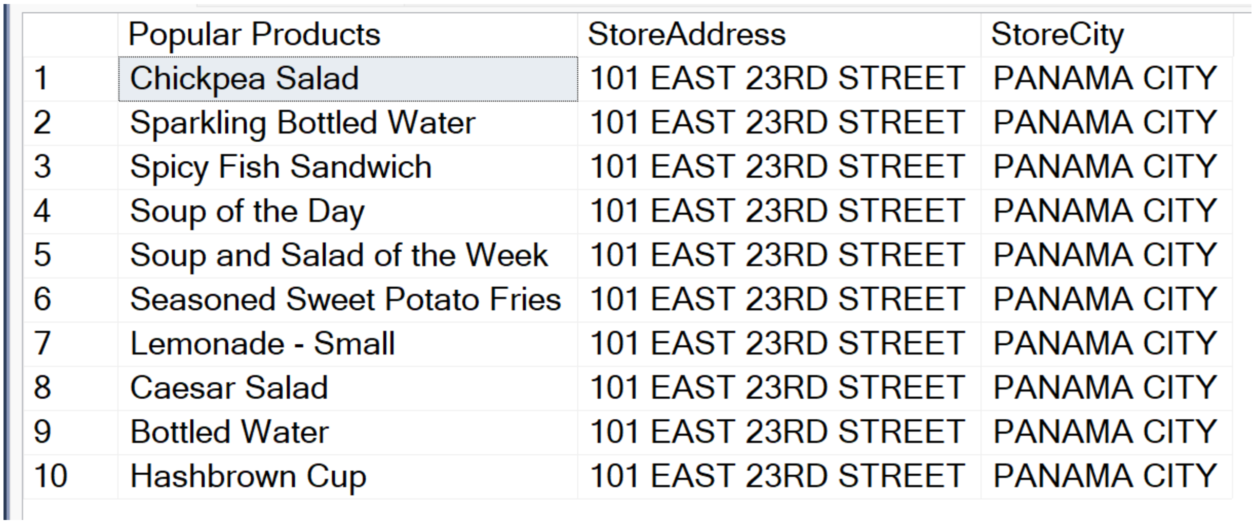
3. What are the top 10 most popular products sold by location annually, monthly, weekly and daily?

Top 10 most popular products sold Annually

```
CREATE VIEW QUERY3A AS
select top 10 (MenuName) as 'Popular Products' ,StoreAddress,StoreCity from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where year_nbr = 2022 and storenbr = 17382
group by MenuName,StoreAddress,StoreCity
order by sum(Quantity) desc
GO
```

Input given to the Query for testing: year_nbr = 2022 and storenbr = 17382

Output:



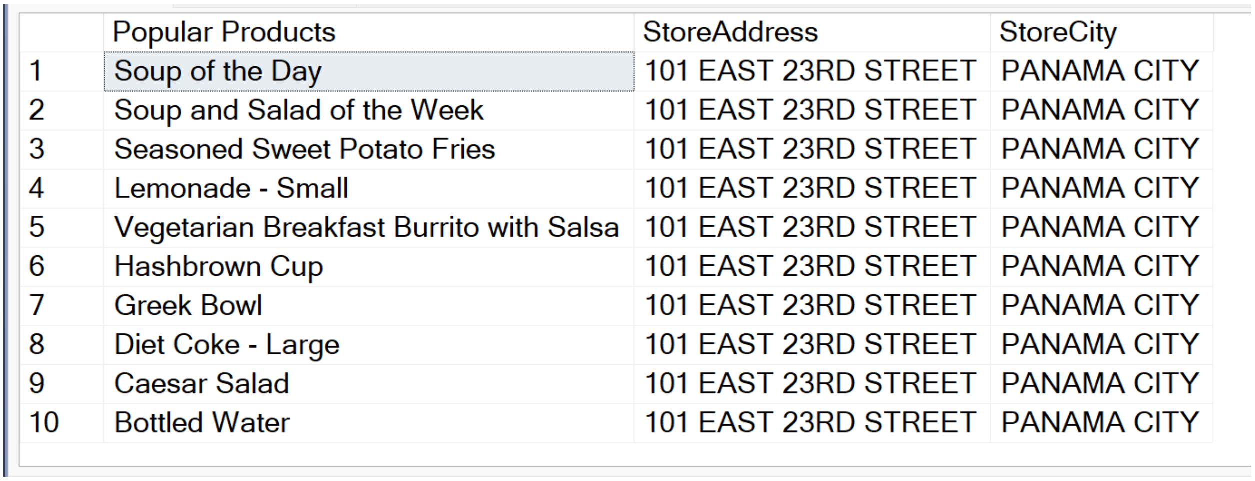
	Popular Products	StoreAddress	StoreCity
1	Chickpea Salad	101 EAST 23RD STREET	PANAMA CITY
2	Sparkling Bottled Water	101 EAST 23RD STREET	PANAMA CITY
3	Spicy Fish Sandwich	101 EAST 23RD STREET	PANAMA CITY
4	Soup of the Day	101 EAST 23RD STREET	PANAMA CITY
5	Soup and Salad of the Week	101 EAST 23RD STREET	PANAMA CITY
6	Seasoned Sweet Potato Fries	101 EAST 23RD STREET	PANAMA CITY
7	Lemonade - Small	101 EAST 23RD STREET	PANAMA CITY
8	Caesar Salad	101 EAST 23RD STREET	PANAMA CITY
9	Bottled Water	101 EAST 23RD STREET	PANAMA CITY
10	Hashbrown Cup	101 EAST 23RD STREET	PANAMA CITY

Top 10 most popular products sold Monthly

```
CREATE VIEW QUERY3B AS
select top 10 (MenuName) as 'Popular Products',StoreAddress,StoreCity from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where year_nbr = 2022 and storenbr = 17382 and month_nbr = 1
group by MenuName,StoreAddress,StoreCity
order by sum(Quantity) desc
GO
```

Input given to the Query for testing: year_nbr = 2022 and storenbr = 17382 and month_nbr = 1

Output:



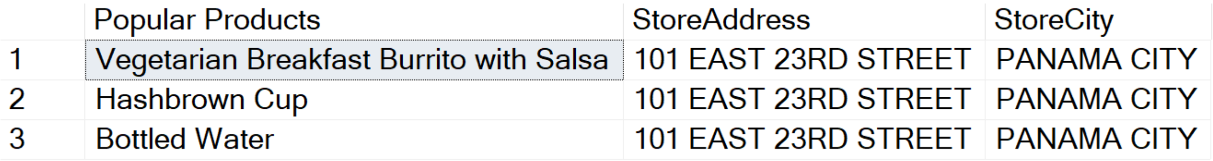
	Popular Products	StoreAddress	StoreCity
1	Soup of the Day	101 EAST 23RD STREET	PANAMA CITY
2	Soup and Salad of the Week	101 EAST 23RD STREET	PANAMA CITY
3	Seasoned Sweet Potato Fries	101 EAST 23RD STREET	PANAMA CITY
4	Lemonade - Small	101 EAST 23RD STREET	PANAMA CITY
5	Vegetarian Breakfast Burrito with Salsa	101 EAST 23RD STREET	PANAMA CITY
6	Hashbrown Cup	101 EAST 23RD STREET	PANAMA CITY
7	Greek Bowl	101 EAST 23RD STREET	PANAMA CITY
8	Diet Coke - Large	101 EAST 23RD STREET	PANAMA CITY
9	Caesar Salad	101 EAST 23RD STREET	PANAMA CITY
10	Bottled Water	101 EAST 23RD STREET	PANAMA CITY

Top 10 most popular products sold weekly

```
CREATE VIEW QUERY3C AS
select top 10 (MenuName) as 'Popular Products',StoreAddress,StoreCity  from
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where CEILING(DAY(fulldate)/7.0) = 3 and year_nbr = 2022 and storenbr = 17382 and
month_nbr = 1 and year_nbr = 2022
group by MenuName,StoreAddress,StoreCity
order by sum(Quantity) desc
GO
```

Input given to the Query for testing: week of the month= 3 and year_nbr = 2022 and storenbr = 17382 and month_nbr = 1 and year_nbr = 2022

Output:



	Popular Products	StoreAddress	StoreCity
1	Vegetarian Breakfast Burrito with Salsa	101 EAST 23RD STREET	PANAMA CITY
2	Hashbrown Cup	101 EAST 23RD STREET	PANAMA CITY
3	Bottled Water	101 EAST 23RD STREET	PANAMA CITY

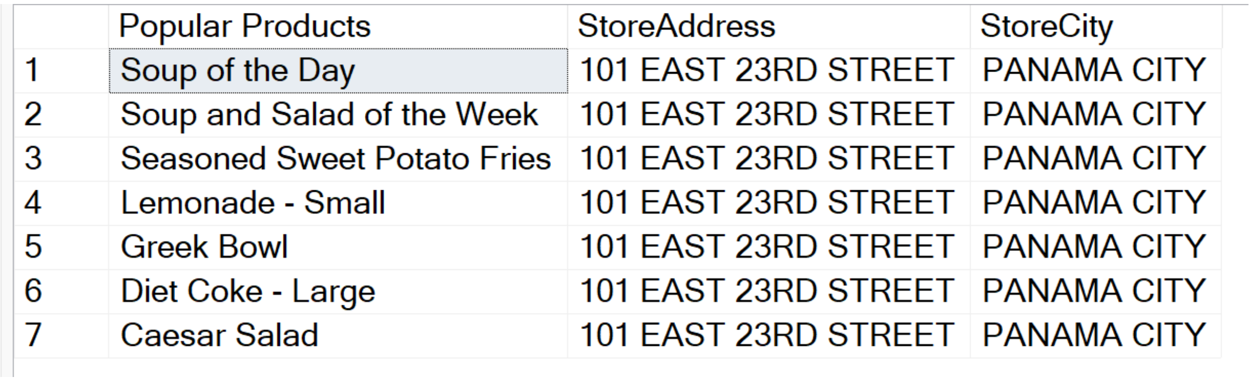
Top 10 most popular products sold daily

```
CREATE VIEW QUERY3D AS
select top 10 (MenuName) as 'Popular Products',StoreAddress,StoreCity  from
```

```
FACT_SALES join DIM_DATE on FACT_SALES.date_key = DIM_DATE.date_key
                        join DIM_LOCATION on FACT_SALES.Location_ID =
DIM_LOCATION.Location_ID
                        join DIM_PRODUCT on FACT_SALES.Product_ID =
DIM_PRODUCT.Product_ID
                        join DIM_store on FACT_SALES.Store_ID = DIM_STORE.Store_ID
where storenbr = 17382 and fulldate = '2022-1-07'
group by MenuName,StoreAddress,StoreCity
order by sum(Quantity) desc
GO
```

Input given to the Query for testing: storenbr = 17382 and fulldate = '2022-1-07'

Output:

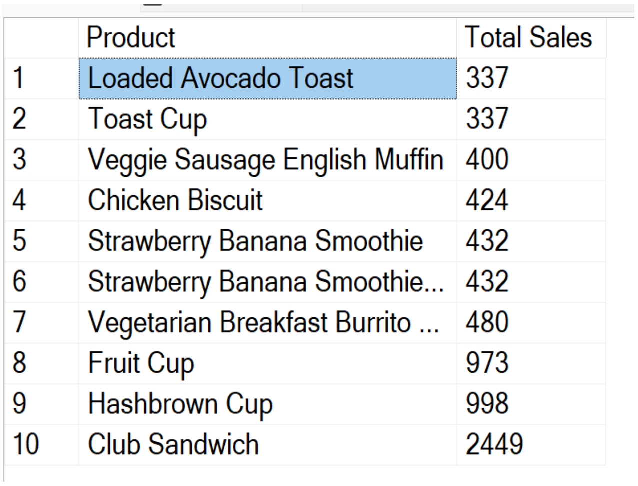


	Popular Products	StoreAddress	StoreCity
1	Soup of the Day	101 EAST 23RD STREET	PANAMA CITY
2	Soup and Salad of the Week	101 EAST 23RD STREET	PANAMA CITY
3	Seasoned Sweet Potato Fries	101 EAST 23RD STREET	PANAMA CITY
4	Lemonade - Small	101 EAST 23RD STREET	PANAMA CITY
5	Greek Bowl	101 EAST 23RD STREET	PANAMA CITY
6	Diet Coke - Large	101 EAST 23RD STREET	PANAMA CITY
7	Caesar Salad	101 EAST 23RD STREET	PANAMA CITY

4 .Which products have no sales or few sales?

```
CREATE VIEW QUERY4  AS
select top 10 MenuName as Product ,SUM(Quantity) as 'Total Sales' from
FACT_SALES
join DIM_PRODUCT on FACT_SALES.Product_ID =DIM_PRODUCT.Product_ID
group by DIM_PRODUCT.menuname
order by SUM(Quantity)
GO
```

Output:

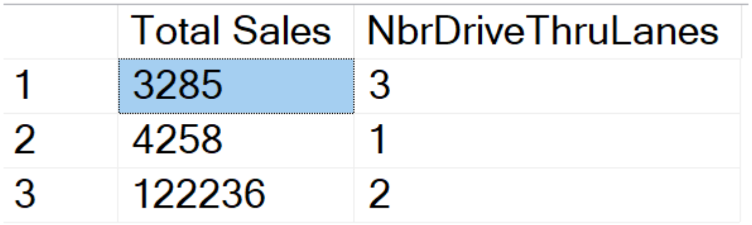


	Product	Total Sales
1	Loaded Avocado Toast	337
2	Toast Cup	337
3	Veggie Sausage English Muffin	400
4	Chicken Biscuit	424
5	Strawberry Banana Smoothie	432
6	Strawberry Banana Smoothie...	432
7	Vegetarian Breakfast Burrito ...	480
8	Fruit Cup	973
9	Hashbrown Cup	998
10	Club Sandwich	2449

5. How is the number of drive-thru lanes impacting sales?

```
CREATE VIEW QUERY5 AS
select SUM(Quantity) as 'Total Sales',NbrDriveThruLanes  from FACT_SALES
join DIM_STORE on FACT_SALES.Store_ID = DIM_STORE.Store_ID
group by  NbrDriveThruLanes
GO
```

Output:



	Total Sales	NbrDriveThruLanes
1	3285	3
2	4258	1
3	122236	2

Impact: As we can infer from the output that Two drive through lines has more impact on the sales than any other. Surprisingly Though there are 3 drive through lines most of the sales happened with 2 drive through lanes. May be even number of drive through lanes may effect the sales in positive way.