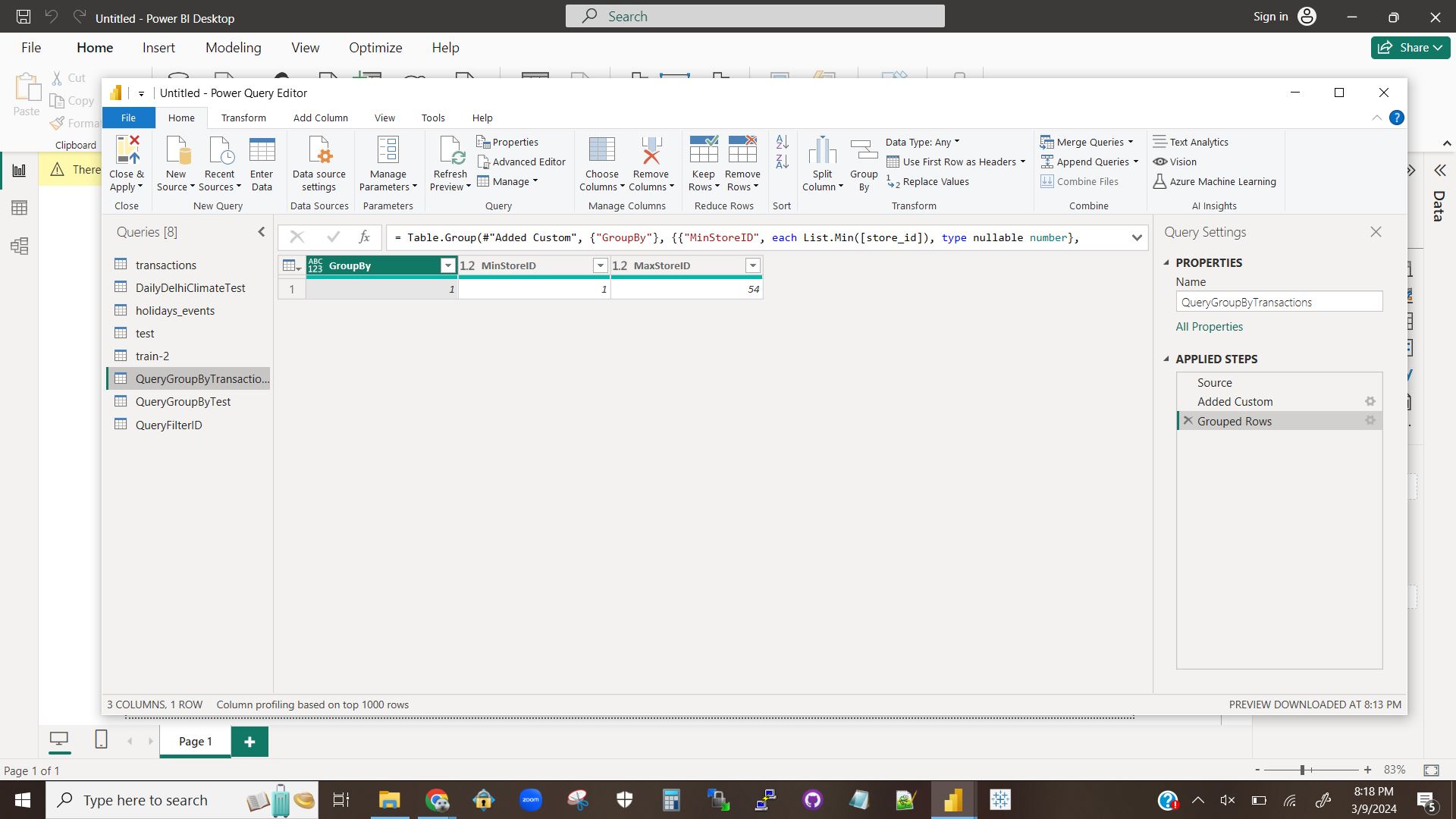
# Question 1

Renamed the store\_nbr column to store\_id.

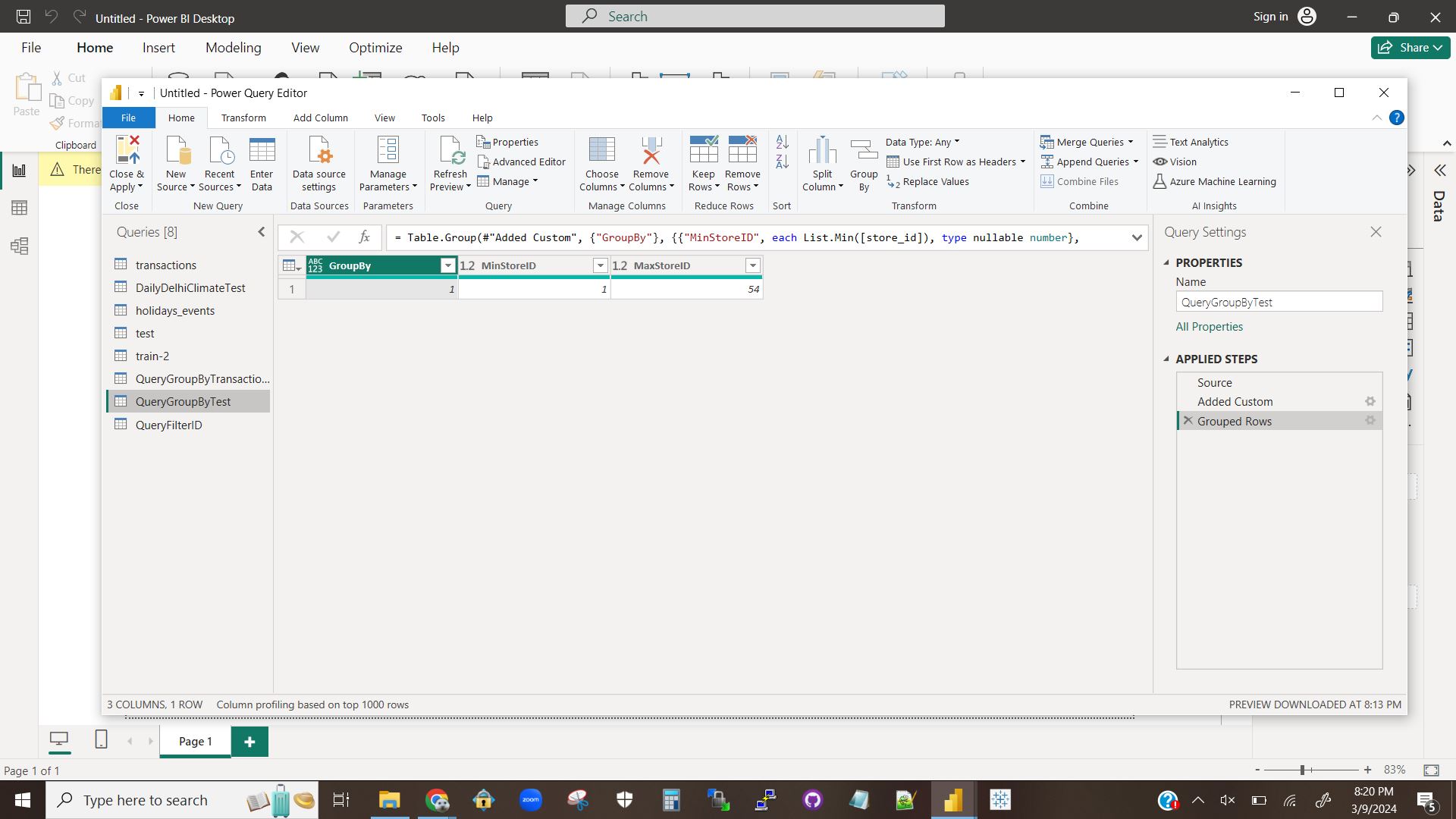
## Question 1.1

### Visualization 1.1.1



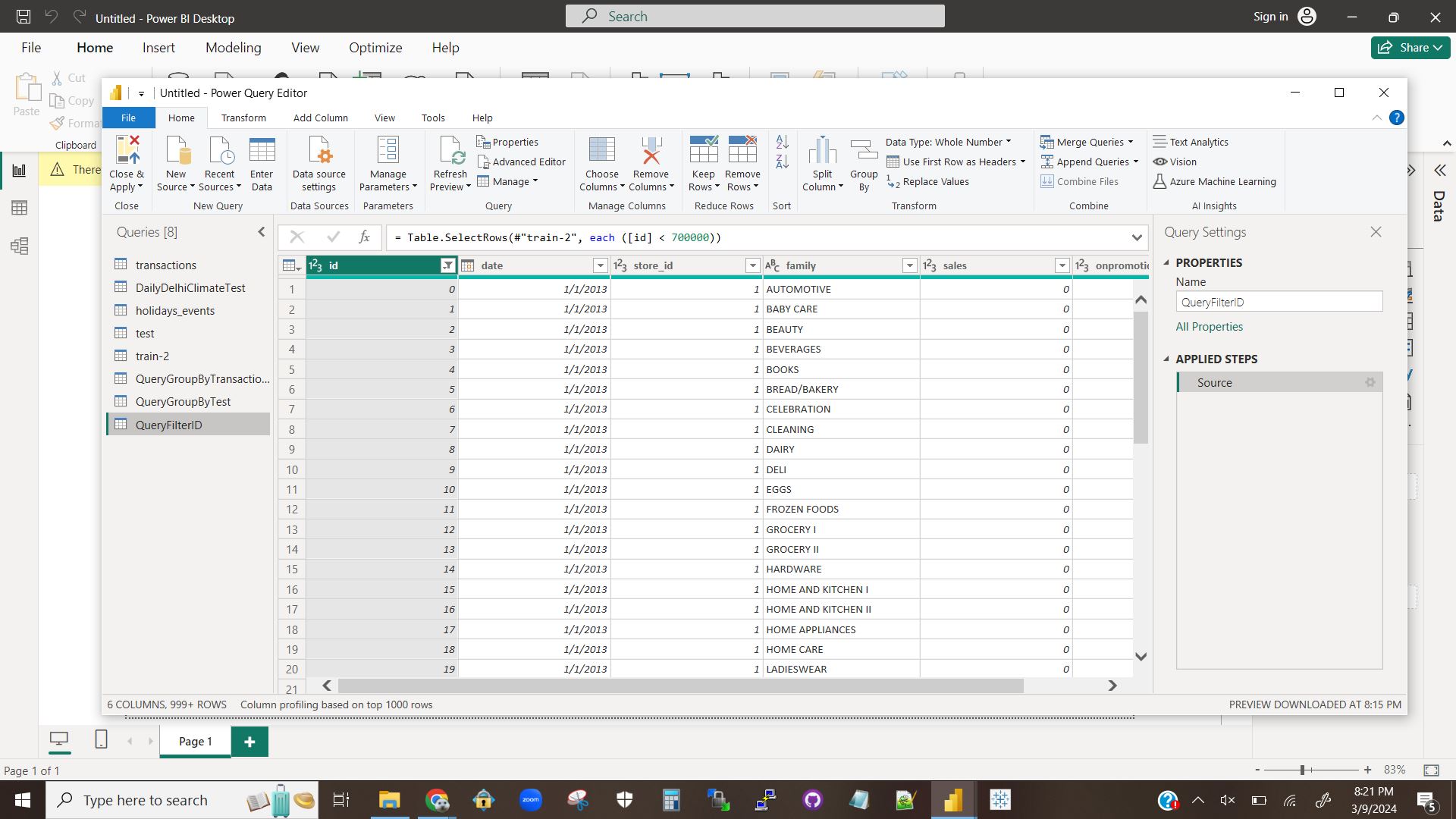
An image of the GroupBy query and results for finding the min and max store ID’s in the transactions table.

### Visualization 1.1.2



An image of the GroupBy query and results for finding the min and max store ID’s in the test table.

### Visualization 1.1.3



An image of the SelectRows query and results for dropping all rows that have a row ID of 700,000 or more.

### Explanation 1.1.1

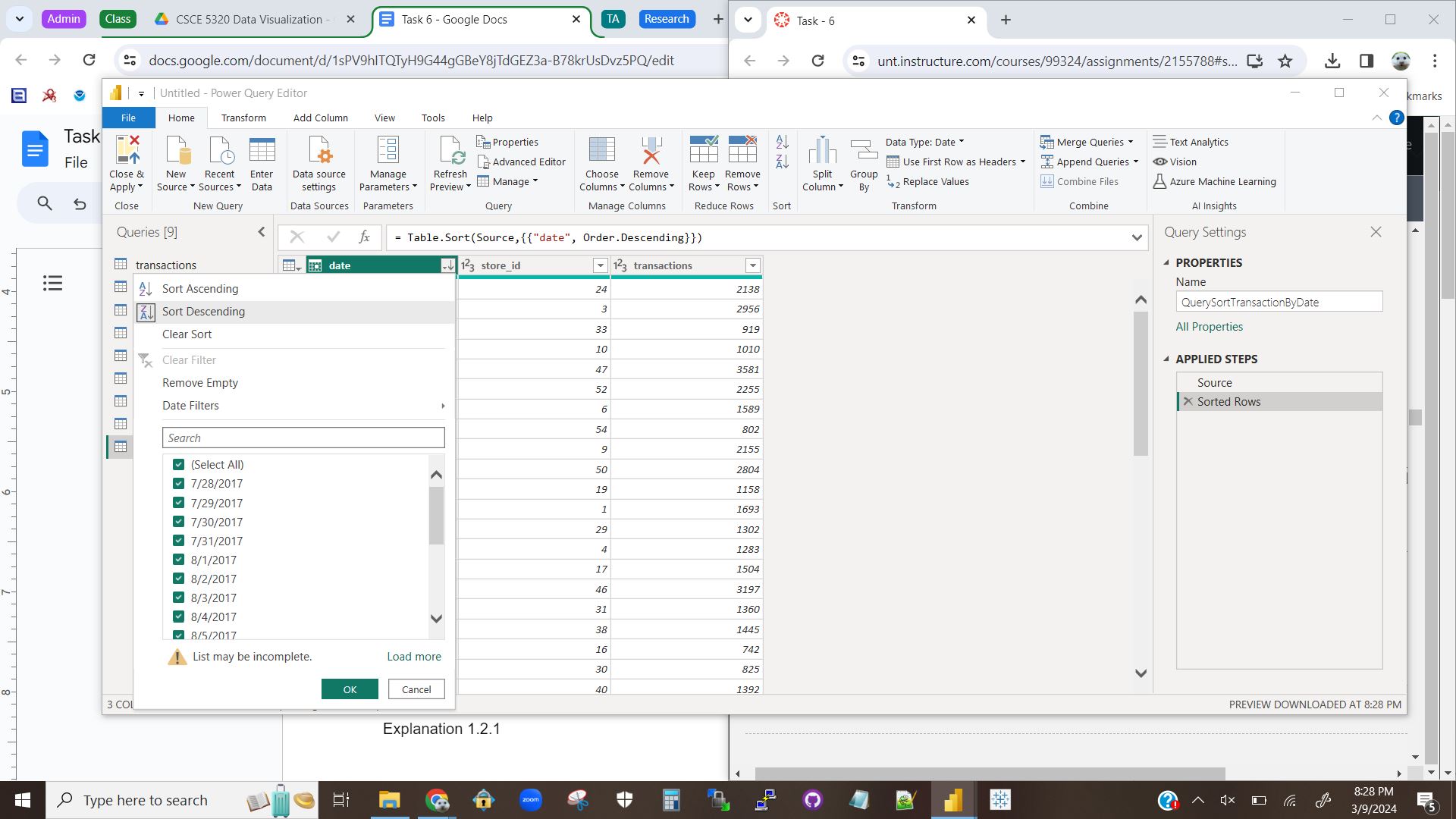
There were only two difficulties for this question.

First, the wording of the third instruction “Apply group by on any table and perform the same steps mentioned in the tutorial for any other table except train” is unclear on if there were supposed to be two separate queries, if GroupBy was supposed to be performed on two tables in one query, or if we were supposed to just not do GroupBy on the train-2 table. I have included two queries in this submission.

Second, writing the query on step 11 gave me a little bit of trouble as I am not very familiar with SQL. It was quickly resolved, though.

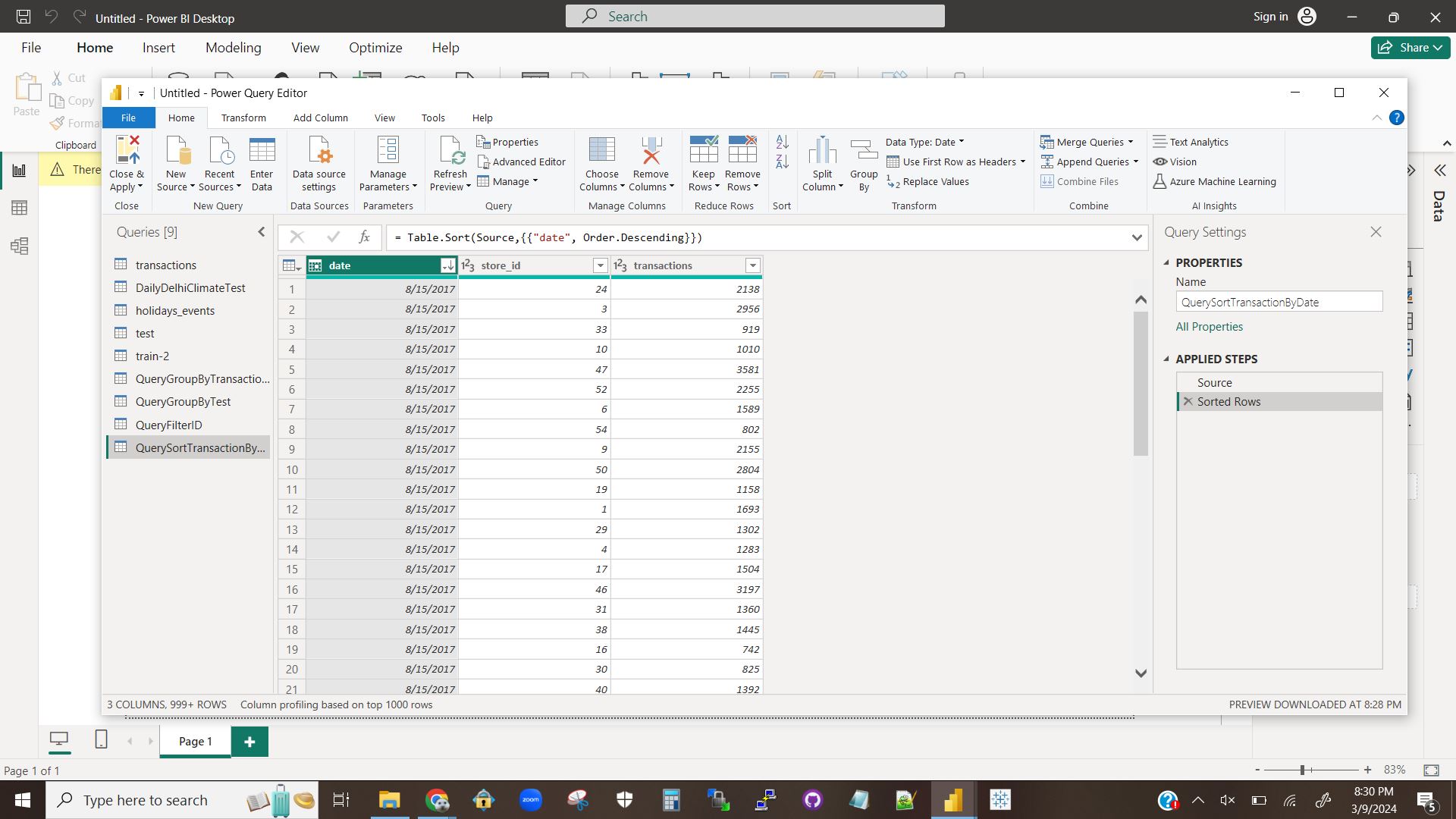
## Question 1.2

### Visualization 1.2.1



A screenshot showing the selection of the option to sort the date column of the transactions table in descending order.

### Visualization 1.2.2



A screenshot showing the date column of the transactions table sorted in descending order.

### Explanation 1.2.1

First I imported the transactions table by typing “transactions” into the text entry bar at the top.

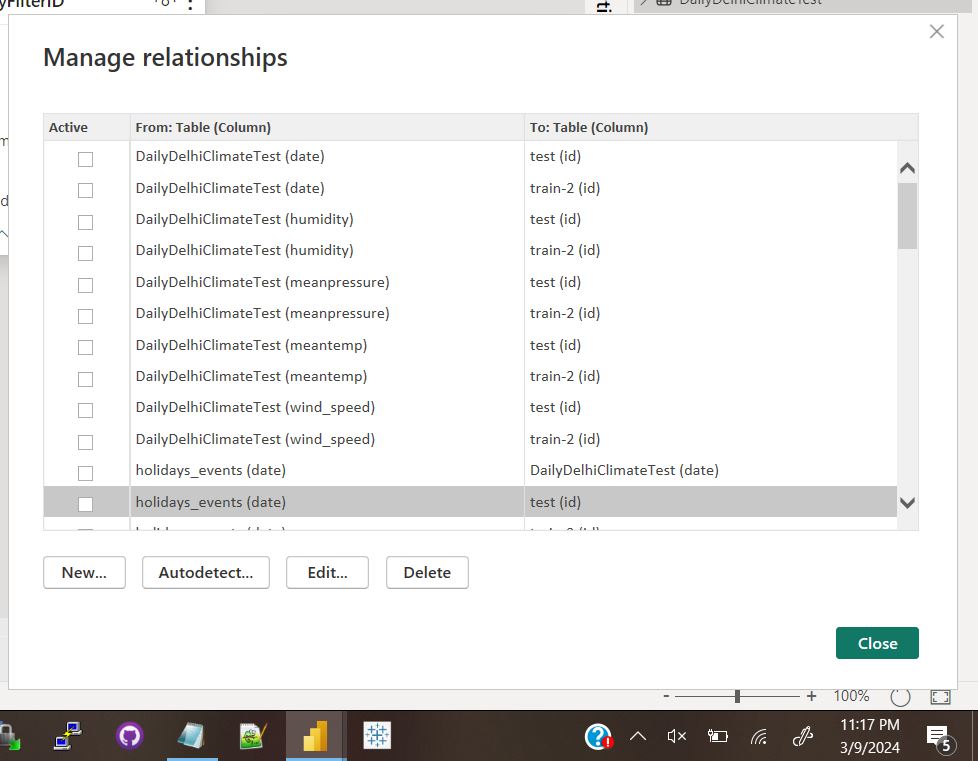
Then I clicked on the down arrow on the header of the date column.

I then selected the “Sort Descending” button.

# Question 2

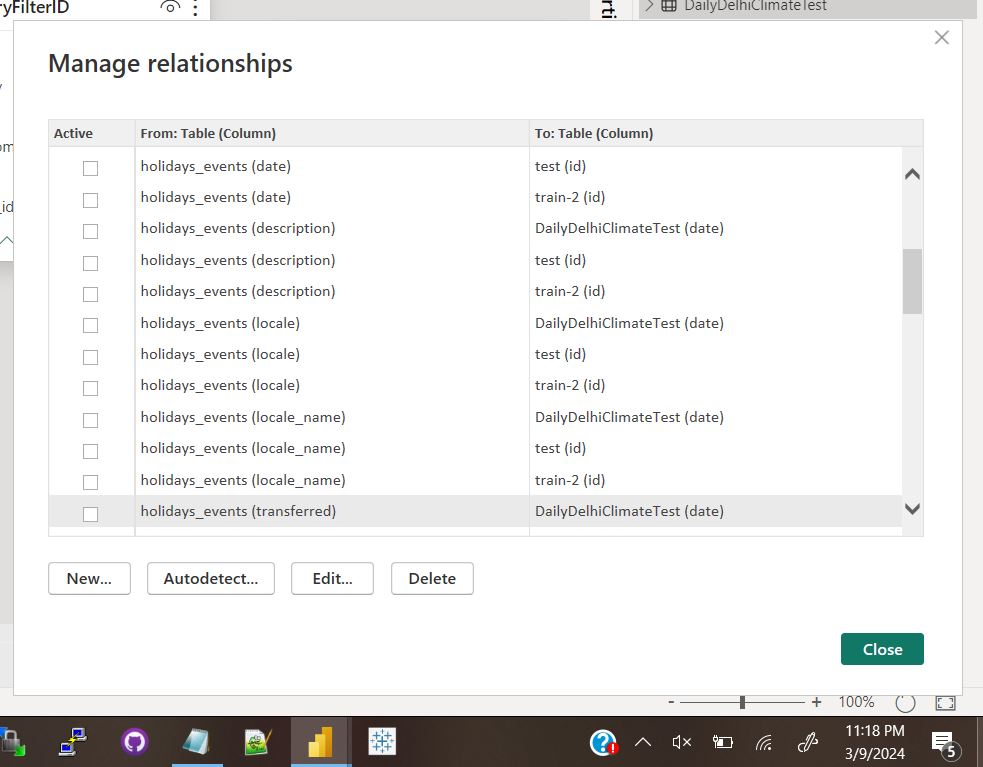
## Question 2.1

### Visualization 2.1.1



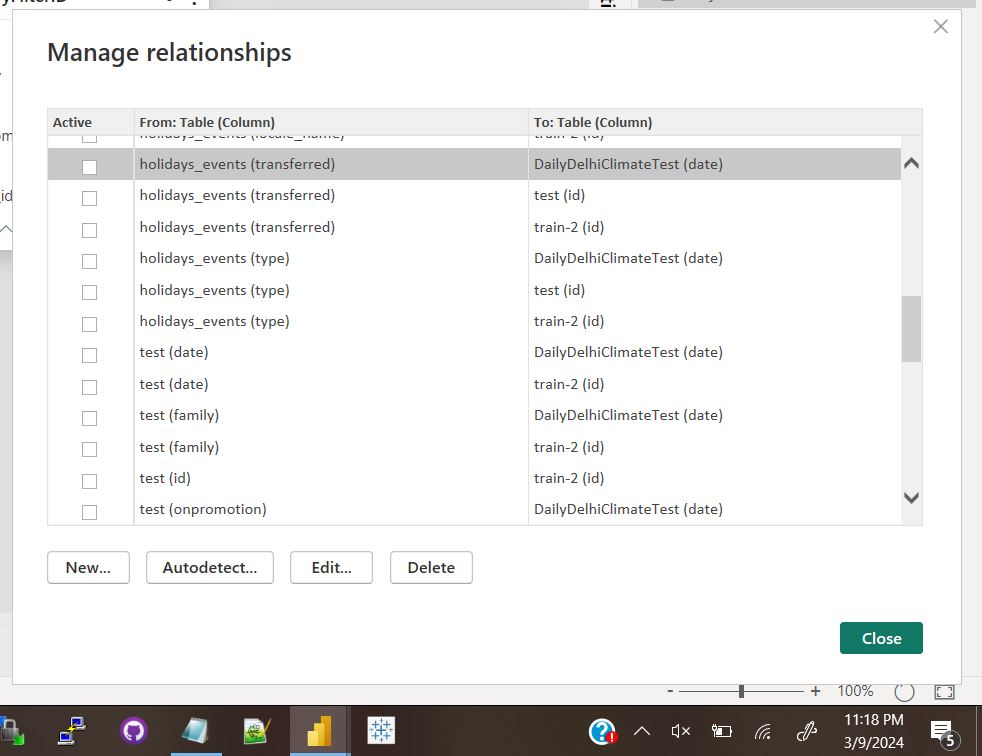
A screenshot showing the first 1/5th of the relationships that are one-to-one or many-to-one.

### Visualization 2.1.2



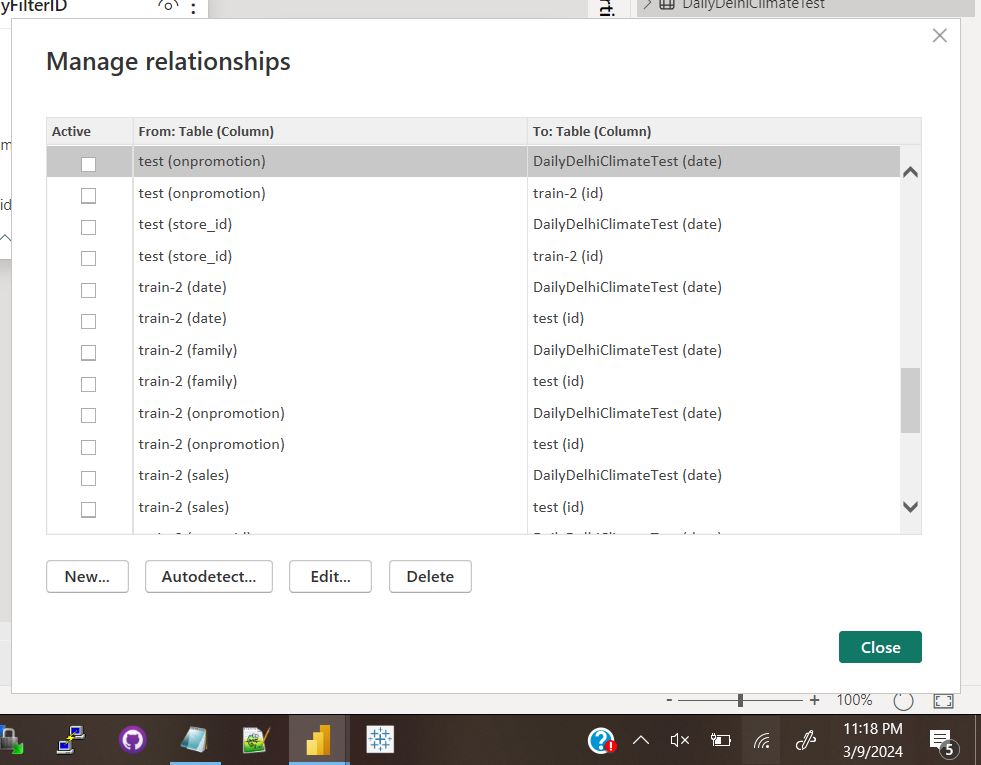
A screenshot showing the second 1/5th of the relationships that are one-to-one or many-to-one.

### Visualization 2.1.3



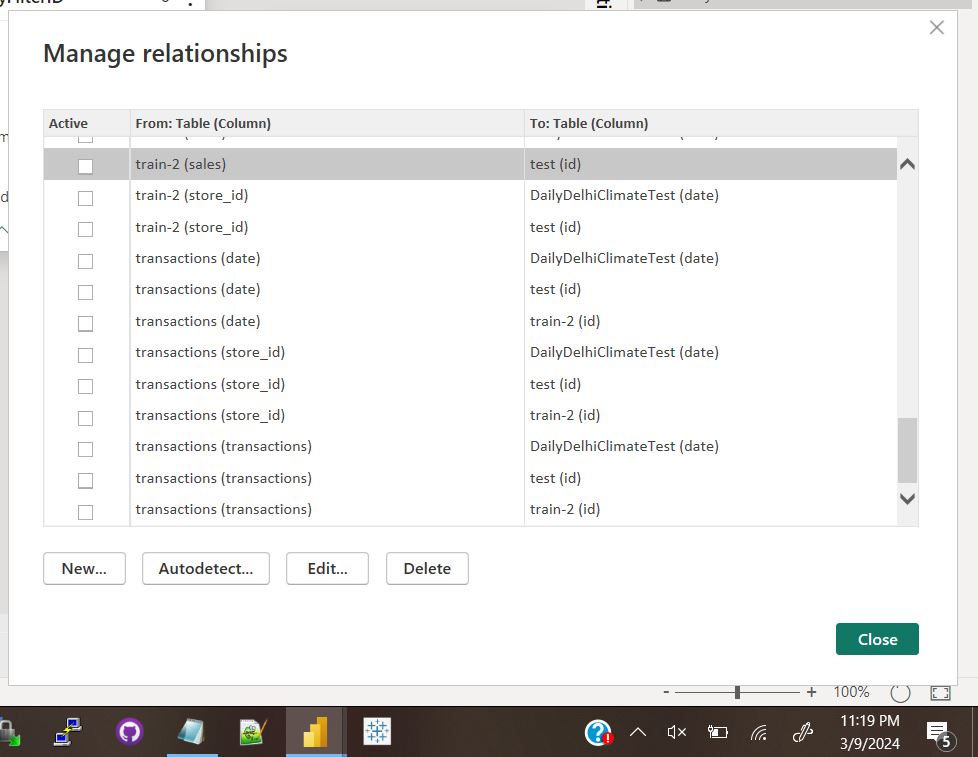
A screenshot showing the third 1/5th of the relationships that are one-to-one or many-to-one.

### Visualization 2.1.4



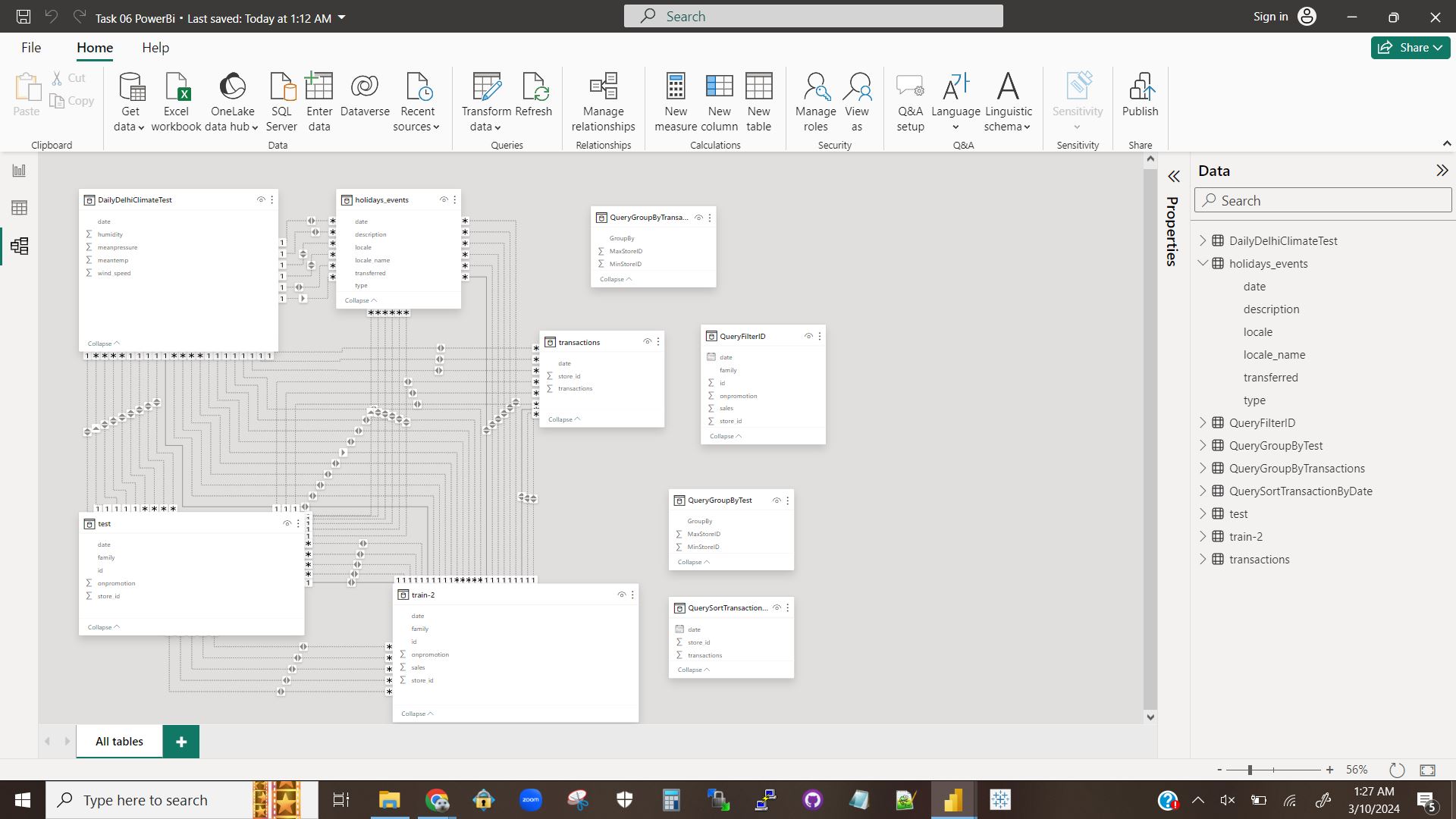
A screenshot showing the fourth 1/5th of the relationships that are one-to-one or many-to-one.

### Visualization 2.1.5



A screenshot showing the last 1/5th of the relationships that are one-to-one or many-to-one.

### Visualization 2.1.6



A screenshot showing the UML diagram of the relationships between the tables.

### Explanation 2.1.1

Yes, the cardinality will massively affect the relationship as it defines said relationship between tables, and it will massively affect how the tables interact because the different types of relationships behave very differently.

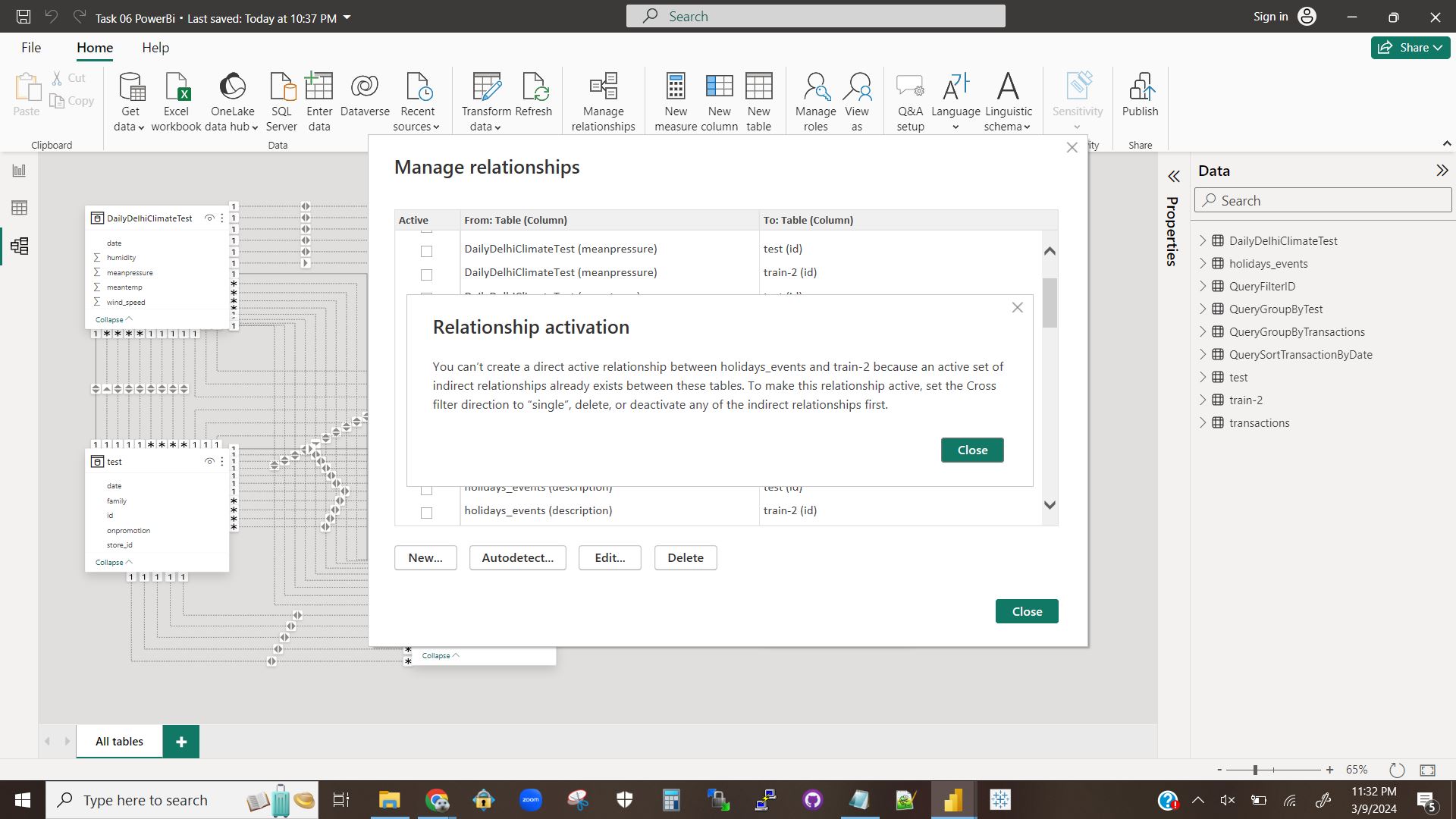
## Question 2.2

### Explanation 2.2.1

My experimentation showed it to have no effect upon the cardinality. This should be the case as cardinality already defines what direction the relationship points to, so it should be cardinality affecting the direction and not the other way around. What it does do is change which relationships can be set to active. If set to single, then you can/have to set the relationships each way to active in order to be able to go between the two. If only one direction is set, then the target table cannot link back to the parent table.

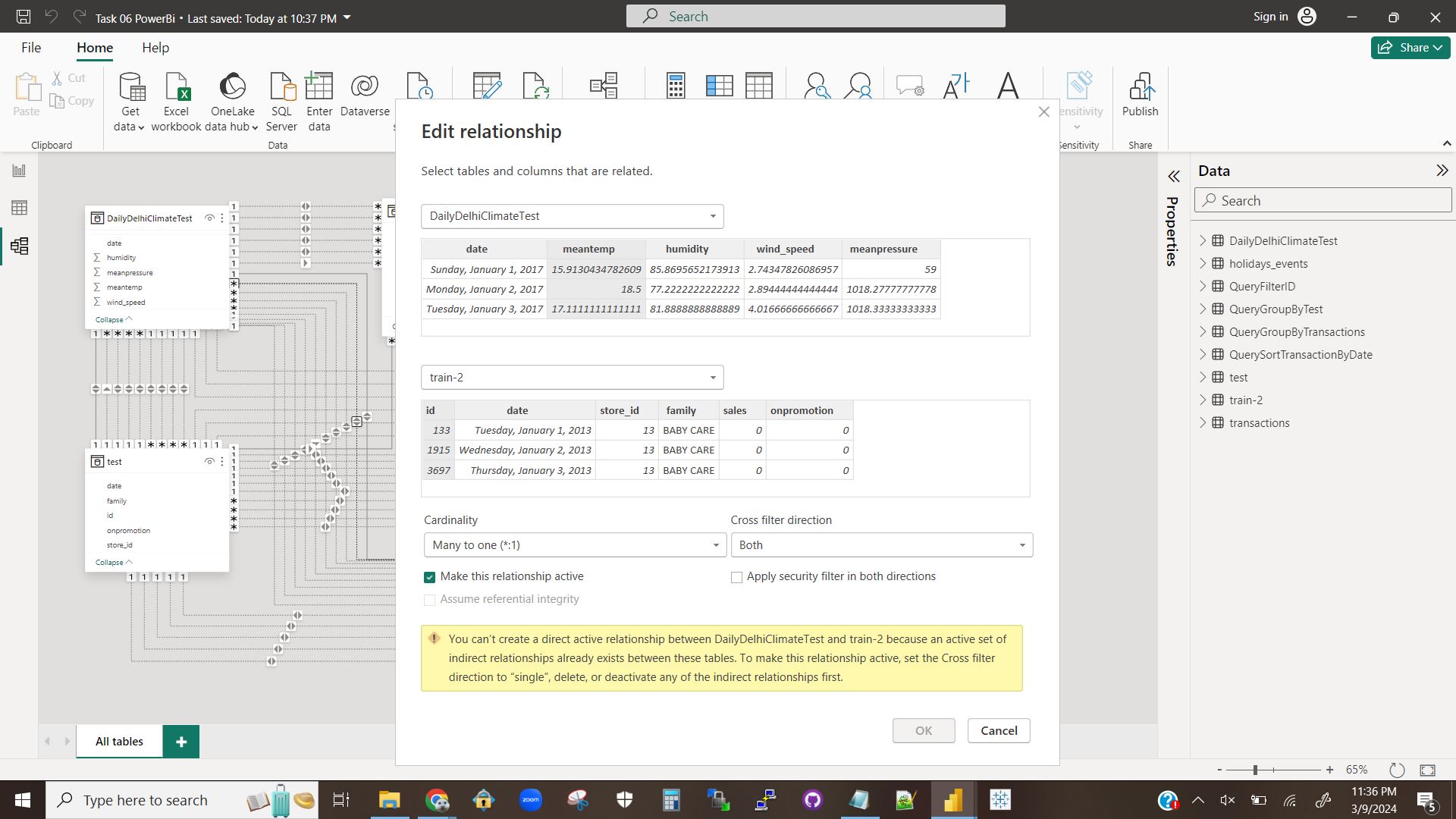
## Question 2.3

### Visualization 2.3.1



A screenshot showing that PowerBI will not allow a relationship to be set active for two tables where there is already an indirect relationship between them.

### Visualization 2.3.2



A screenshot showing that PowerBI will not allow multiple relationships between the same two tables to be active.

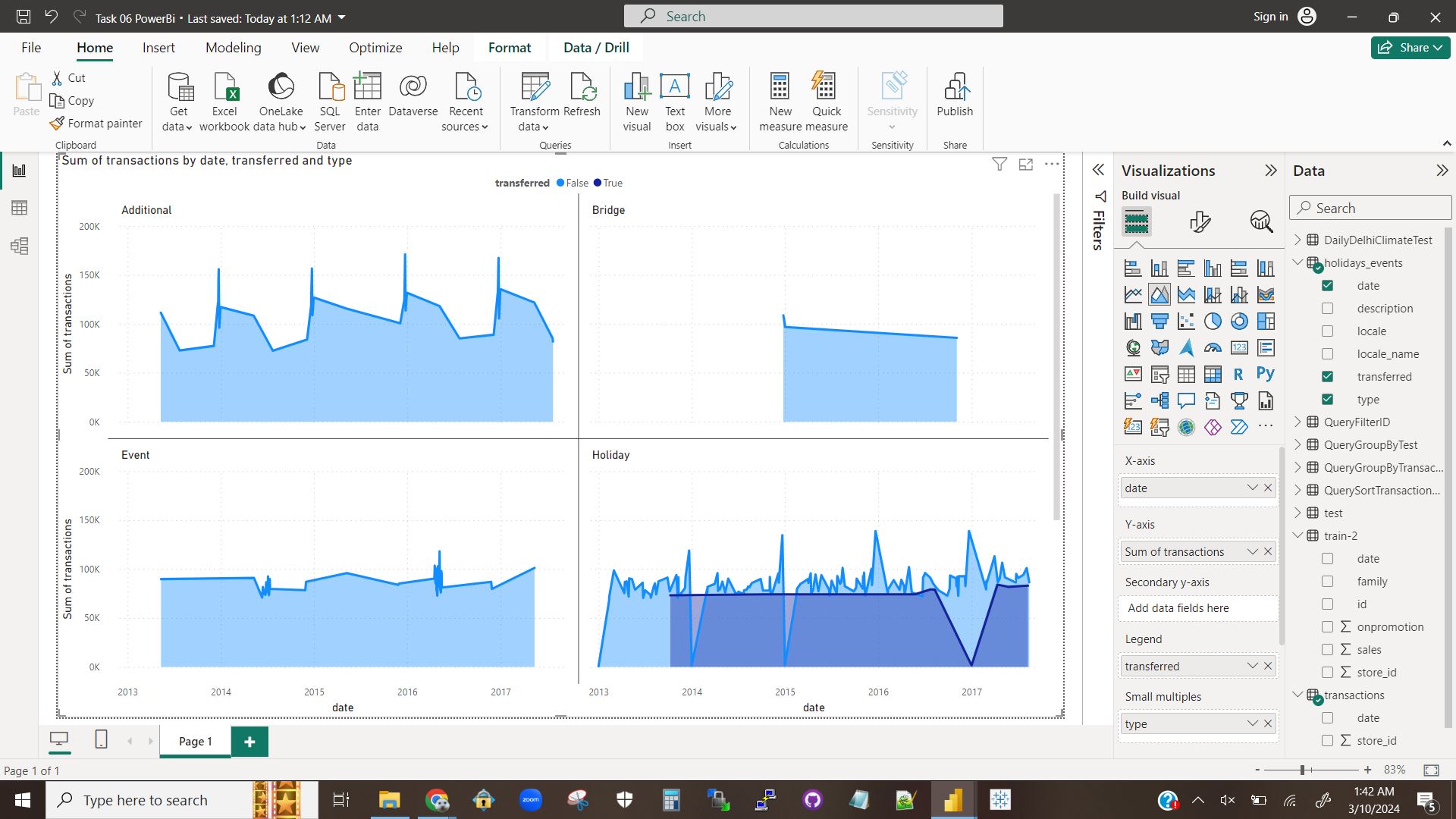
### Explanation 2.3.1

In PowerBI you can set multiple relationships in one table to be active as can be seen in Visualization 2.1.6. However, as shown in Visualization 2.3.2 you cannot set multiple relationships between the same two tables to be active, which makes sense. You only need one relationship to move between the two tables. If the filter is set to single, you should only be able to set one relationship in that direction active, but have more receiving relationships. Visualization 2.3.1 also shows that PowerBi will not allow a relationship to be set to active if the two tables in question already have an indirect relationship. This does not make as much sense. There are many possible scenarios where a table is indirectly linked many relationships deep, and queries trying to navigate deep relationship paths are long, complicated, and take much longer to run. A direct relationship occurring where an indirect relationship already exists might be a bit inefficient from a storage perspective, but in practice could result in a more efficient system in many cases.

# Question 3

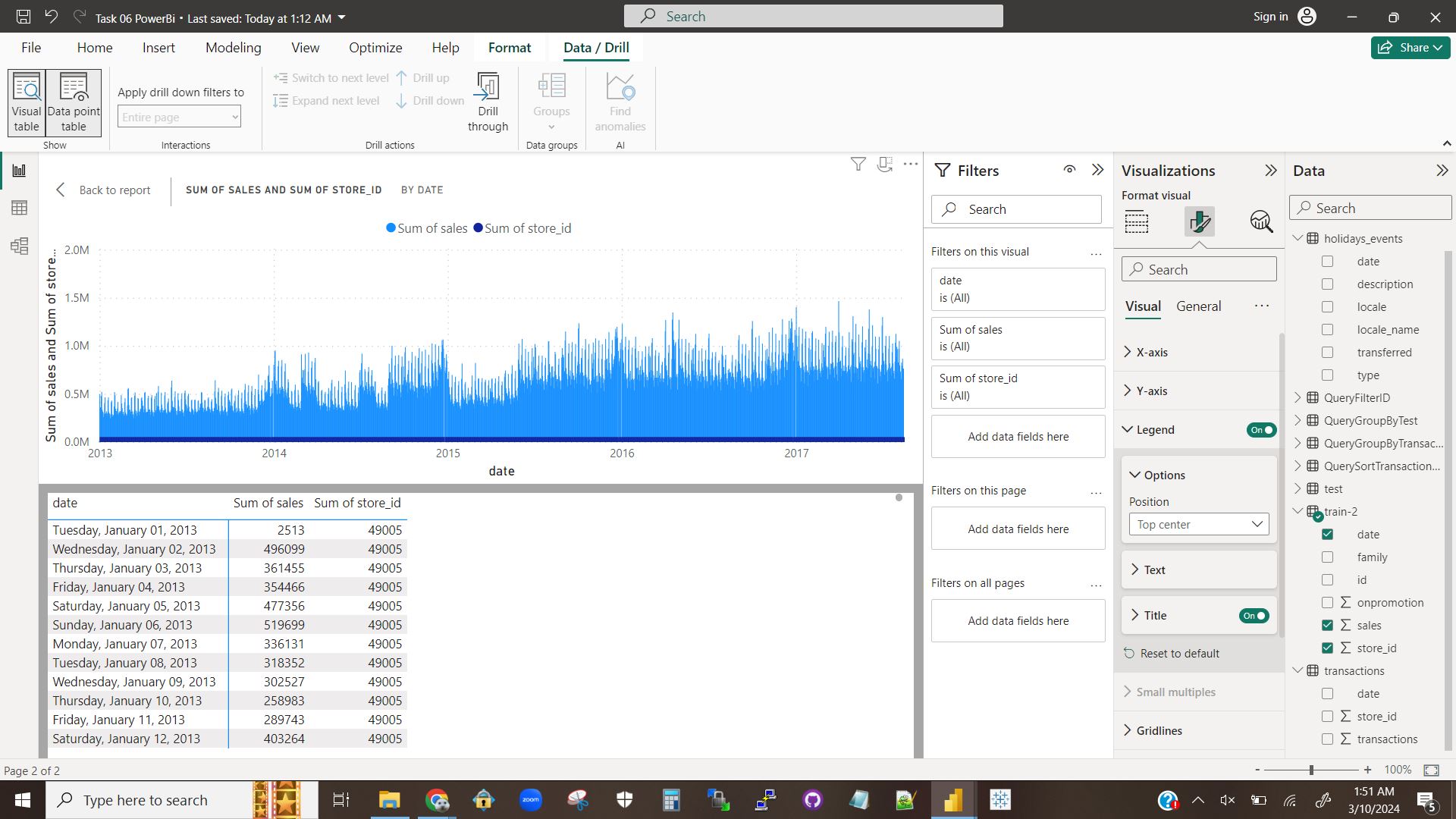
## Question 3.1

### Visualization 3.1.1



A screenshot showing the visualization of an area chart over date, transferred, and type vs. transactions.

### Visualization 3.1.2



A screenshot showing the two visualizations of a column chart sales and store id by date as well as the table of the raw data.

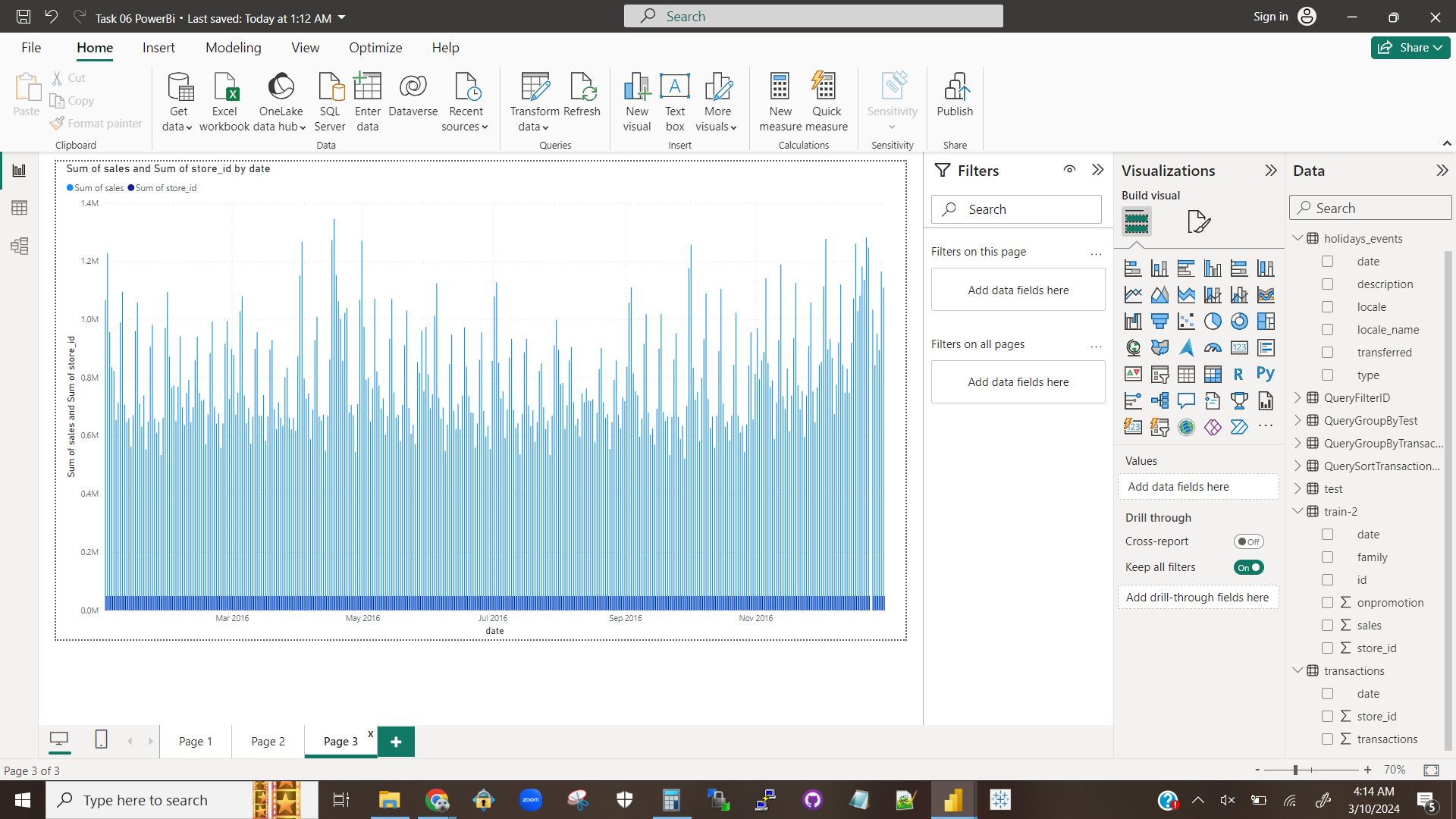
### Explanation 3.1.1

The first chart shows the sum of the transactions over time, broken down by date and type. The light blue lines represent the total transactions for non-transfers while the dark blue line is for the transactions of transfers. There are four sub-graphs for each type (Additional, Bridge, Holiday, Event). The graphs clearly show that only Holidays have transfers. Bridge and event transactions hold at a steady rate, though bridge transactions only happened for a short period of time. Additional transactions had a regular cyclical pattern, while holiday transactions were significantly more varied and irregular.

The second chart shows a comparison of transaction sales over time as well as a comparison of the number of stores over time. The table shows the numerical data for this chart. As can be seen by the dark blue columns the number of stores does not change over time. As can be seen by the light blue columns the sales have been steadily increasing over time.

## Question 3.2

### Visualization 3.2.1

A screenshot of a clustered column chart similar to Visualization 3.1.2, just filtered for only dates in 2016.

### Explanation 3.2.1

The highest sales in 2016 was $1,345,922 on April 18th and the second highest sales was $1,282,146 on December 23rd with a difference of $63,776.

## Question 3.3

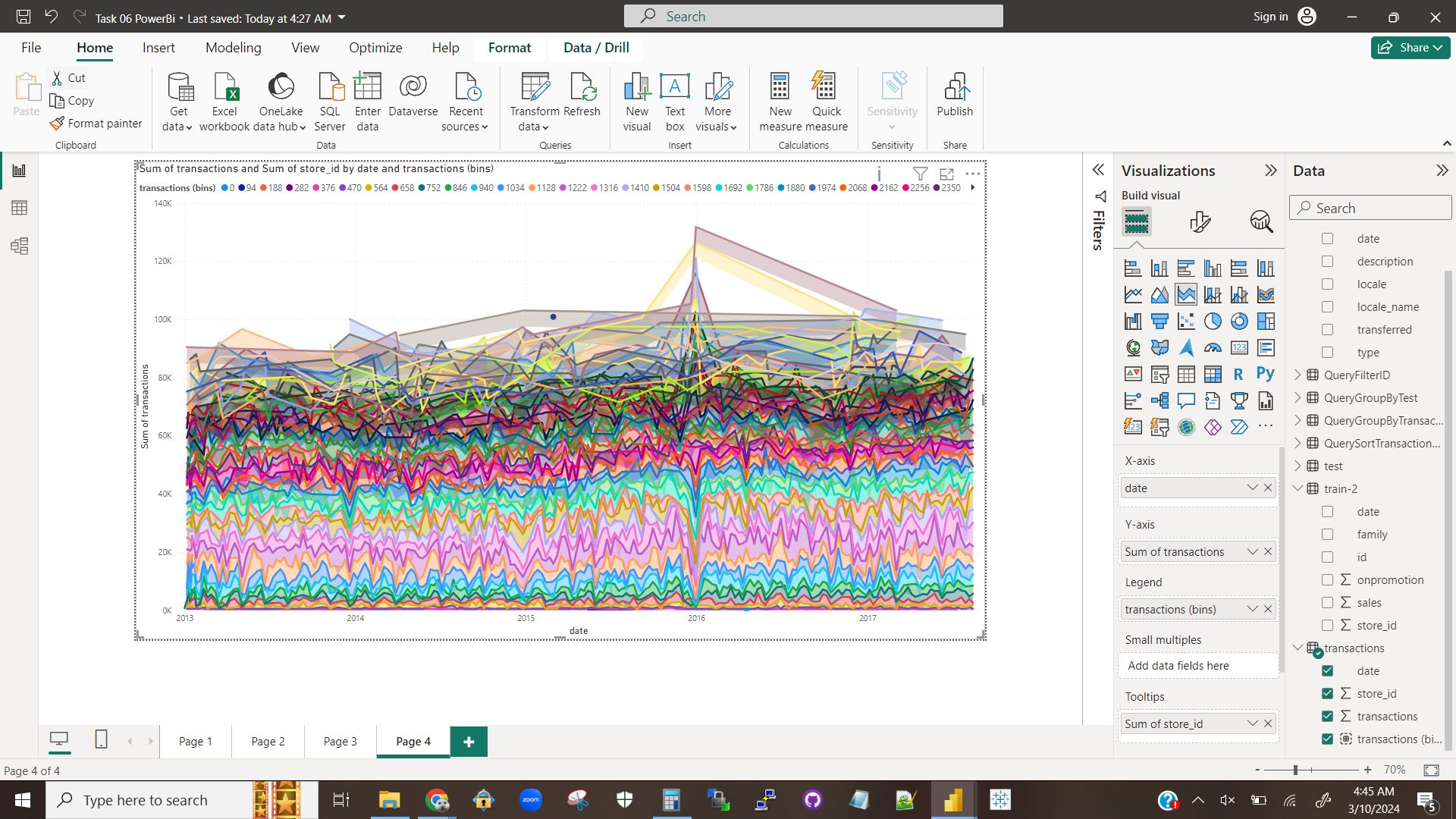
### Explanation 3.3.1

The different attributes work differently for different styles of visualization because each attribute represents something different, and different visualizations are best suited to representing different things. Attributes work best when paired with a visualization meant to represent the thing that the attribute also represents.

# Question 4

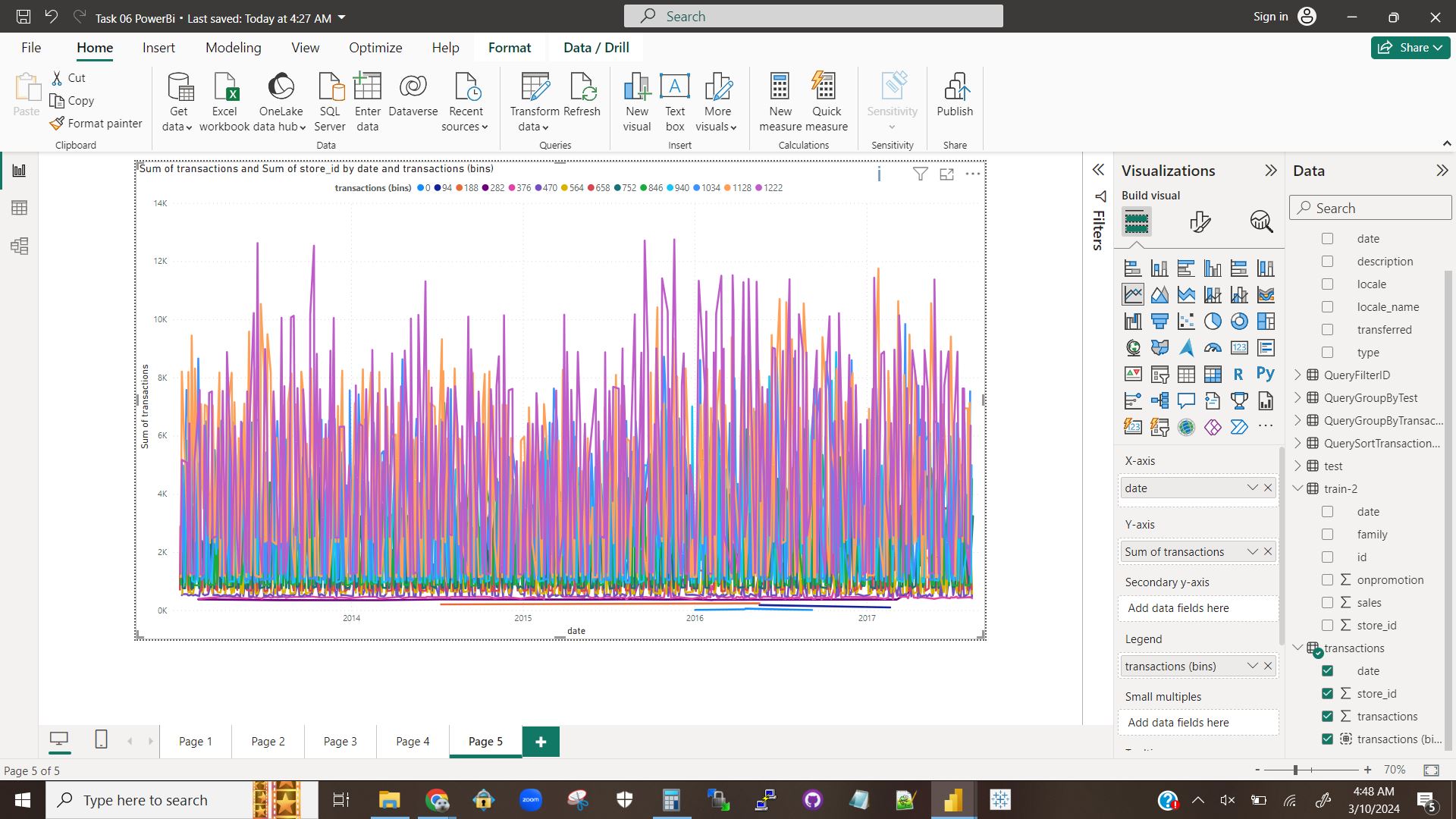
## Question 4.1

### Visualization 4.1.1



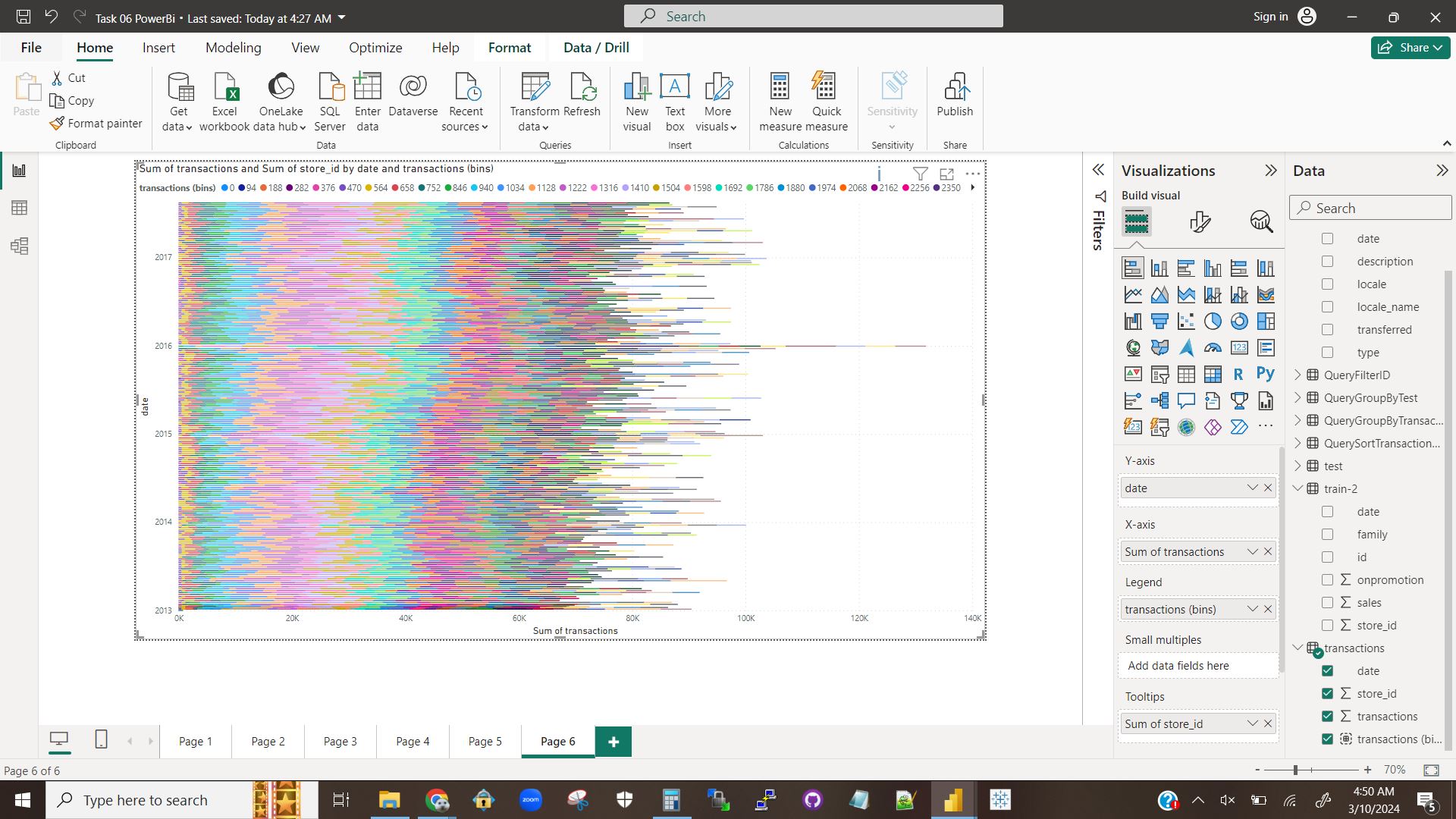
A stacked area chart representing the total transactions over time, color-coded by the bin the amount of transactions each entry falls into.

### Visualization 4.1.2



A line chart representing the same information as visualization 4.1.1.

### Visualization 4.1.3



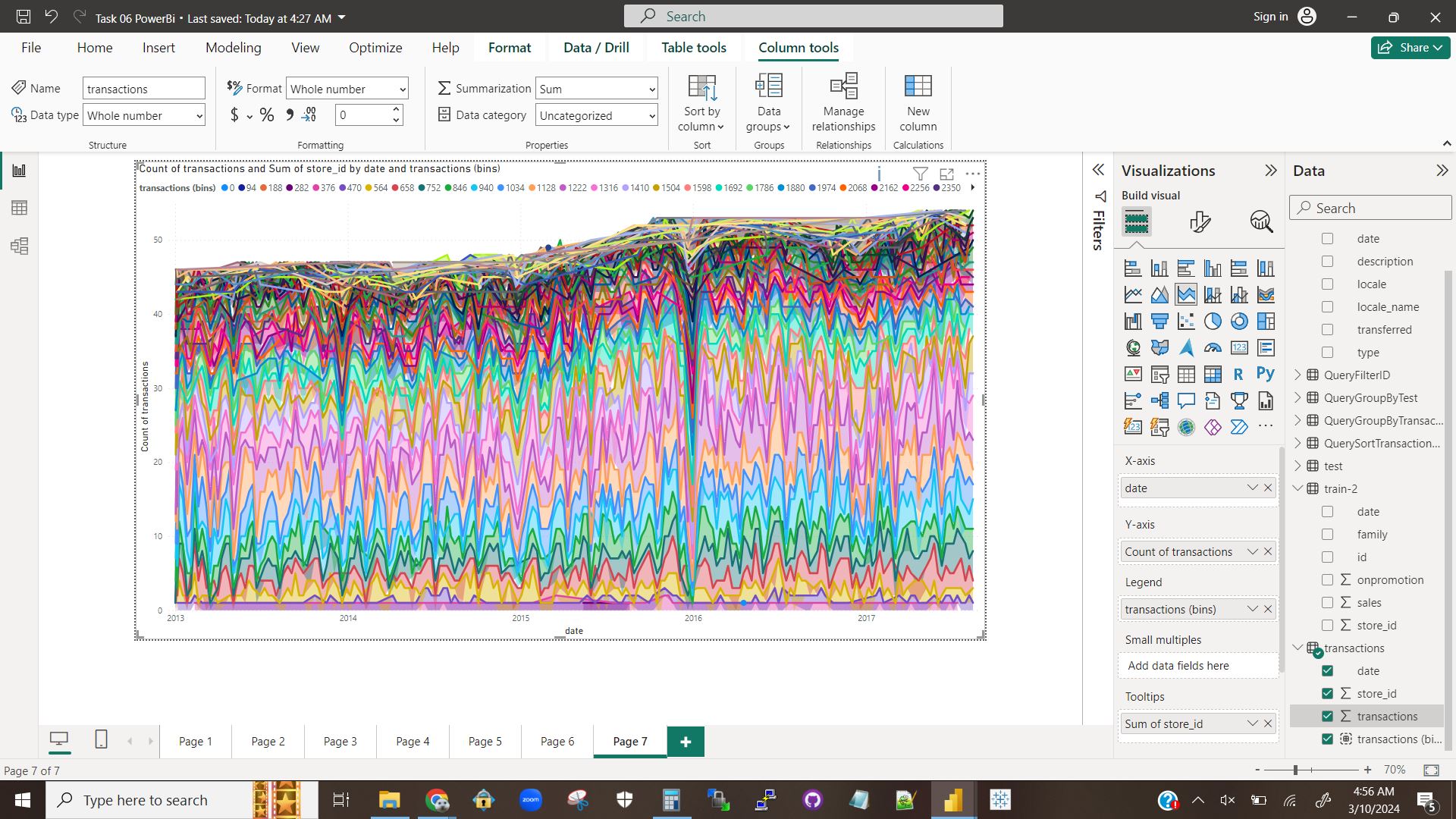
A stacked horizontal bar chart representing the same information as visualizations 4.1.1 and 4.1.2.

### Explanation 4.1.1

The stacked area chart of visualization 4.1.1 (and horizontal bar chart of visualization 4.3.1) show the total transactions for each date, with the percentage each bin takes up represented by how much of that bin’s color-coding occupies the assigned space. The line chart of visualization 4.1.2, on the other hand, does not stack everything so the performances of each bin can be directly compared within each date as well as across dates.

## Question 4.2

### Visualization 4.2.1



A stacked area chart showing the count, not the number, of transactions for each date.

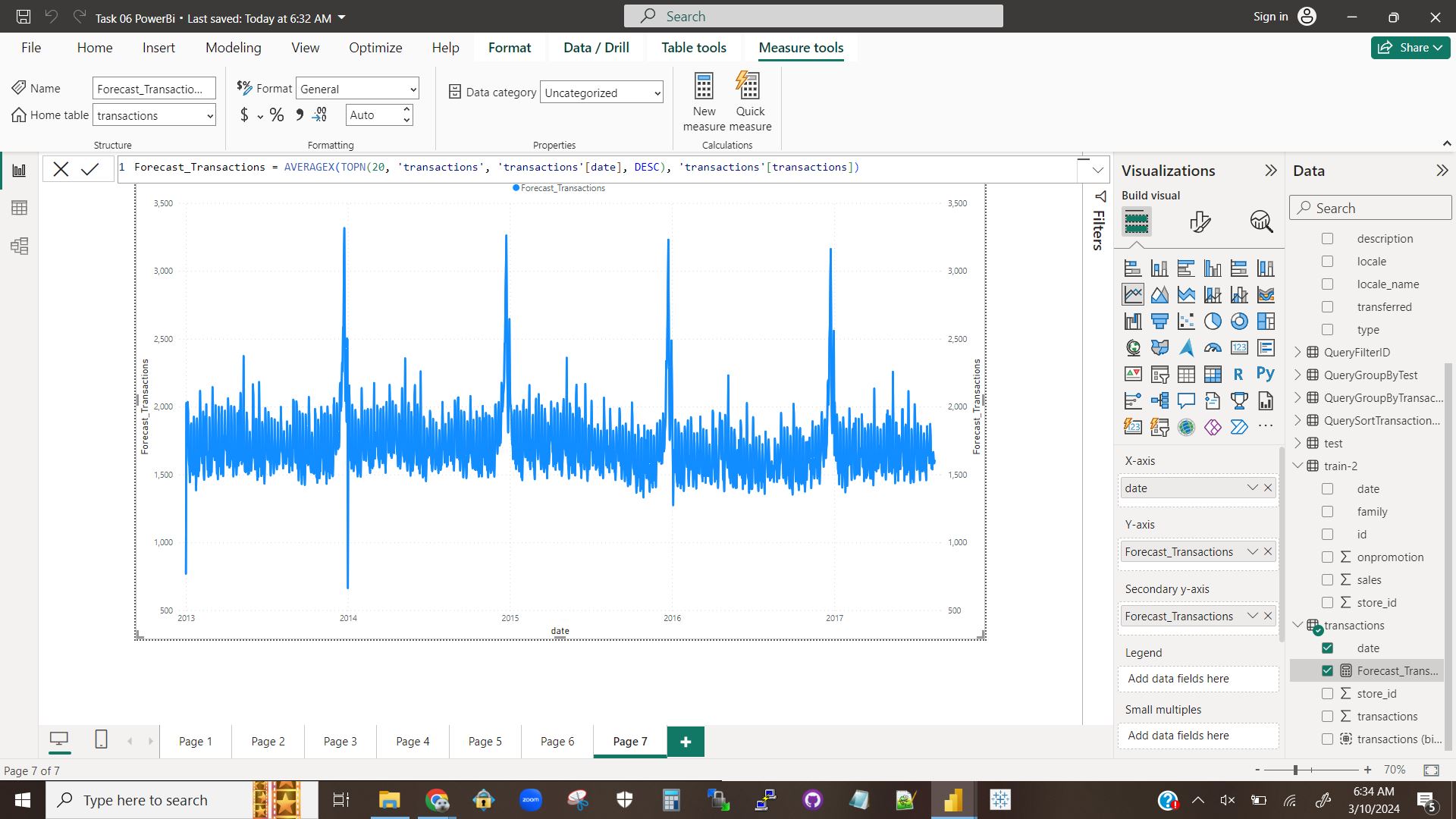
### Explanation 4.2.1

The values went down because the y-axis changed to measure how many transactions occurred, rather than how much money was involved in all transactions. Unless every transaction was for less than $1, there will be fewer transactions than there will be total transaction amount.

# Question 5

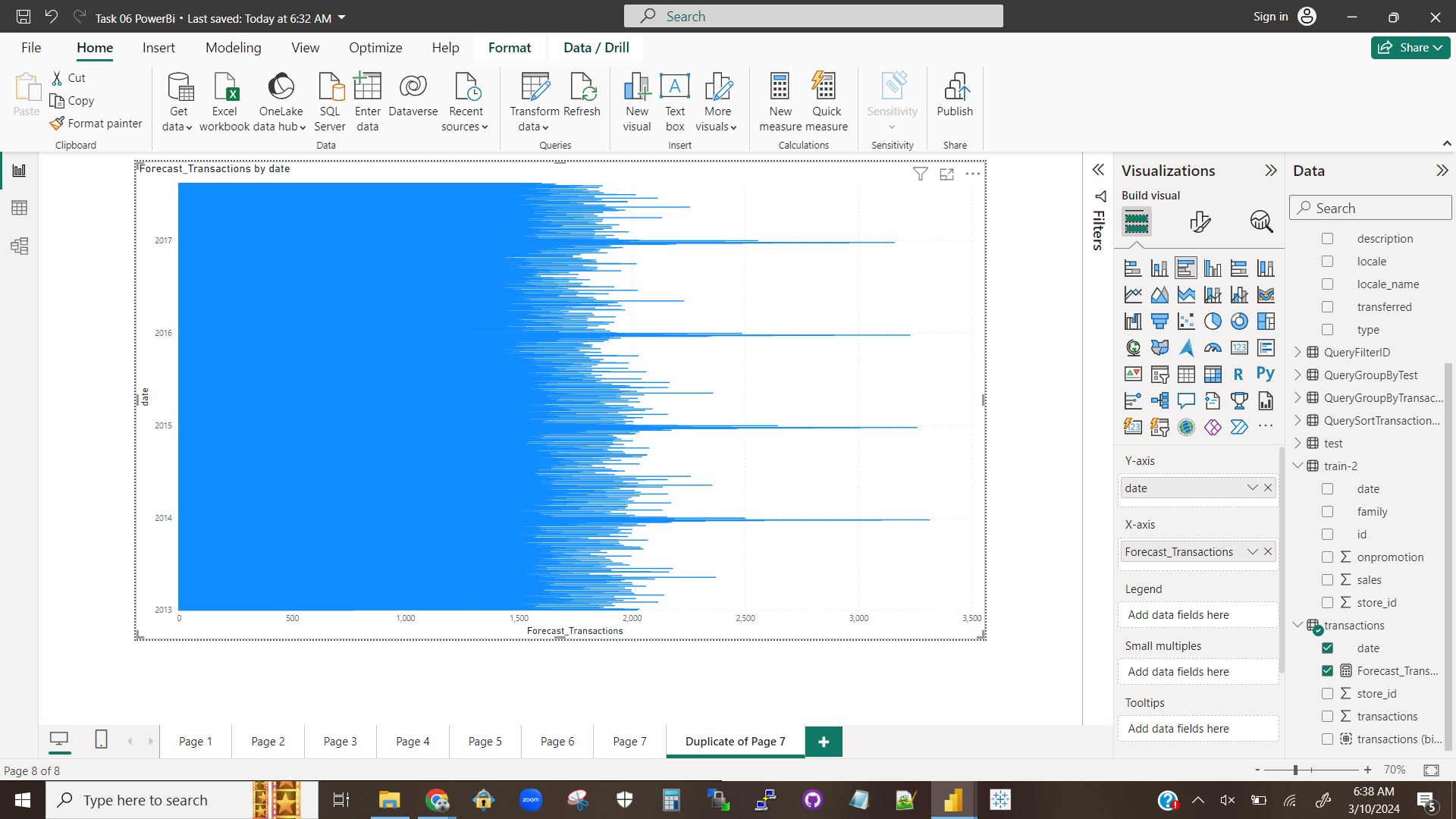
## Question 5.1

### Visualization 5.1.1



A screenshot showing a line chart forecast of the transactions for a date based on the date using 20 other data points to calculate an average.

### Visualization 5.1.2



A screenshot showing a horizontal bar chart forecast of the transactions for a date based on the date using 20 other data points to calculate an average.

### Explanation 5.1.1

The query was successfully created and two visualizations provided.

## Question 5.2

### Explanation 5.2.1

Time series forecasting is important because it allows people to use data to make predictions, and those predictions about what is likely to happen will greatly affect people’s decisions. For example, time series forecasting can be used to predict stock behavior, allowing investors to either make decisions that either make them more money or lose as little money as possible.

## Question 5.3

### Explanation 5.3.1

My understanding of this activity is that the tutorial showed us how to perform a basic time forecasting computation using PowerBi. It also showed us how to visualize the information so that we had a comprehensive report that makes the forecast very clear, as opposed to just tables of numbers that take manual comparison and thought to fully interpret.