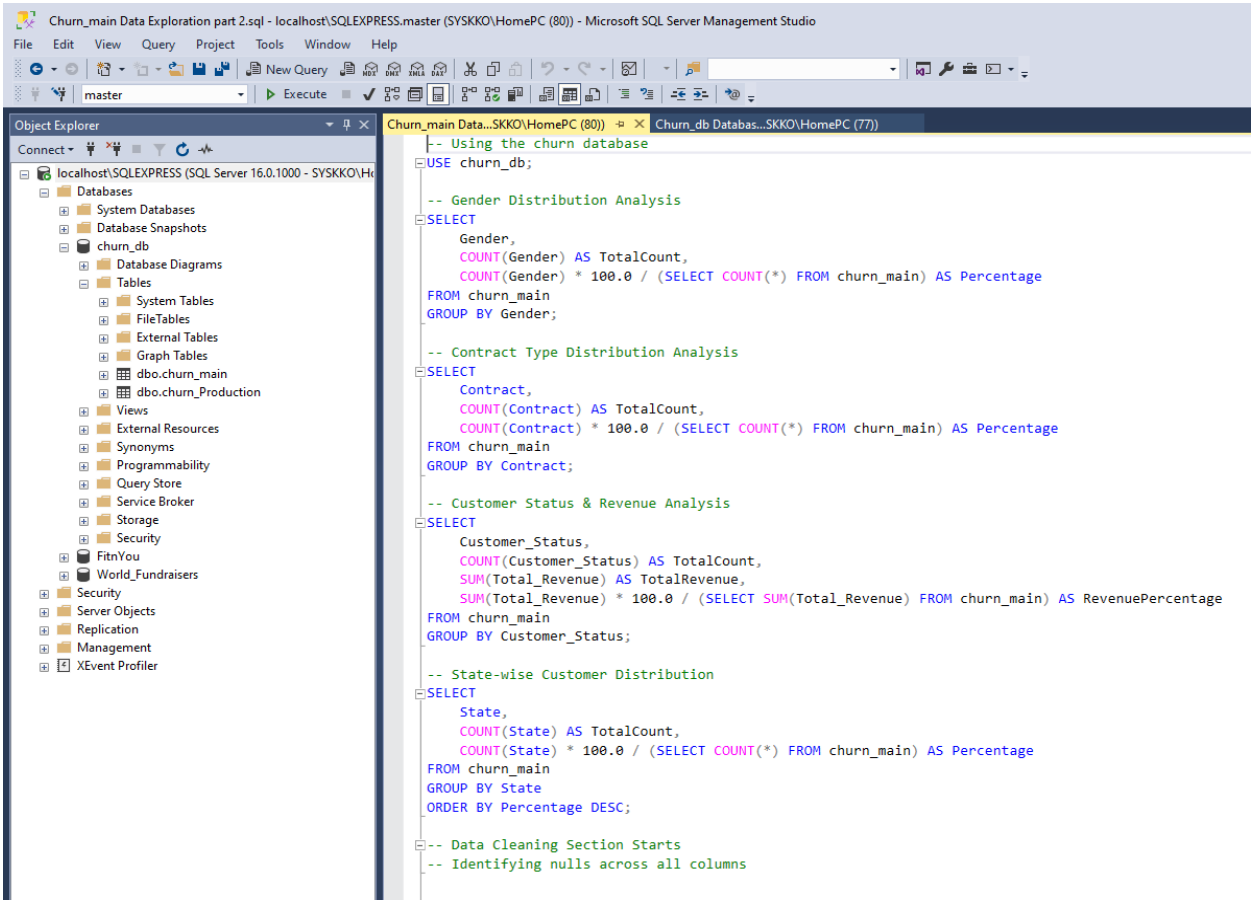
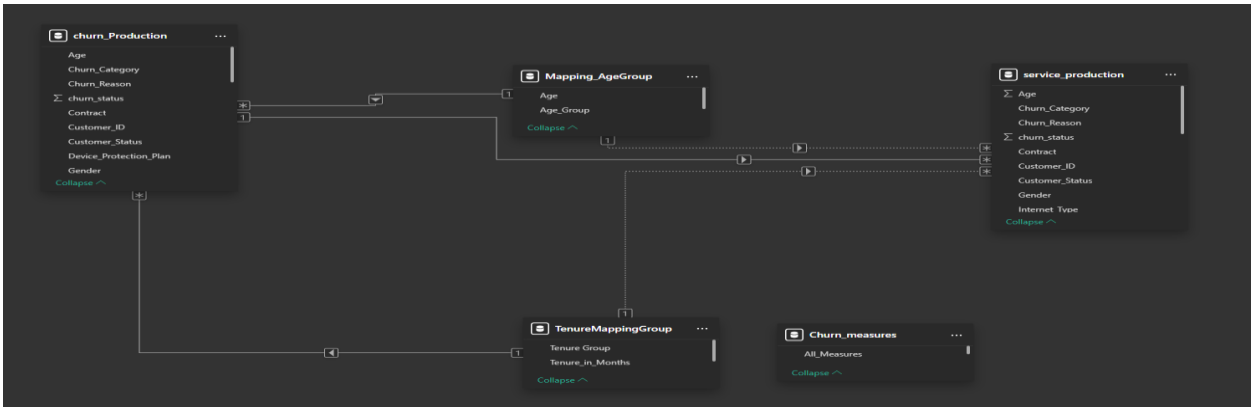
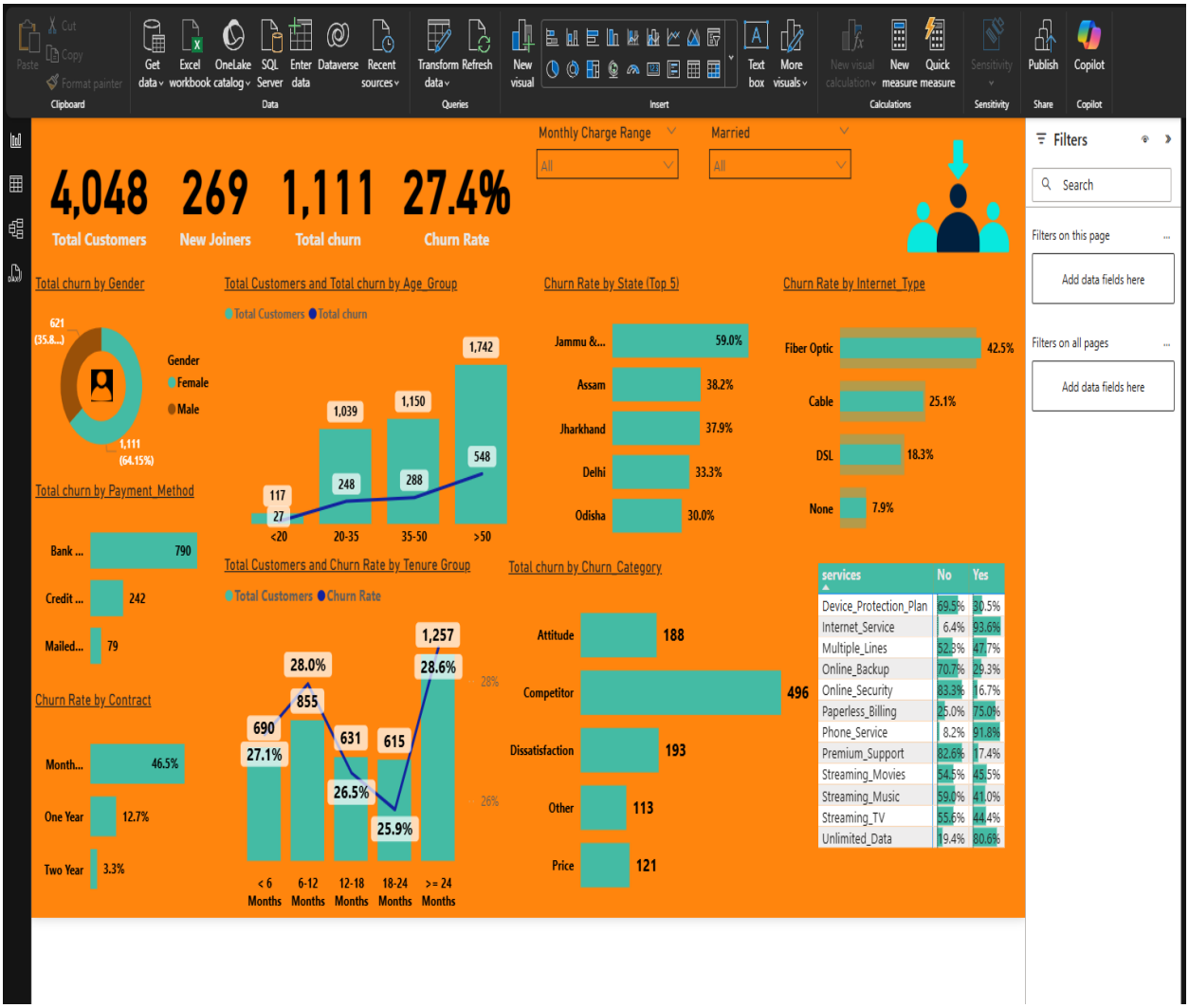
END TO END CHURN ANALYSIS PROJECT






Cut
Copy
Format painter
Paste
Clipboard
Get data
Excel workbook
OneLake catalog
SQL Server
Enter data
Dataverse
Recent sources
Data
Transform data
Refresh
Queries
New visual
Text box
More visuals
Insert
New visual calculation
New measure
Quick measure
Calculations
Sensitivity
Sensitivity
Publish
Share
Copilot
Copilot
Monthly Charge Range
All
Married
All
4,048
Total Customers
269
New Joiners
1,111
Total churn
27.4%
Churn Rate
Filters
Search
Filters on this page
Add data fields here
Filters on all pages
Add data fields here
Total churn by Gender
621
(35.8...)
Gender
Female
Male
1,111
(64.15%)
Total Customers and Total churn by Age_Group
Total Customers
Total churn
117
27
1,039
248
1,150
288
1,742
548
<20
20-35
35-50
>50
Churn Rate by State (Top 5)
Jammu &...
59.0%
Assam
38.2%
Jharkhand
37.9%
Delhi
33.3%
Odisha
30.0%
Churn Rate by Internet_Type
Fiber Optic
42.5%
Cable
25.1%
DSL
18.3%
None
7.9%
Total churn by Payment_Method
Bank ...
790
Credit ...
242
Mailed...
79
Total Customers and Churn Rate by Tenure Group
Total Customers
Churn Rate
690
27.1%
855
28.0%
631
26.5%
615
25.9%
1,257
28.6%
28%
26%
< 6 Months
6-12 Months
12-18 Months
18-24 Months
>= 24 Months
Total churn by Churn_Category
Attitude
188
Competitor
496
Dissatisfaction
193
Other
113
Price
121
Churn Rate by Contract
Month...
46.5%
One Year
12.7%
Two Year
3.3%
services
No
Yes
Device_Protection_Plan
69.5%
30.5%
Internet_Service
6.4%
93.6%
Multiple_Lines
52.3%
47.7%
Online_Backup
70.7%
29.3%
Online_Security
83.3%
16.7%
Paperless_Billing
25.0%
75.0%
Phone_Service
8.2%
91.8%
Premium_Support
82.6%
17.4%
Streaming_Movies
54.5%
45.5%
Streaming_Music
59.0%
41.0%
Streaming_TV
55.6%
44.4%
Unlimited_Data
19.4%
80.6%

1. Churn Analysis Introduction

In today's competitive corporate environment, customer retention is critical to long-term success. One important method for comprehending and lowering this client attrition is churn analysis. It entails looking at customer data to find trends and explanations for why customers leave. Businesses may identify which clients are most likely to leave and comprehend the variables influencing their choices by utilizing machine learning and advanced data analytics. With this information, businesses may take proactive measures to increase client loyalty and happiness.

1.1. Who is the Target Audience?

The methods and insights used in this research are applicable to a variety of businesses, even though it concentrates on churn analysis for a telecom company. This Churn research is useful for any company that values keeping customers, from retail and banking to healthcare and beyond. I will examine the strategies, resources, and best practices for enhancing customer loyalty and lowering attrition while turning data into useful insights for long-term success.

1.2. My Project Goal

To use the customer data and accomplish the following objectives, I will create a complete ETL process in a database and a Power BI dashboard:

- Analyze and visualize customer data at the following levels.
- Services for Demographic and Geographic
- Payment and Account Information

- Examine the Churner Profile and

- Determine Where Marketing Campaigns Should Be Implemented

- Find a Way to Forecast Upcoming Churners

1.3. Metrics Needed

- Total Clients
- Total Churn
- Churn Rate for New Hires

2. SQL Server ETL Process

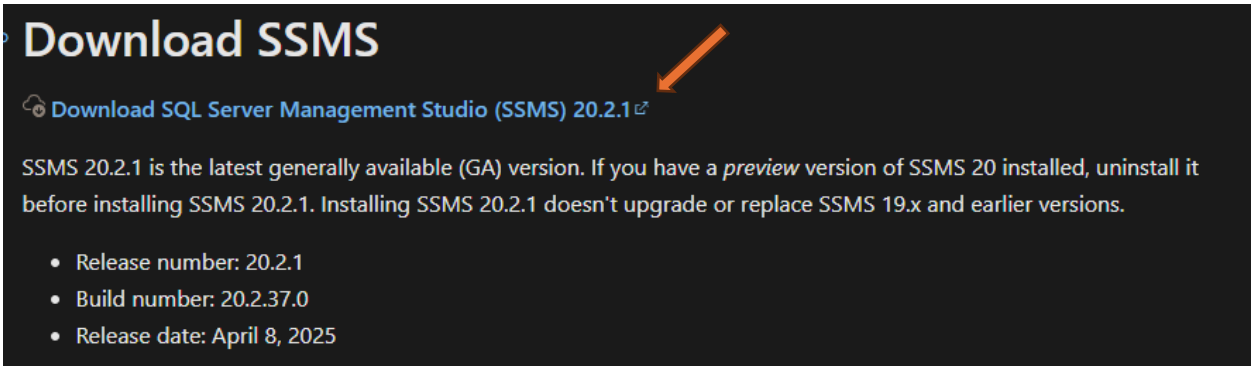
Step 1

The first step in analyzing churn involves importing the data from my source file. I will be utilizing Microsoft SQL Server for this, as it is a popular solution in the industry and offers superior capability in managing recurring data loads and ensuring data integrity compared to an Excel file.

2.1 Download SSMS

To execute our SQL queries, Microsoft offers a graphical user interface called SQL Server Management Studio. You can obtain the latest version from the link provided below.

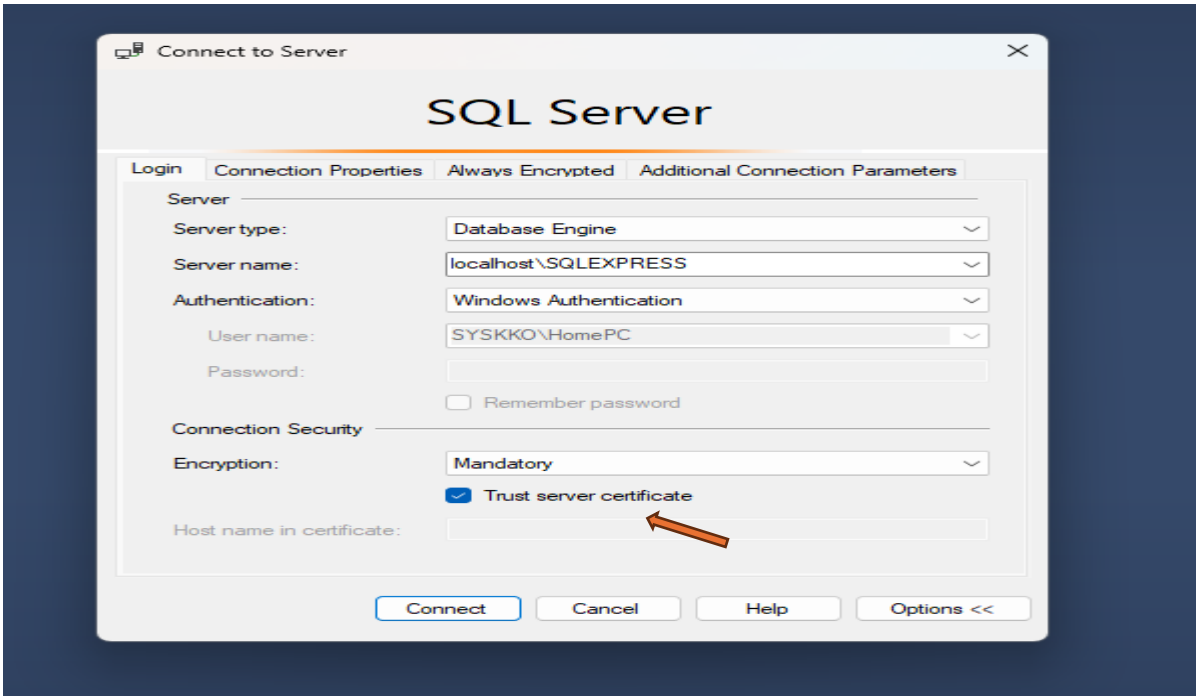
https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16



2.2 Creating Database

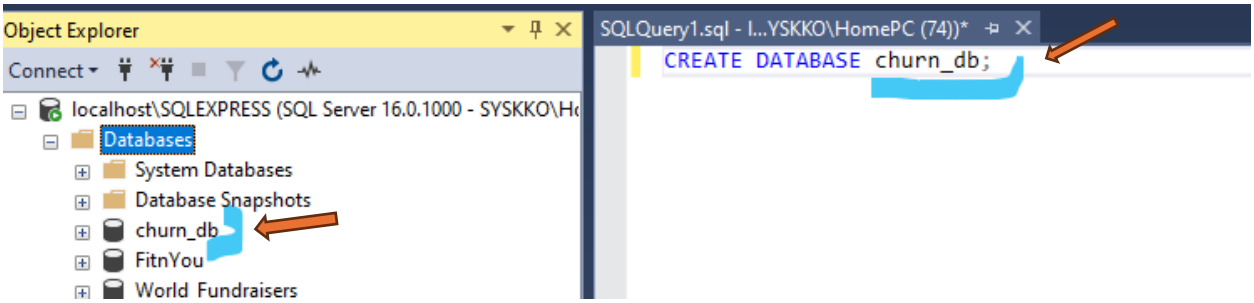
After completing the installation, I will be directed to the following screen. I made sure to copy and save the server's name somewhere, as it will be needed later. Additionally, I checked the box that says, "Trust Server Certificate," and then I clicked on Connect.

Once I was connected, I clicked on the NEW QUERY button located in the top ribbon and then entered the following query. This will create a new database named Churn_db



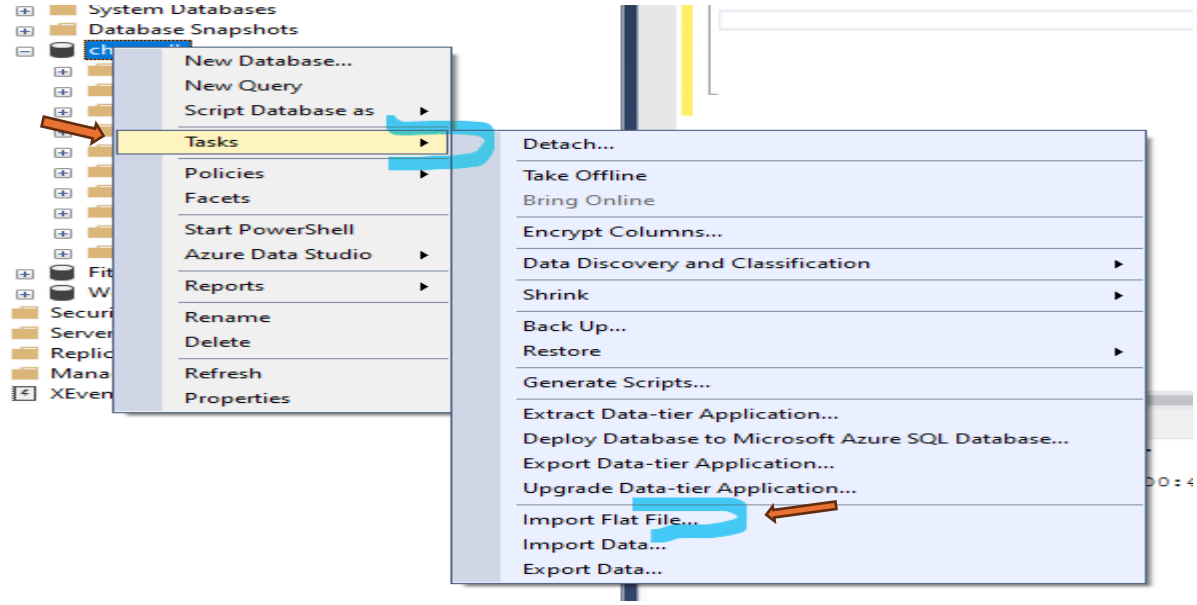
The steps to do from here are to:

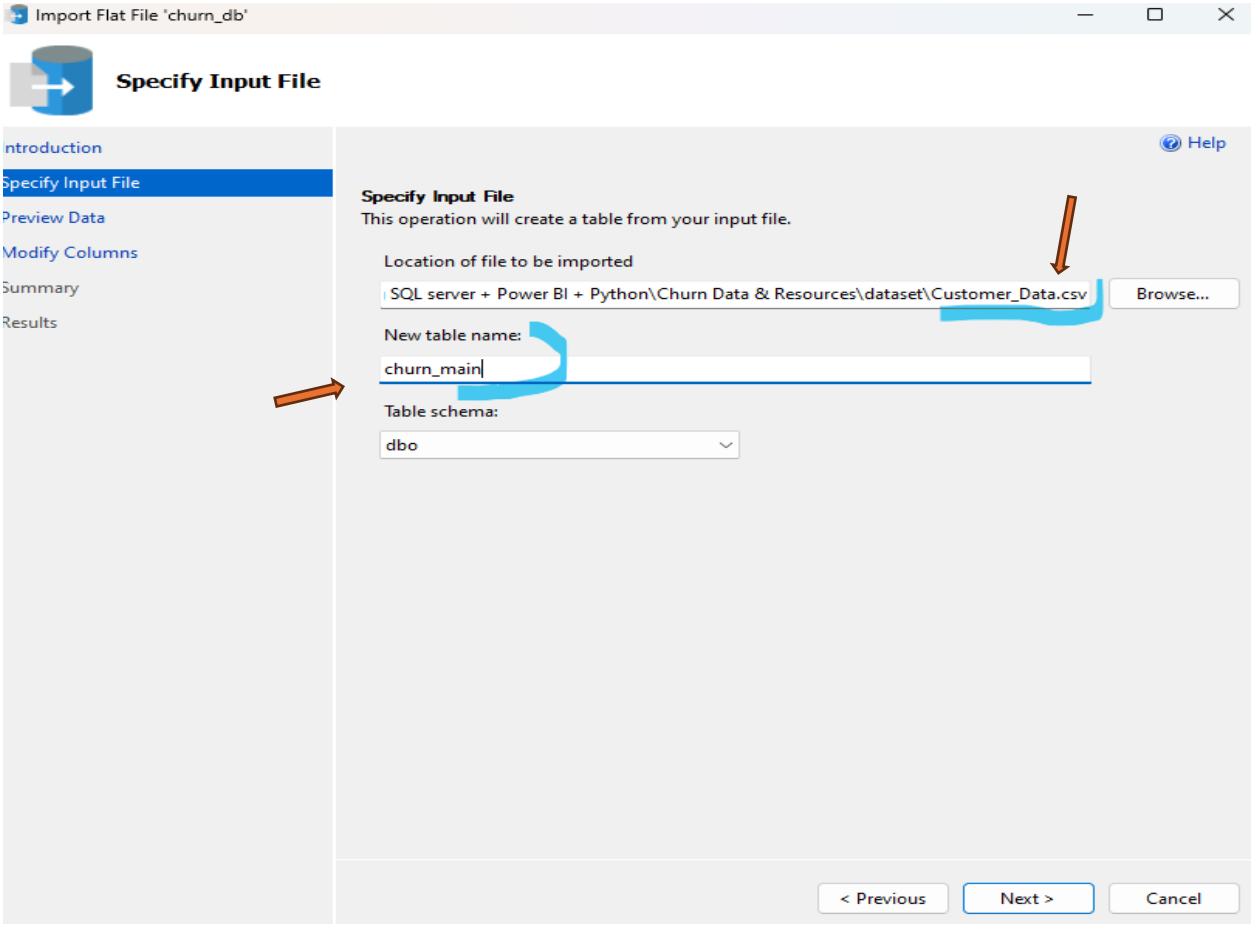
- Creating a database
- Creating a Table
- Load The Data into the Table



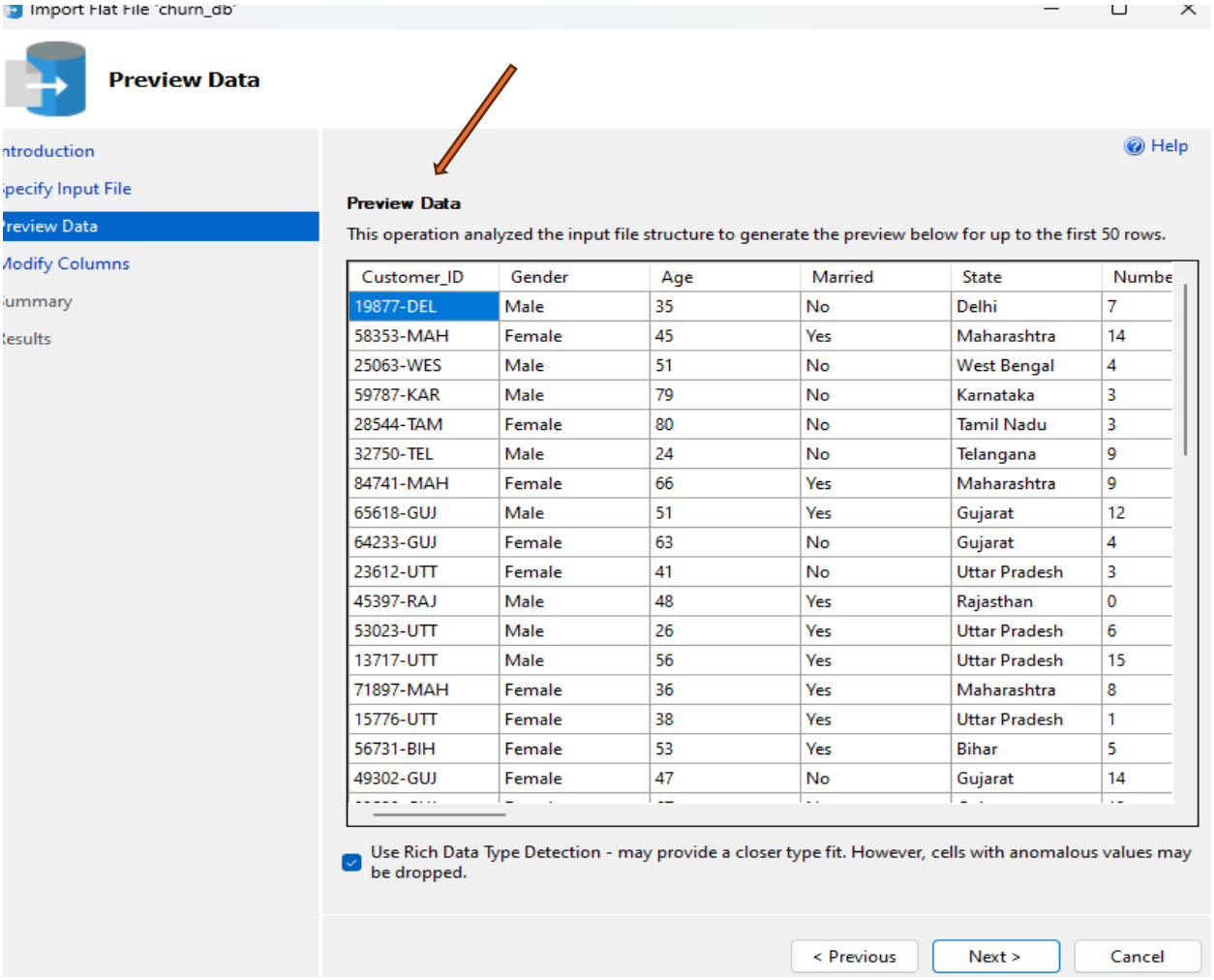
Now the database is shown on the left explorer panel.

To import a CSV file into a SQL Server staging table using the Import Wizard, I began by right-clicking on the newly created churn_db database in the explorer panel, then navigated to Task >> Import >> Flat file >> Browse for the CSV file called Customer_Data. It's important that I set customerId as the primary key and permit null values for all other columns to prevent any data load errors. Additionally, ensure that I modify the data type from Bit to Varchar(50). This change is necessary because I encountered issues with the BIT data type while using the import wizard, whereas Varchar (50) worked without problems.

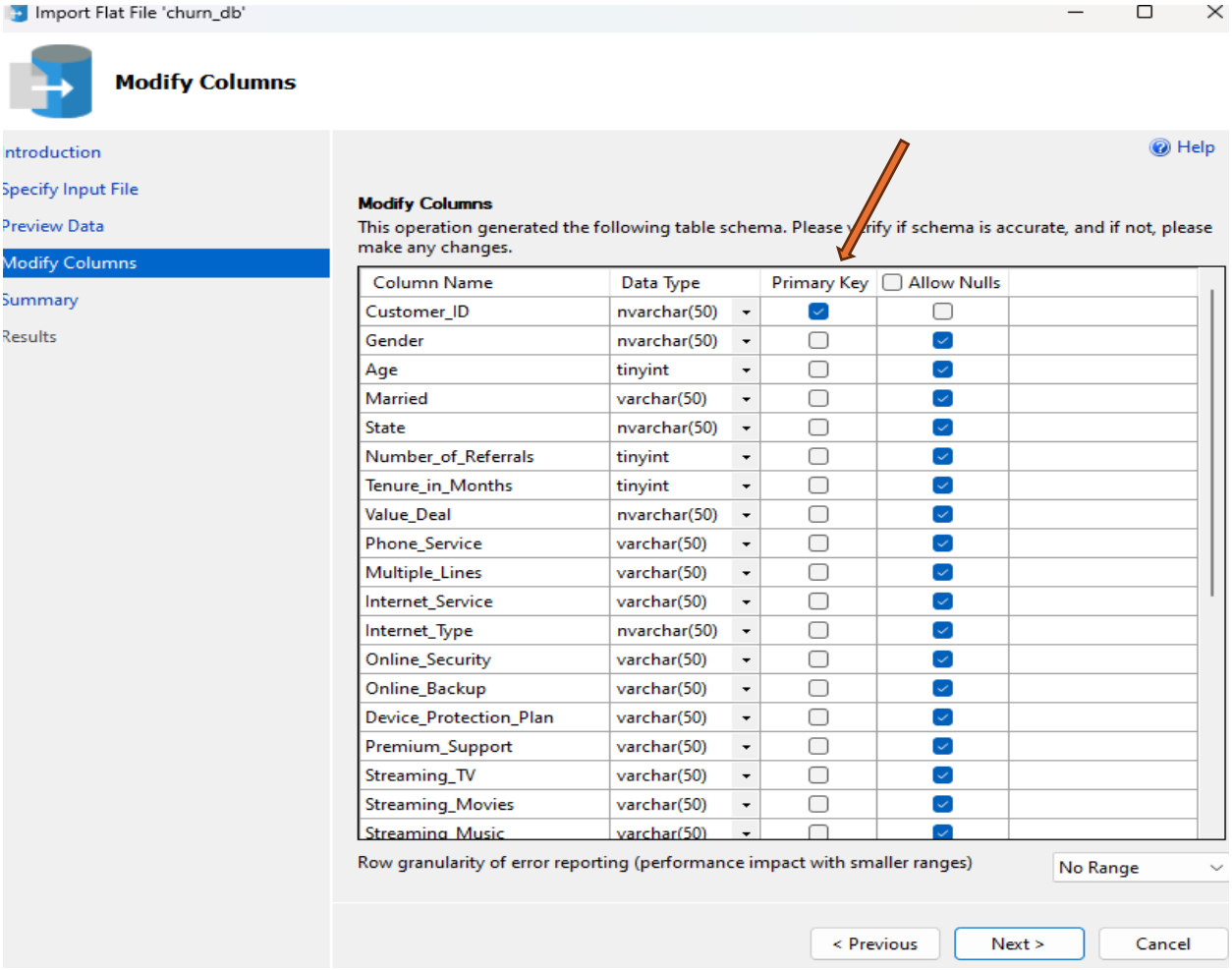

System Databases
Database Snapshots
New Database...
New Query
Script Database as
Tasks
Policies
Facets
Start PowerShell
Azure Data Studio
Reports
Rename
Delete
Refresh
Properties
Detach...
Take Offline
Bring Online
Encrypt Columns...
Data Discovery and Classification
Shrink
Back Up...
Restore
Generate Scripts...
Extract Data-tier Application...
Deploy Database to Microsoft Azure SQL Database...
Export Data-tier Application...
Upgrade Data-tier Application...
Import Flat File...
Import Data...
Export Data...


Import Flat File 'churn_db'
Specify Input File
ntroduction
Specify Input File
Preview Data
Modify Columns
Summary
Results
Help
Specify Input File
This operation will create a table from your input file.
Location of file to be imported
SQL server + Power BI + Python\Churn Data & Resources\dataset\Customer_Data.csv
Browse...
New table name:
churn_main
Table schema:
dbo
< Previous
Next >
Cancel

The dbo means database owner which is the default database/schema of the SQL SERVER.

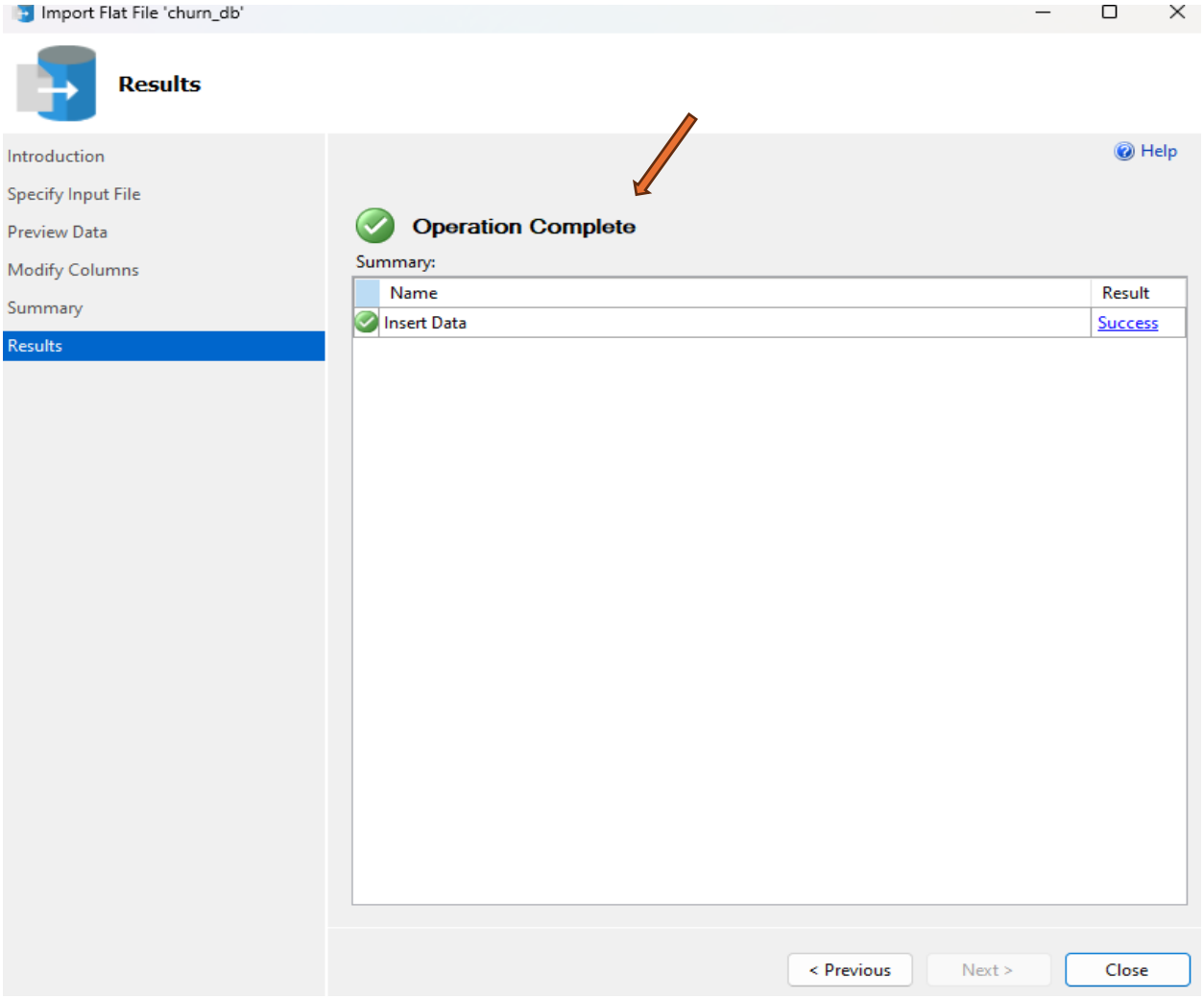


Now this is the preview of the data being loaded.

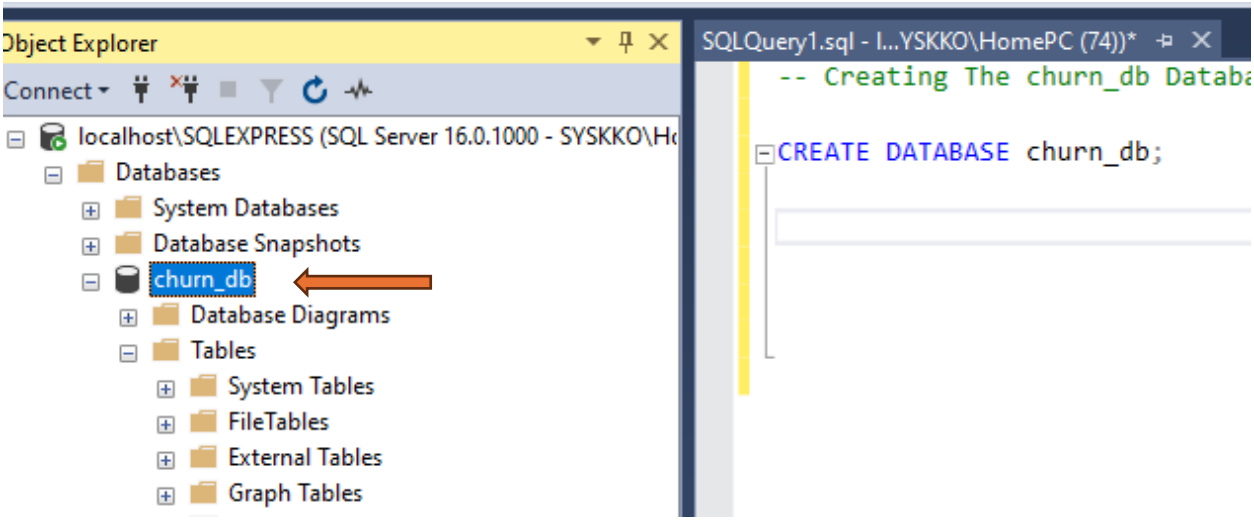


Now this Modify column is where I modified my Datatype, Primary key, Null values, is important because with the wrong placement of datatypes, the data will not load, and it will show an error.

So, I changed all datatypes showing bits to varchar(50) and I ticked the box for Customer_ID to be the primary key and I unchecked Allow Null for Customer_ID because originally all unique identifiers shouldn't have Null values.



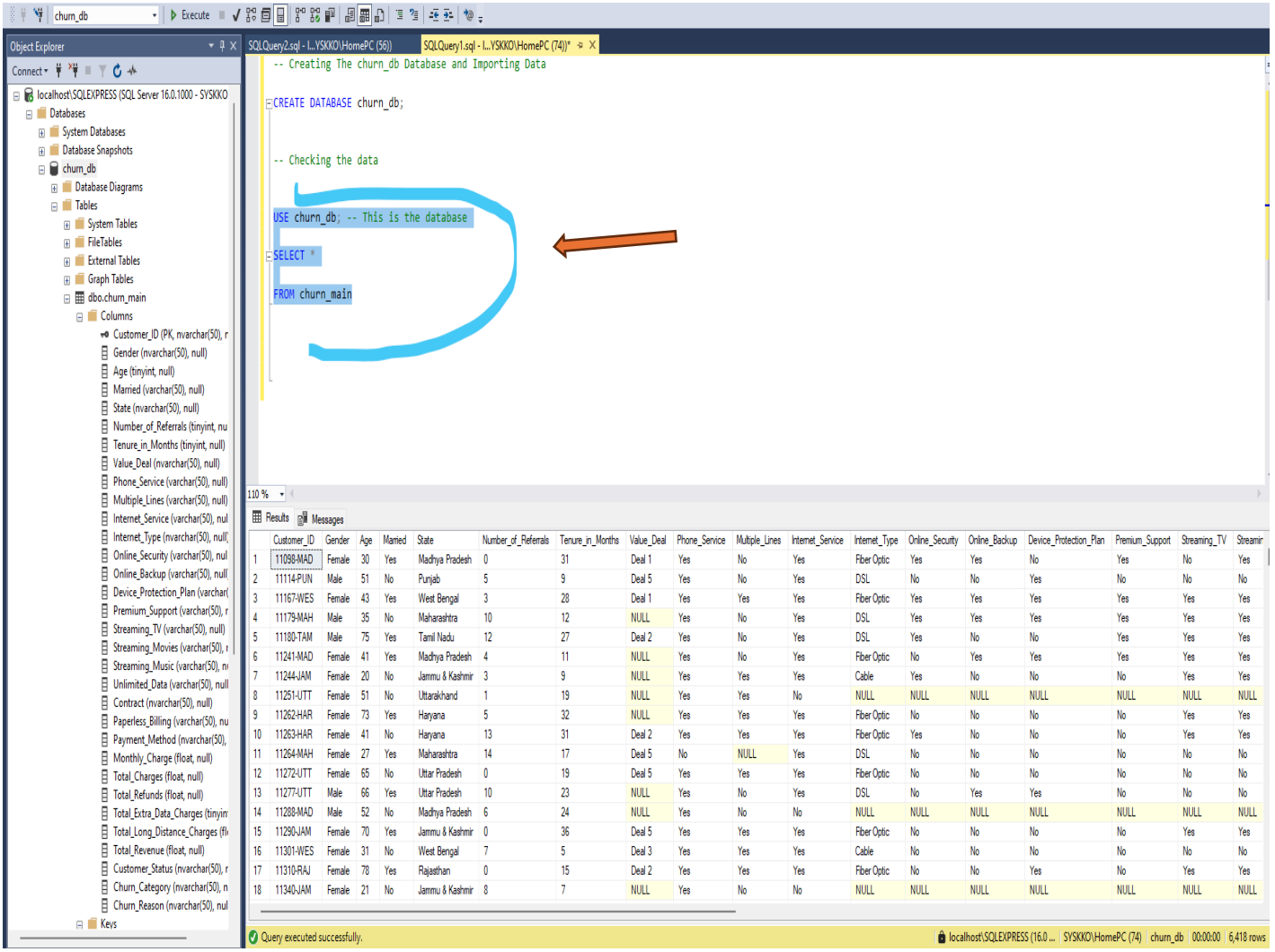
Finally, it got Imported and Operation successful.



Now the table has appeared on the explorer panel.

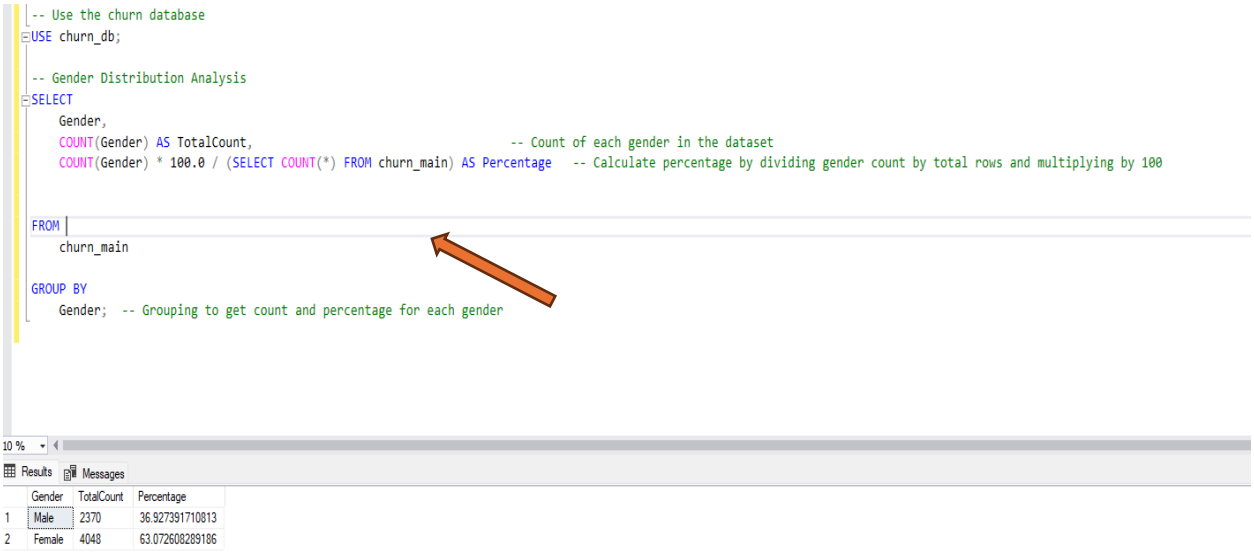
2.3. Data Exploration – Checking for Distinct Values

First, I had to check the columns to see that everything was alright and understand the distribution of data.



While looking at the gender column,

- I would like to know how many items are on the gender column
- What is the COUNT on each of these items.
- And I will need an additional column as an Alias which tells me the Percentage Distribution of this column.



```sql
-- Use the churn database
USE churn_db;

-- Gender Distribution Analysis
SELECT
    Gender,
    COUNT(Gender) AS TotalCount,                                -- Count of each gender in the dataset
    COUNT(Gender) * 100.0 / (SELECT COUNT(*) FROM churn_main) AS Percentage   -- Calculate percentage by dividing gender count by total rows and multiplying by 100


FROM
    churn_main

GROUP BY
    Gender;  -- Grouping to get count and percentage for each gender
```

10 %

Results | Messages

	Gender	TotalCount	Percentage
1	Male	2370	36.927391710813
2	Female	4048	63.072608289186

In this query, I explored the gender distribution within the churn_main table of the churn_db database. The goal is to understand the proportion of each gender represented in the dataset.

- COUNT(Gender) calculates the total number of records for each gender.
- The subquery (SELECT COUNT(*) FROM churn_main) returns the total number of rows in the table.
- Multiplying COUNT(Gender) / total by 100.0 converts the proportion into a **percentage representation** of each gender in the dataset.

//
//

```
Gender;  -- Grouping to get count and percentage for each gender
```

SQLQuery2.sql - localhost\SQLEXPRESS.churn_db (SYSKKO\HomePC (56))*

```
-- Use the churn database

USE churn_db;

-- Contract Type Distribution Analysis
SELECT
    Contract,
    COUNT(Contract) AS TotalCount,  -- Count of each contract type
    COUNT(Contract) * 1.0 / (SELECT COUNT(*) FROM churn_main) AS Percentage  -- Percentage of each contract type
FROM
    churn_main
GROUP BY
    Contract;  -- Group by contract type
```

110 %

Results | Messages

	Contract	TotalCount	Percentage
1	Month-to-Month	3286	0.511997507011
2	One Year	1413	0.220162044250
3	Two Year	1719	0.267840448737

In this query, I analyzed how different **contract types** are distributed among customers.

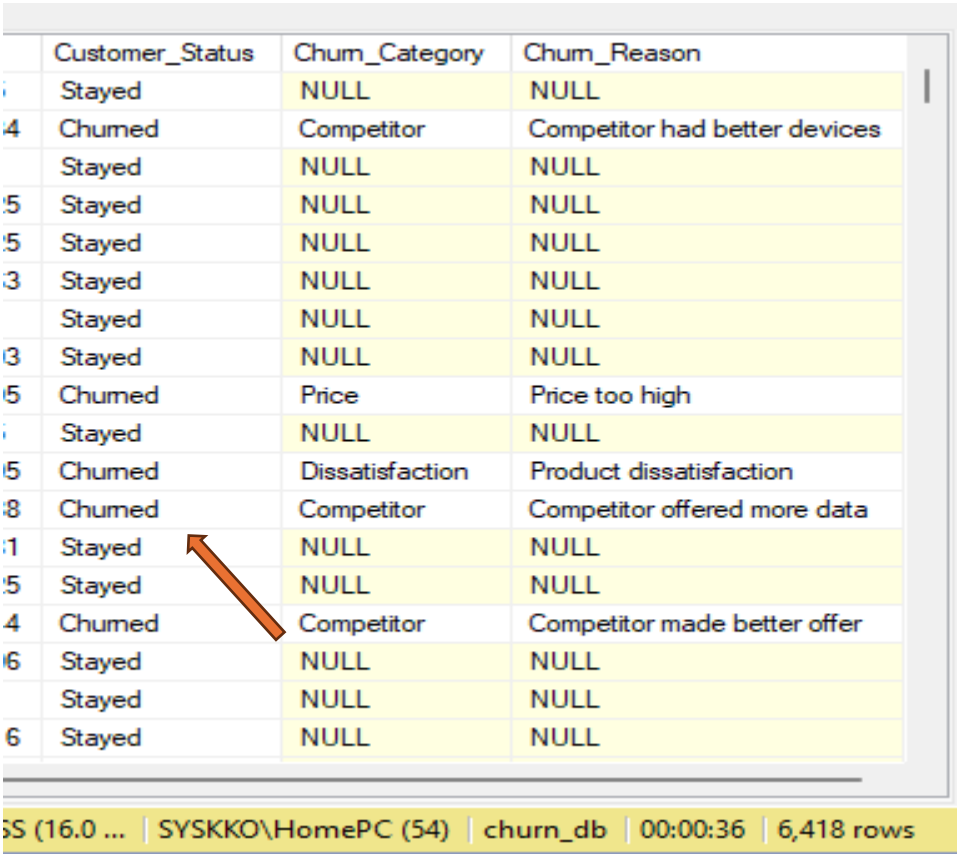
- COUNT(Contract) gives the total number of customers for each contract type.
- The subquery (SELECT COUNT(*) FROM churn_main) returns the total number of records in the dataset.
- By Multiplying by 1.0 ensures **decimal division**, so I get the proportion as a **decimal percentage.**

//
//

Now I will be exploring the Customer_Status column

This column tells me if a customer has stayed or churned(left)

Now I will be checking for the Customer_Status for who stayed, churned, or joined.



	Customer_Status	Churn_Category	Churn_Reason
	Stayed	NULL	NULL
4	Churned	Competitor	Competitor had better devices
	Stayed	NULL	NULL
5	Stayed	NULL	NULL
5	Stayed	NULL	NULL
3	Stayed	NULL	NULL
	Stayed	NULL	NULL
3	Stayed	NULL	NULL
5	Churned	Price	Price too high
	Stayed	NULL	NULL
5	Churned	Dissatisfaction	Product dissatisfaction
8	Churned	Competitor	Competitor offered more data
1	Stayed	NULL	NULL
5	Stayed	NULL	NULL
4	Churned	Competitor	Competitor made better offer
6	Stayed	NULL	NULL
	Stayed	NULL	NULL
6	Stayed	NULL	NULL

```sql
-- Now i will be checking for the Customer_Status for who stayed, churned, or Joined.

USE churn_db;  -- Using the churn database

-- Revenue breakdown by customer status
SELECT
    Customer_Status,
    COUNT(Customer_Status) AS TotalCount,  -- Number of customers per status
    SUM(Total_Revenue) AS TotalRevenue,  -- Total revenue generated by each status group
    SUM(Total_Revenue) * 100.0 / (SELECT SUM(Total_Revenue) FROM churn_main) AS RevenuePercentage  -- Percentage share of total revenue
FROM
    churn_main
GROUP BY
    Customer_Status;  -- Grouping results by customer status
```

110 %

Results | Messages

	Customer_Status	TotalCount	TotalRevenue	RevenuePercentage
1	Joined	411	49281.5598697662	0.253097281975677
2	Chumed	1732	3411960.5796299	17.5229426827105
3	Stayed	4275	16010148.2622757	82.2239600353138

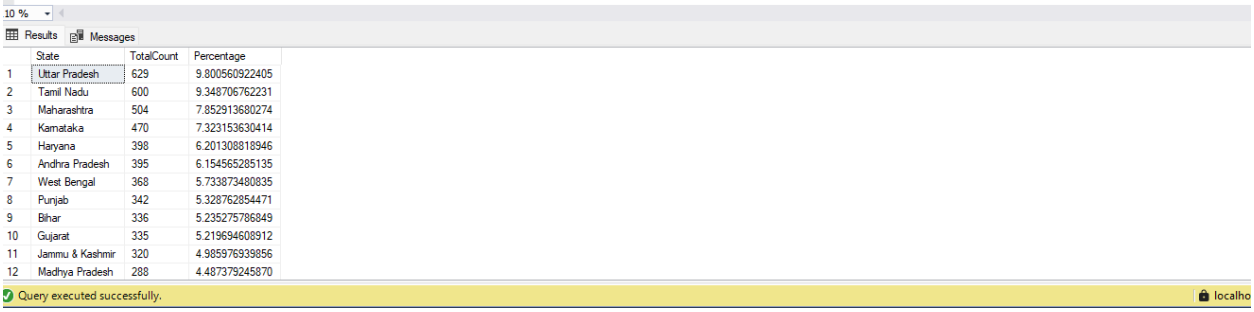
In this query, I analyzed how customer status (e.g., Churned, Joined, Stayed) affects revenue contribution:

- COUNT(Customer_Status) shows how many customers belong to each group.
- SUM(Total_Revenue) aggregates the revenue from each status group.
- Dividing each group's revenue by the total revenue and multiplying by 100 gives the **percentage contribution** of each group.

//
//

And finally, I will start exploring the State column and see how many states represent what percentage of values

```sql
-- Finally i will start to explore the State column to see how many states
-- Represents what percentage the values shows.

USE churn_db;  -- Using the churn database

SELECT State,
       COUNT(State) AS TotalCount,                                          -- Count the number of customers in each state
       COUNT(State) * 100.0 / (SELECT COUNT(*) FROM churn_main) AS Percentage           -- Calculate the percentage of total customers for each state
FROM
    churn_main
GROUP BY
    State                                  -- Group the data by state to calculate the count and percentage for each state
ORDER BY
    Percentage DESC;                       -- Order the results by percentage in descending order (highest percentage first)
```



10 %

Results | Messages

	State	TotalCount	Percentage
1	Uttar Pradesh	629	9.800560922405
2	Tamil Nadu	600	9.348706762231
3	Maharashtra	504	7.852913680274
4	Karnataka	470	7.323153630414
5	Haryana	398	6.201308818946
6	Andhra Pradesh	395	6.154565285135
7	West Bengal	368	5.733873480835
8	Punjab	342	5.328762854471
9	Bihar	336	5.235275786849
10	Gujarat	335	5.219694608912
11	Jammu & Kashmir	320	4.985976939856
12	Madhya Pradesh	288	4.487379245870

Query executed successfully. localho

In this query, I analyzed how the **state** of customers (e.g., Churned, Joined, Stayed) affects their distribution:

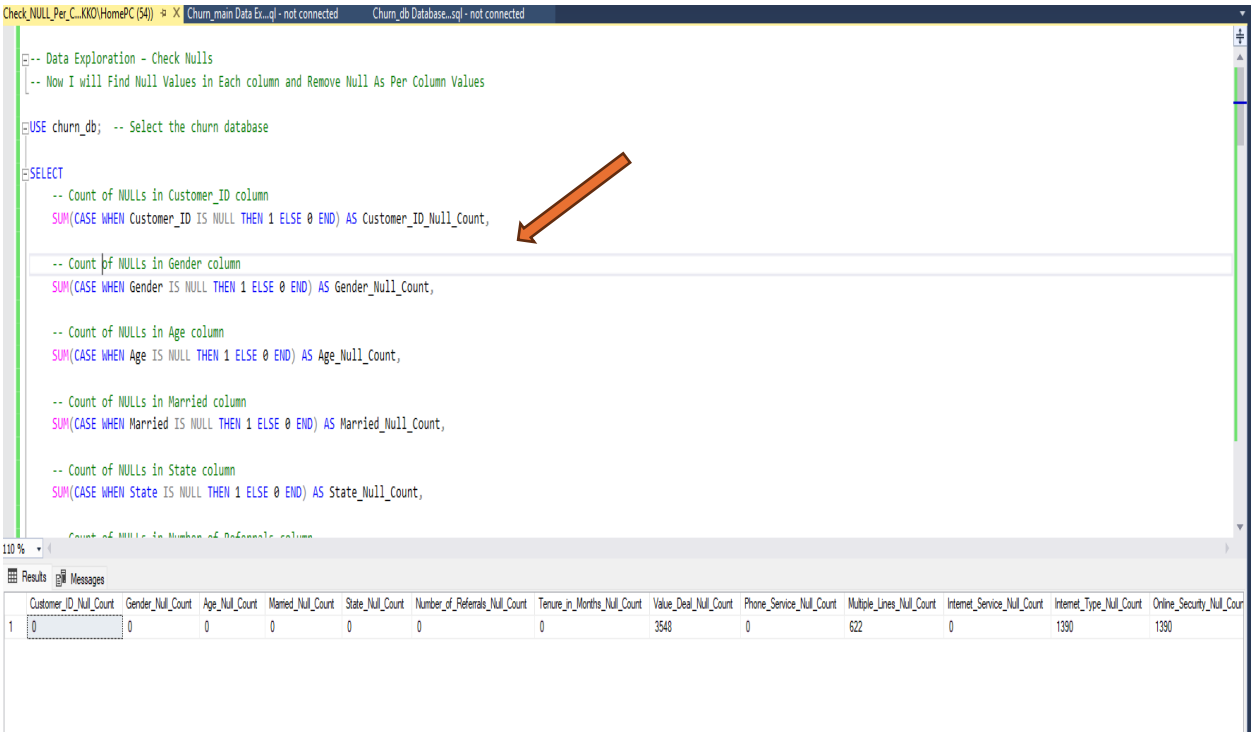
- COUNT(State) shows how many customers belong to each **state** (such as 'Active', 'Churned', etc.).
- The COUNT(State) * 100.0 / (SELECT COUNT (*) FROM churn_main) calculates the percentage of total customers that belong to each state. It compares the count of customers in each state to the total number of customers in the **churn_main** table.
- The **ORDER BY Percentage DESC** sorts the results so that states with the highest percentage of customers appear at the top of the list.

//
//

2.4 Data Exploration – Data Cleaning - Null Value Checking

Now I will Find Null Values in Each column and Remove Null as Per Column Values.

N/B: NULL can create a lot of problems if I ignore it, so I will check the columns for Missing NULL



```
-- Data Exploration - Check Nulls
-- Now I will Find Null Values in Each column and Remove Null As Per Column Values

USE churn_db;  -- Select the churn database

SELECT
    -- Count of NULLs in Customer_ID column
    SUM(CASE WHEN Customer_ID IS NULL THEN 1 ELSE 0 END) AS Customer_ID_Null_Count,

    -- Count of NULLs in Gender column
    SUM(CASE WHEN Gender IS NULL THEN 1 ELSE 0 END) AS Gender_Null_Count,

    -- Count of NULLs in Age column
    SUM(CASE WHEN Age IS NULL THEN 1 ELSE 0 END) AS Age_Null_Count,

    -- Count of NULLs in Married column
    SUM(CASE WHEN Married IS NULL THEN 1 ELSE 0 END) AS Married_Null_Count,

    -- Count of NULLs in State column
    SUM(CASE WHEN State IS NULL THEN 1 ELSE 0 END) AS State_Null_Count,
```

Results | Messages

	Customer_ID_Null_Count	Gender_Null_Count	Age_Null_Count	Married_Null_Count	State_Null_Count	Number_of_Referrals_Null_Count	Tenure_in_Months_Null_Count	Value_Deal_Null_Count	Phone_Service_Null_Count	Multiple_Lines_Null_Count	Internet_Service_Null_Count	Internet_Type_Null_Count	Online_Security_Null_Cour
1	0	0	0	0	0	0	0	3548	0	622	0	1390	1390

This query checks for **missing (NULL) values** in each column of the churn_main table.

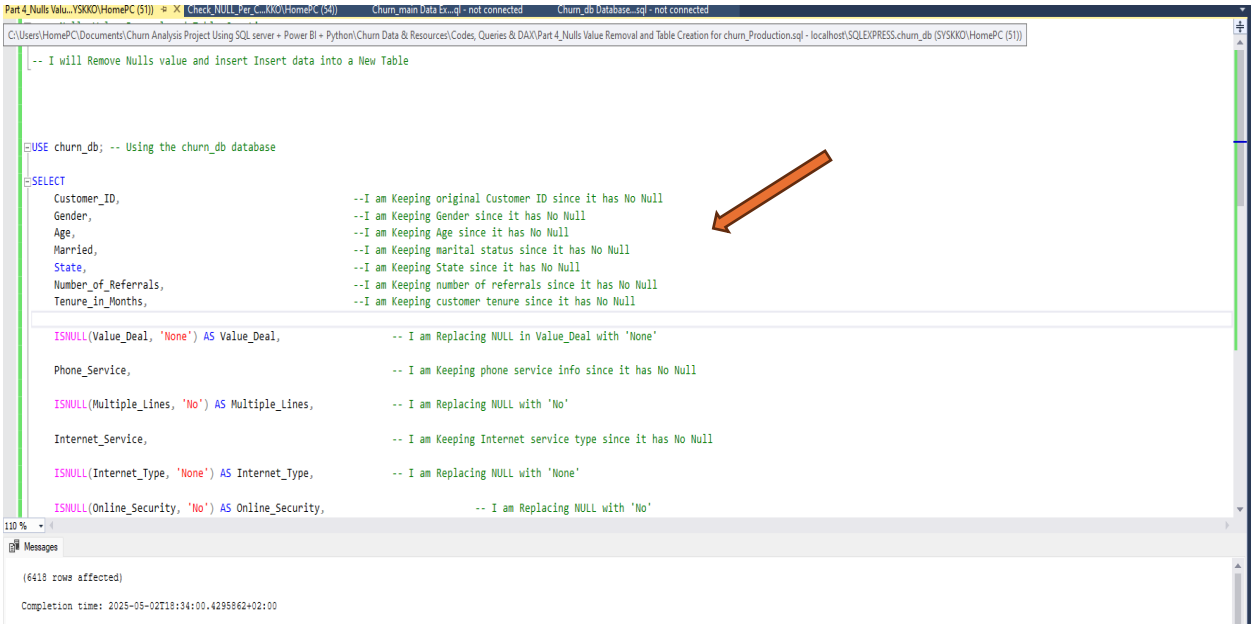
- It counts how many NULLs are in each column using SUM(CASE WHEN column IS NULL THEN 1 ELSE 0 END).
- This result tells me which columns have missing data.
- Is First Assigning NULL Values As 1 and else 0 and saves it As an Alias

2.5 Data Exploration – Data Cleaning - Null Value Removal

Now the next Step, is to Remove this NULLs Values

I will be replacing these Nulls values and inserting the data into a new table which will be a Production Table.

This table will be used as the source of all upcoming steps



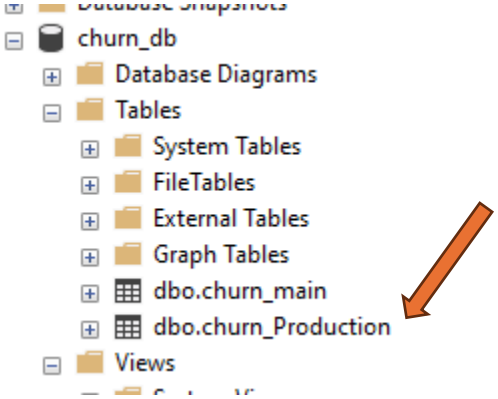
```
-- I will Remove Nulls value and insert Insert data into a New Table



USE churn_db; -- Using the churn_db database

SELECT
    Customer_ID,                                --I am Keeping original Customer ID since it has No Null
    Gender,                                     --I am Keeping Gender since it has No Null
    Age,                                        --I am Keeping Age since it has No Null
    Married,                                    --I am Keeping marital status since it has No Null
    State,                                      --I am Keeping State since it has No Null
    Number_of_Referrals,                        --I am Keeping number of referrals since it has No Null
    Tenure_in_Months,                           --I am Keeping customer tenure since it has No Null

    ISNULL(Value_Deal, 'None') AS Value_Deal,               -- I am Replacing NULL in Value_Deal with 'None'

    Phone_Service,                              -- I am Keeping phone service info since it has No Null

    ISNULL(Multiple_Lines, 'No') AS Multiple_Lines,         -- I am Replacing NULL with 'No'

    Internet_Service,                           -- I am Keeping Internet service type since it has No Null

    ISNULL(Internet_Type, 'None') AS Internet_Type,         -- I am Replacing NULL with 'None'

    ISNULL(Online_Security, 'No') AS Online_Security,                   -- I am Replacing NULL with 'No'
```

```
(6418 rows affected)

Completion time: 2025-05-02T18:34:00.4295862+02:00
```

In this step, I performed data cleaning by **removing or handling NULL values** and saved the cleaned results into a new table called churn_Production:

- I used the ISNULL() function to **replace NULLs** with default values for specific columns:
    - For service-related columns like Multiple_Lines, Online_Security, etc., I replaced NULL with 'No'.

    - For Value_Deal and Internet_Type, I replaced NULL with 'None'.
    - For churn-related fields like Churn_Category and Churn_Reason, I replaced NULL with 'Others'.
- I kept columns like Customer_ID, Gender, Age, Contract, and others **unchanged** because they **didn't contain any NULLs**.
- Finally, I saved the cleaned data onto a **new table** called churn_Production to keep the original data intact and work with a clean version going forward.



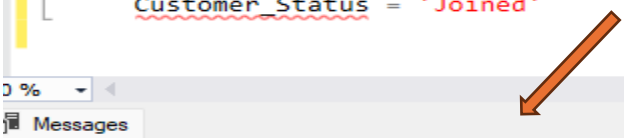
Churn_Production has now appeared on the explorer panel.

//
//

2.6 Data Exploration – View for Visual Table

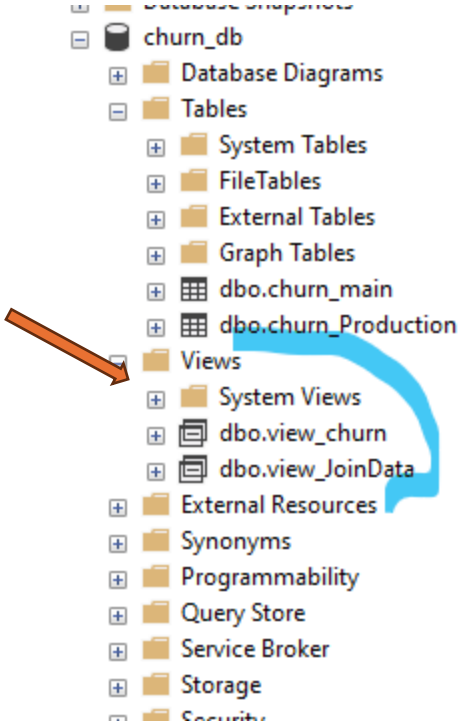
In a Database, the main purpose of VIEW is to create a Virtual Table, so as this view will not have any physical Presence on the Database itself, which I am being careful of, and it will provide me with the Output when a Query calls for it

```
USE churn_db;


CREATE VIEW view_churn as

SELECT *

FROM
      churn_Prod uction

WHERE
     Customer_Status in ('Churned', 'Stayed')



CREATE VIEW view_JoinData as

SELECT *

FROM
      churn_Production

WHERE
     Customer_Status = 'Joined'
```



```
Messages
Commands completed successfully.

Completion time: 2025-05-02T19:29:48.1430587+02:00
```

The Dropdown message shows success and if you look at the left-hand corner of the explorer panel, you will see the view option



Now that's my virtual table

Now all data from the churn_Production table will be on the view

Now I will move into the Power BI Visualization Tool, which is a powerful business intelligence platform used to create interactive dashboards and visual reports from the cleaned and processed data.

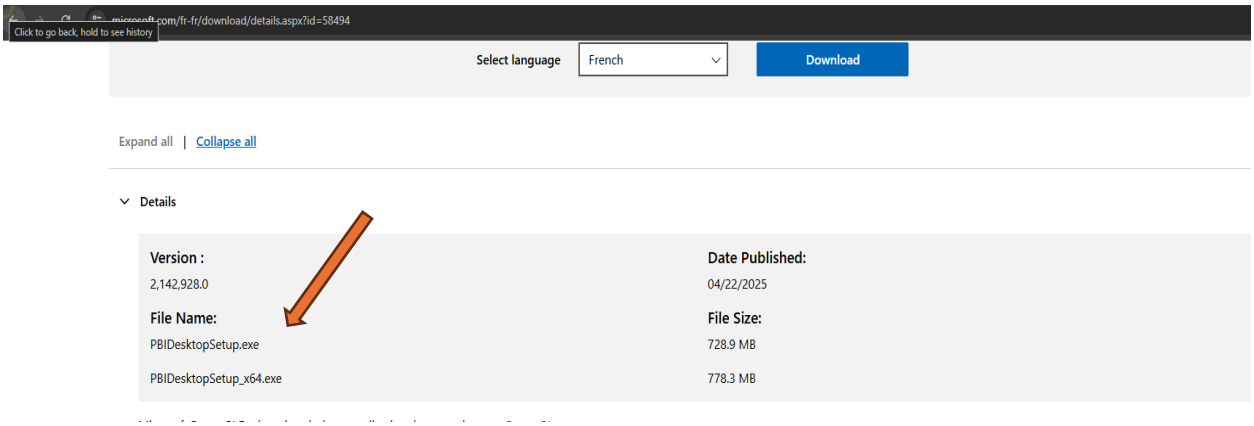
I'm done with the SQL aspect of this Project.

//
//

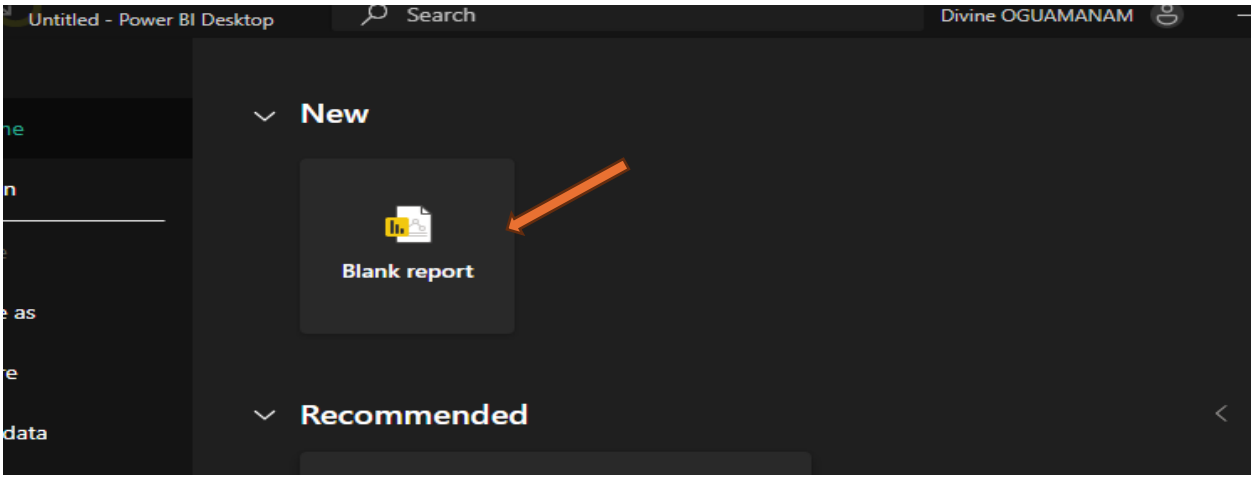
3 Power BI Transform

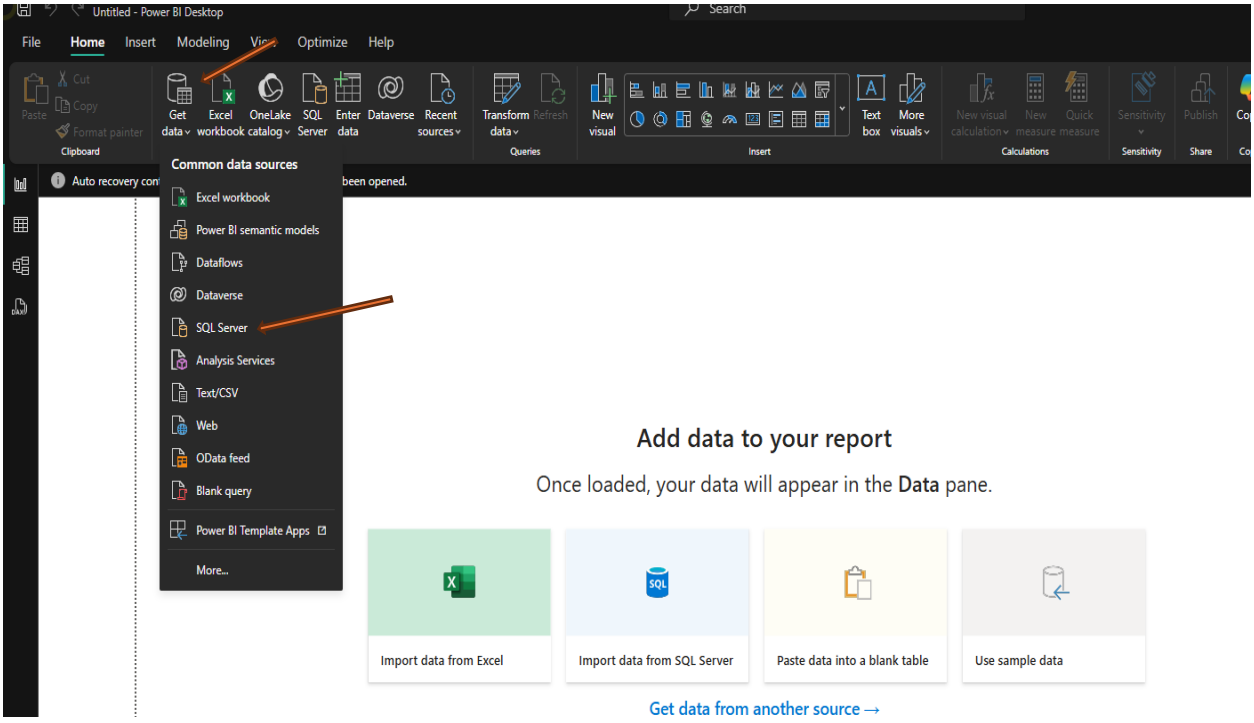
I will be using the Microsoft Power BI Desktop Application for my visualization.

https://www.microsoft.com/fr-fr/download/details.aspx?id=58494

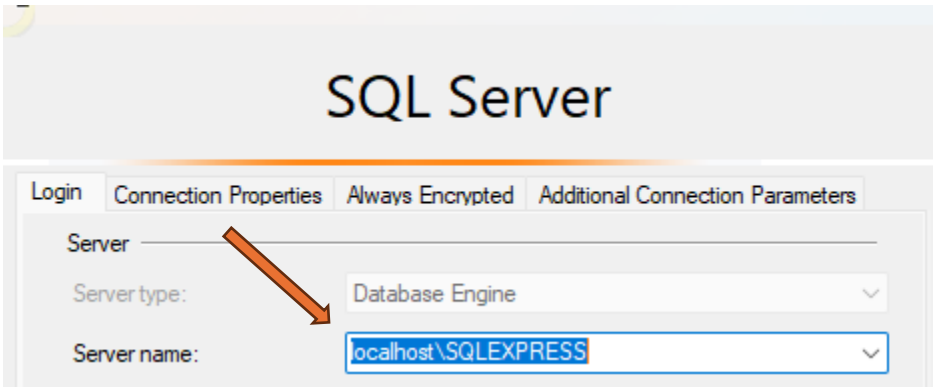


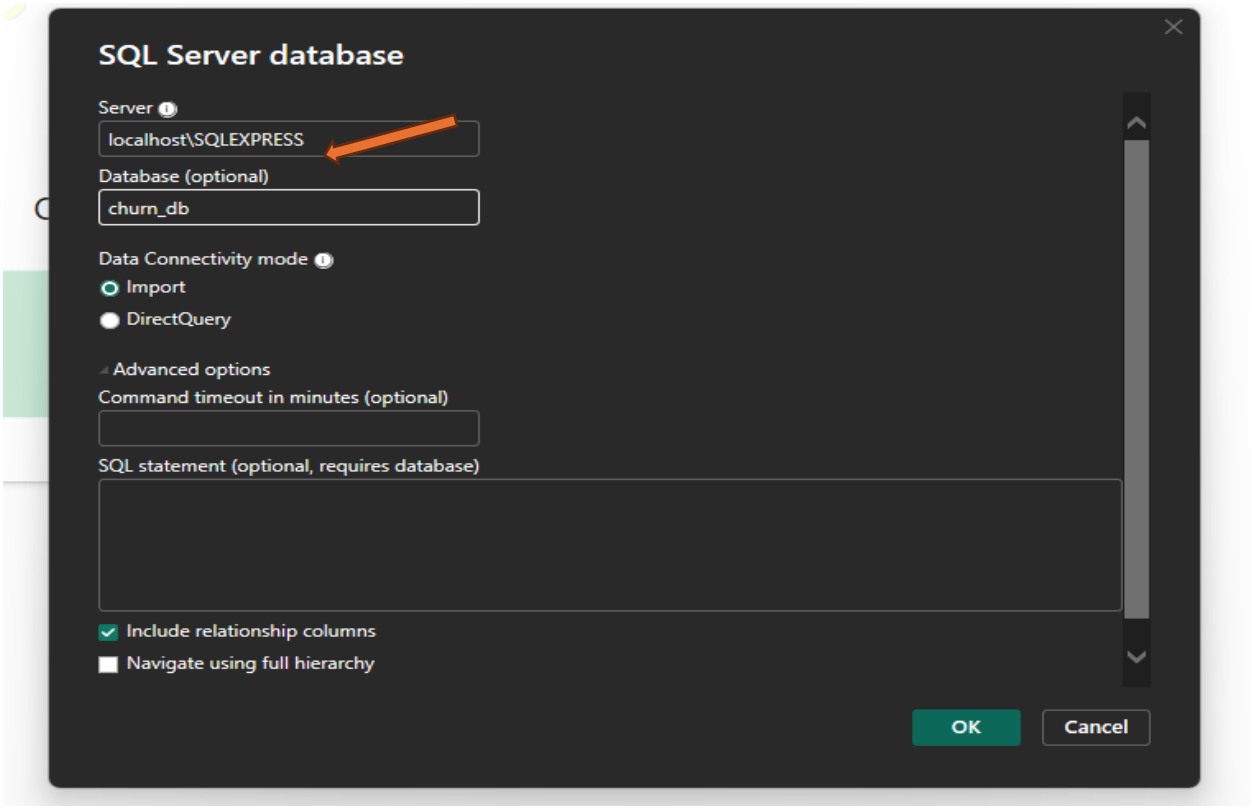
After completing the download and installation, I will open a blank report. Since I want to import my SQL queries, I will click the **Get Data** button in the top left corner and select the **SQL Server** option.



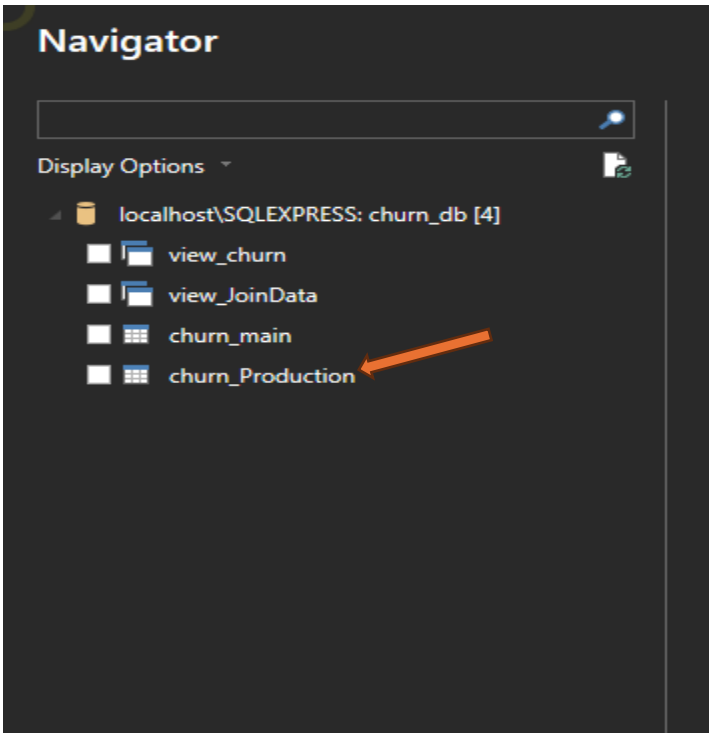


Now I will copy my server's name and import on the server and click on import, which is on my connection page of my SQL server Application.

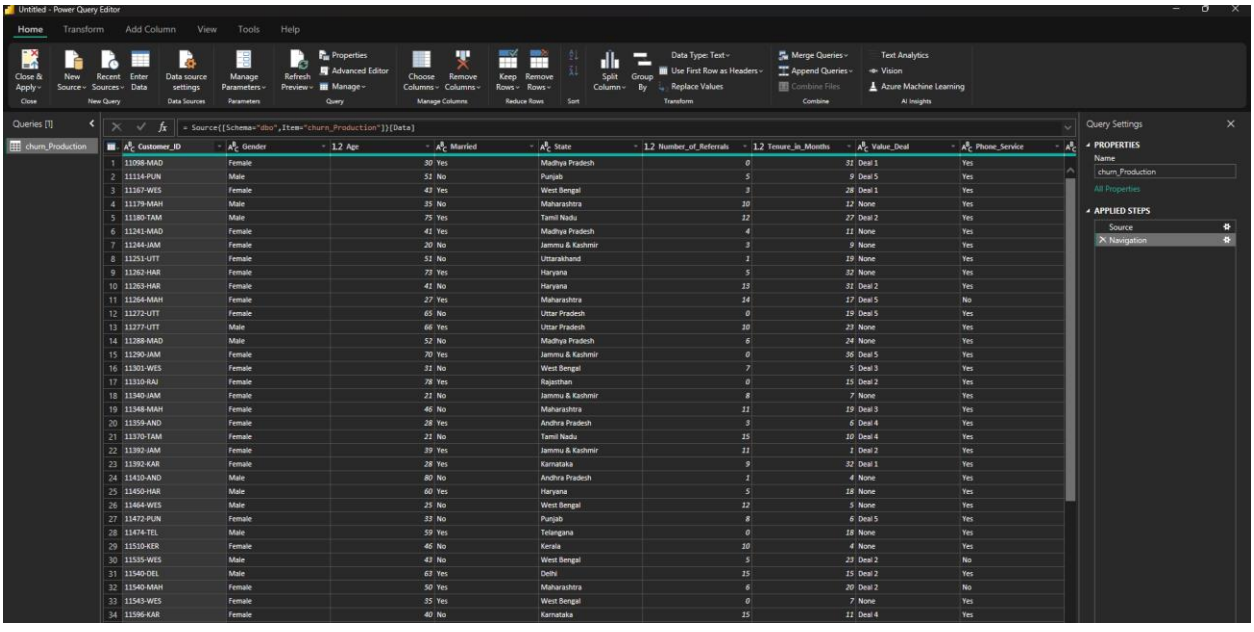




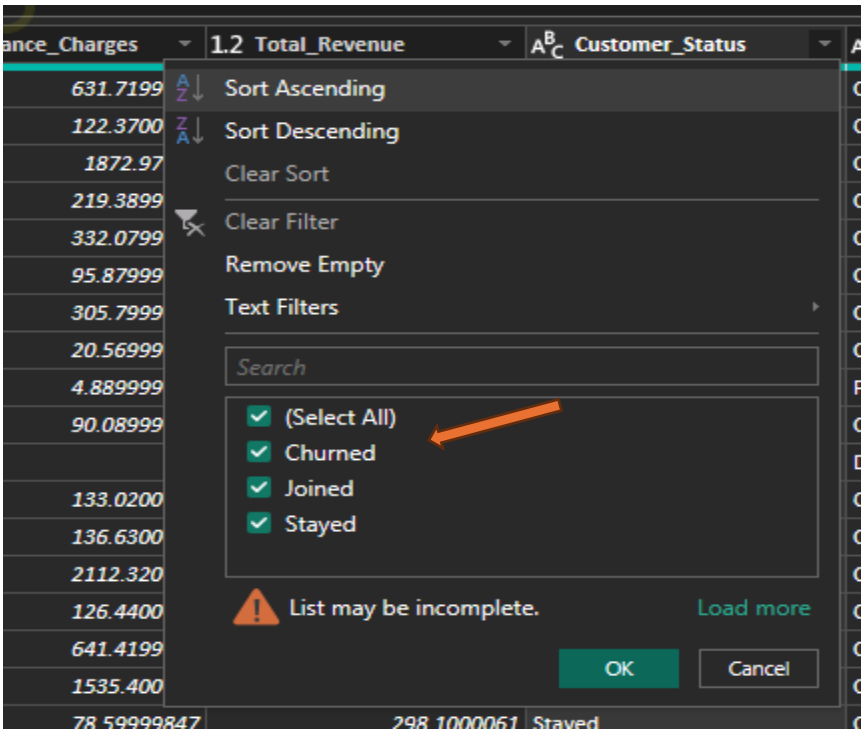
Then I will click on OK, which will prompt me to the Navigator panel



Now I will select the churn_production and click on transform Data which will prompt me to the power query editor where I will transform and edit this data.

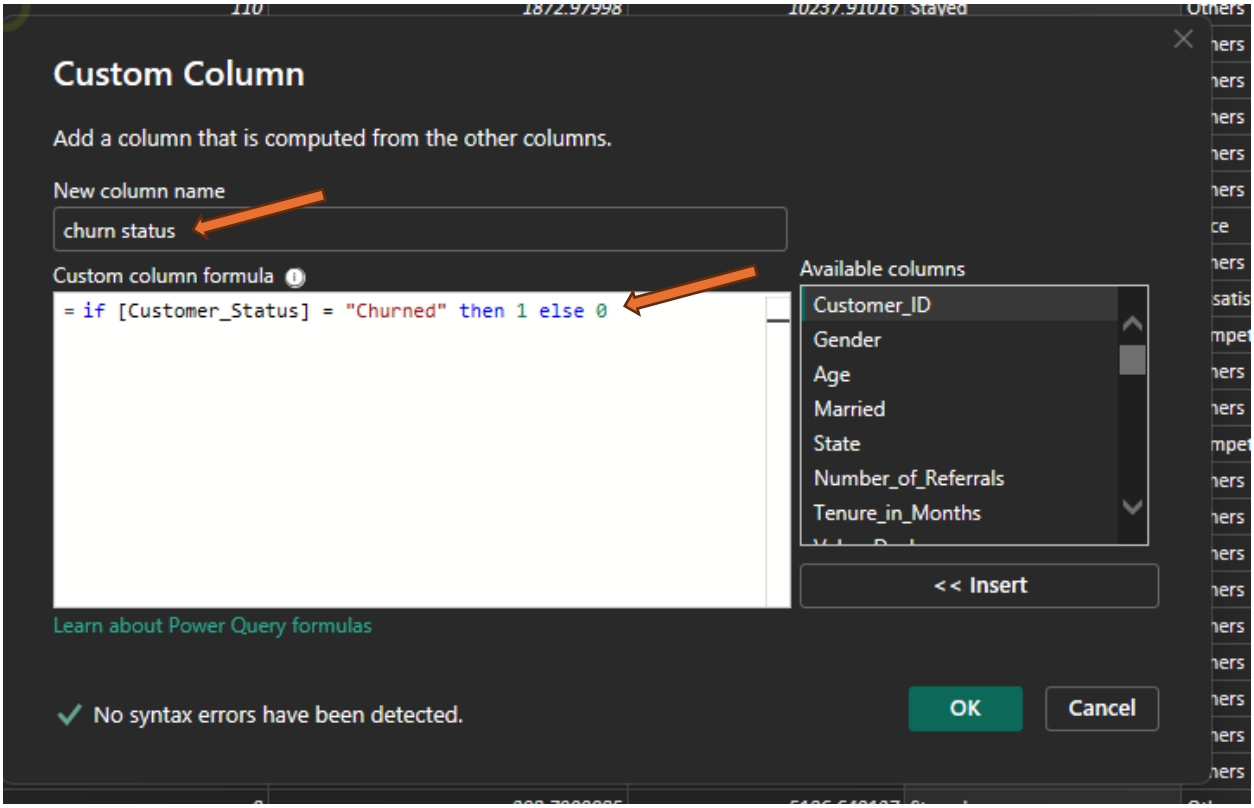


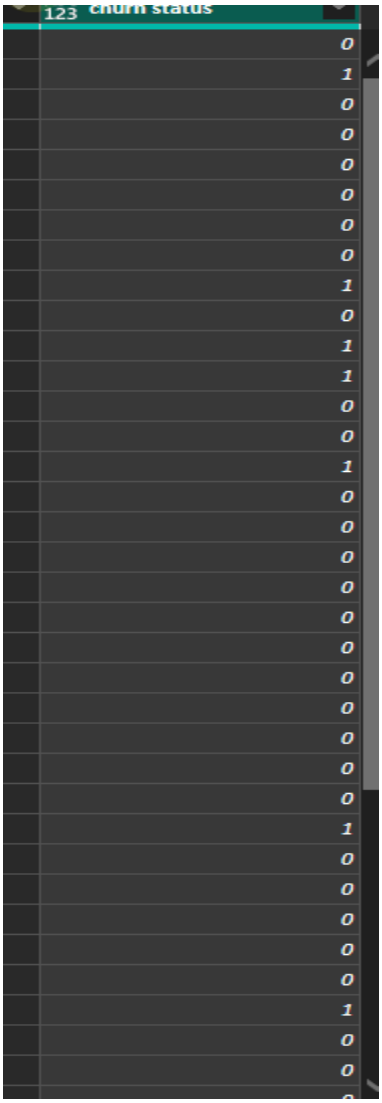
Now I will need to create a couple of columns based on certain conditions.



Looking at the Customer_Status column which has three values which are stayed, churned, and joined. I would like to create another column which will be a number column which defines whether a certain row is for churned or not

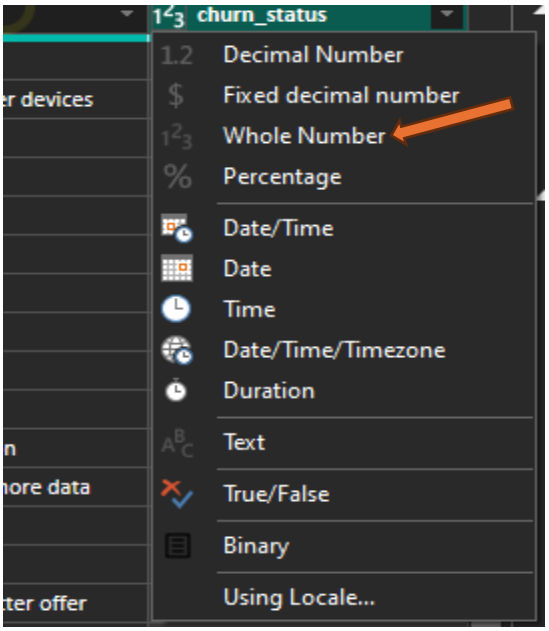
- I will click on add column then custom column and call it churn status
- And add this DAX formulae to it If [Customer_Status] = “Churned” then 1 else 0


Custom Column
Add a column that is computed from the other columns.
New column name
churn status
Custom column formula
= if [Customer_Status] = "Churned" then 1 else 0
Available columns
Customer_ID
Gender
Age
Married
State
Number_of_Referrals
Tenure_in_Months
<< Insert
Learn about Power Query formulas
No syntax errors have been detected.
OK
Cancel



Having been added, I renamed the column to churn_status

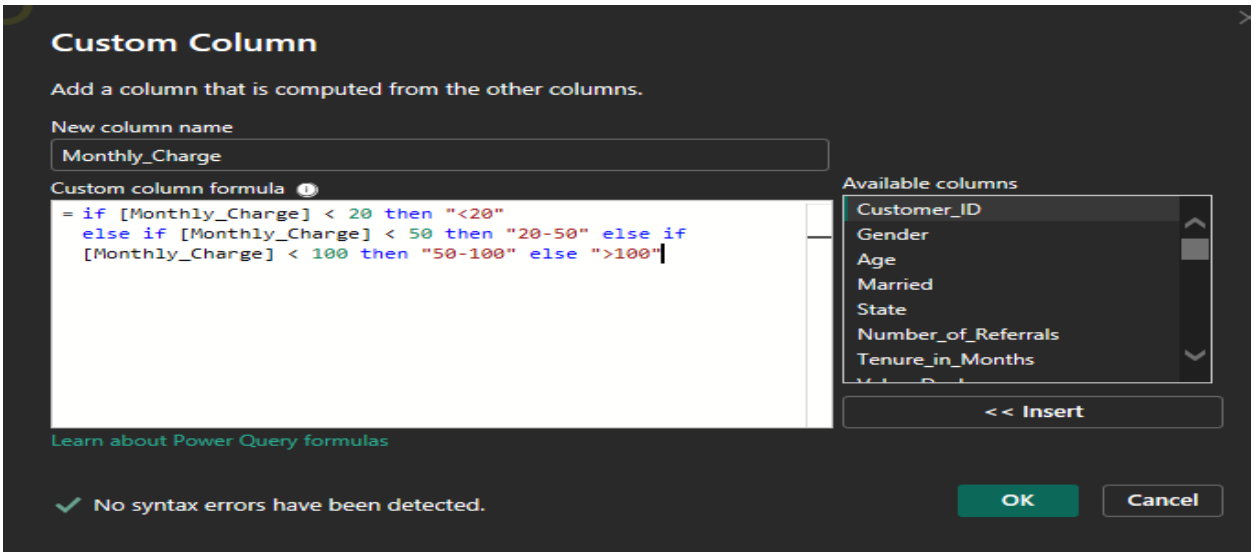
Now I have the column I will have to change it from string text to whole number



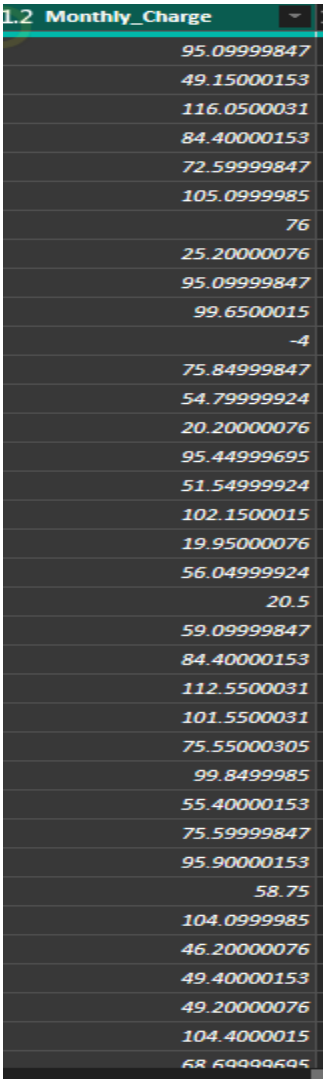
I will be using this column to do some calculations

Now I will be creating another column called Monthly_ChargeStatus

Using this formula if [Monthly_charge] < 20 then “<20” else if [Monthly_Charge] < 50 then “20-50” else if [Monthly_Charge] < 100 then “50-100” else “>100”

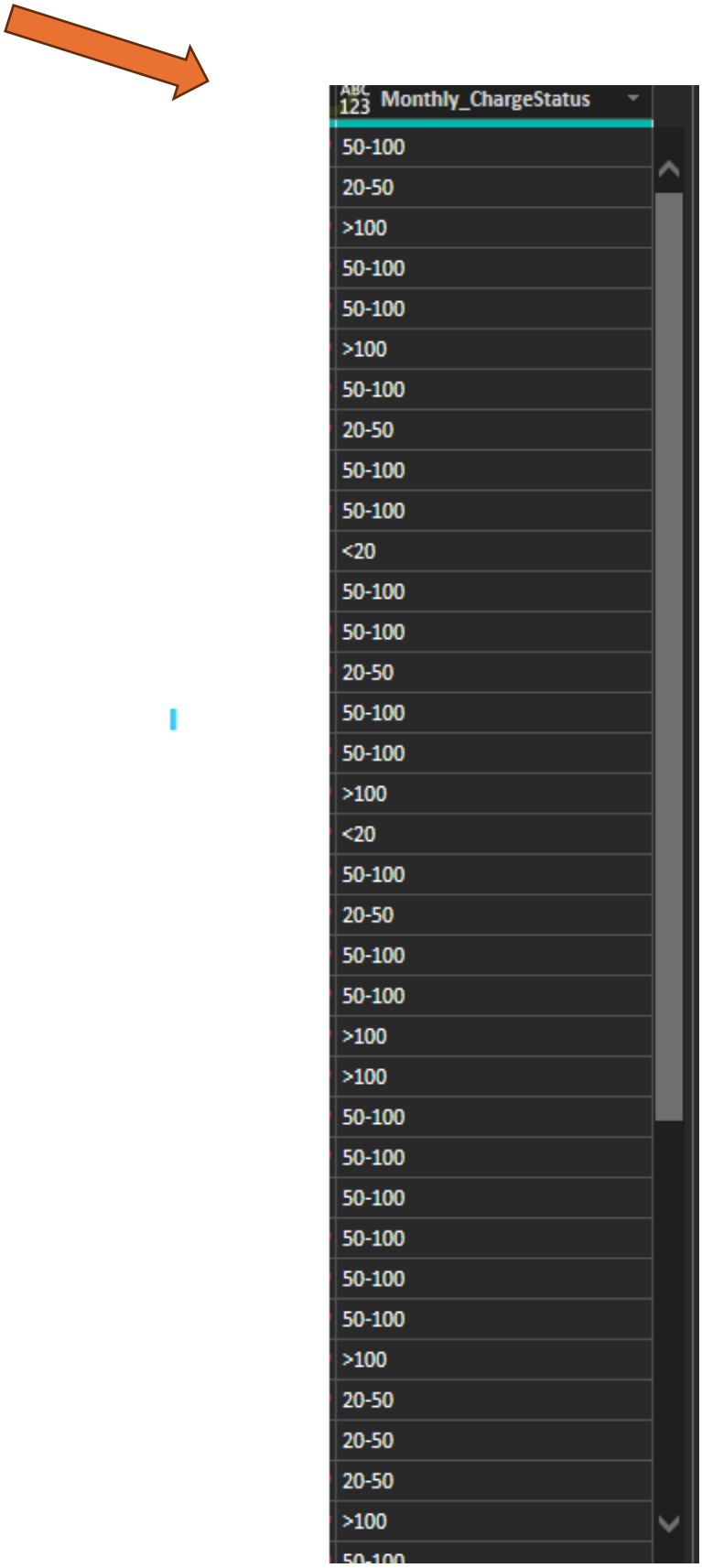


The reason for this is that the **Monthly_Charge** column contains a wide range of values, which can make it difficult to process effectively during the visualization stage. To address this, I created a new column called **Monthly_ChargeStatus** that categorizes the values into defined ranges, making the data easier to analyze and visualize.

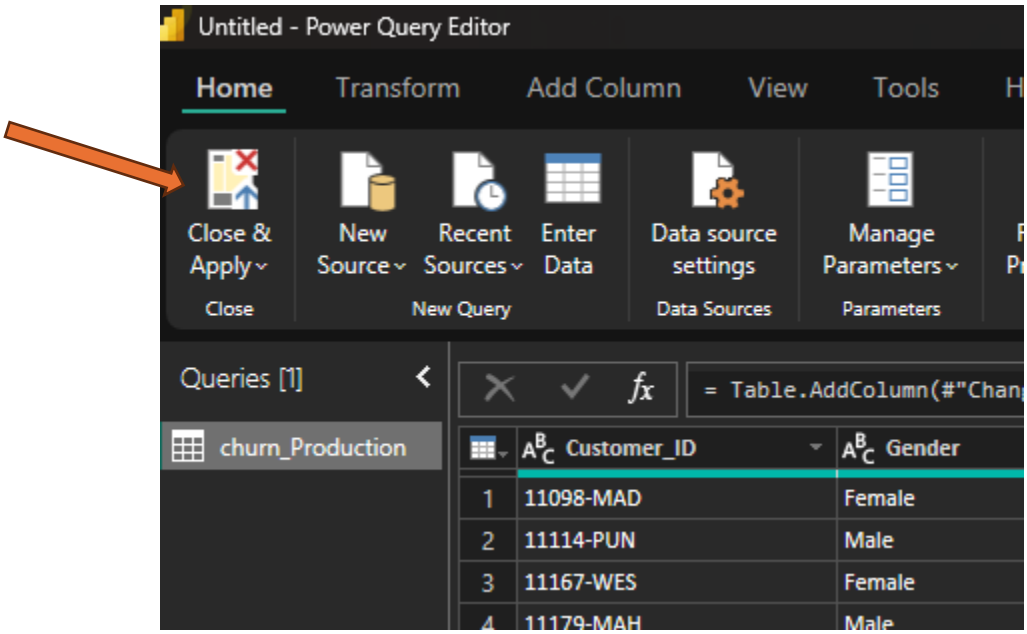


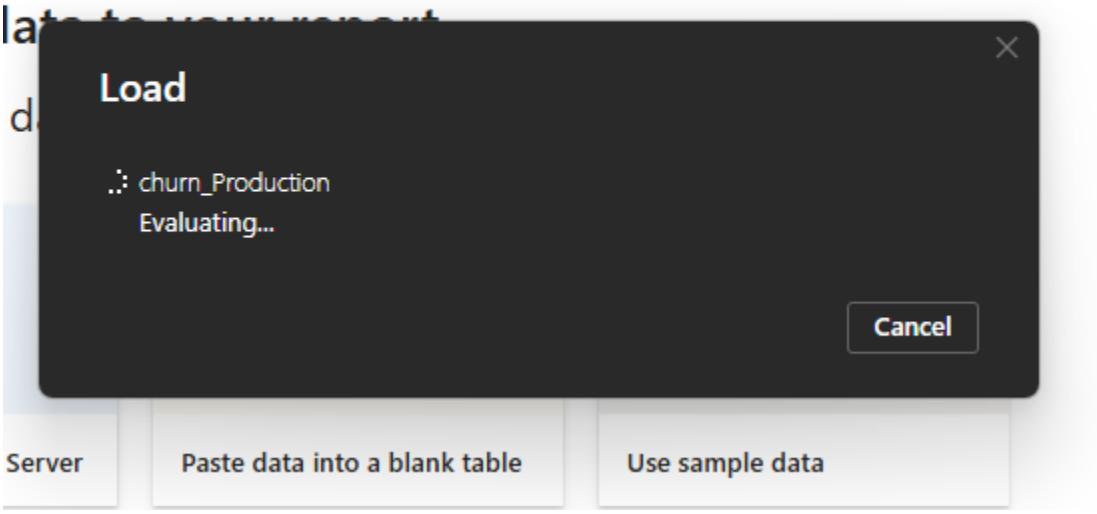
This is the numerous values on it

Now down below is the new range column


ABC 123 Monthly_ChargeStatus
50-100
20-50
>100
50-100
50-100
>100
50-100
20-50
50-100
50-100
<20
50-100
50-100
20-50
50-100
50-100
>100
<20
50-100
20-50
50-100
50-100
>100
>100
50-100
50-100
50-100
50-100
50-100
50-100
>100
20-50
20-50
20-50
>100

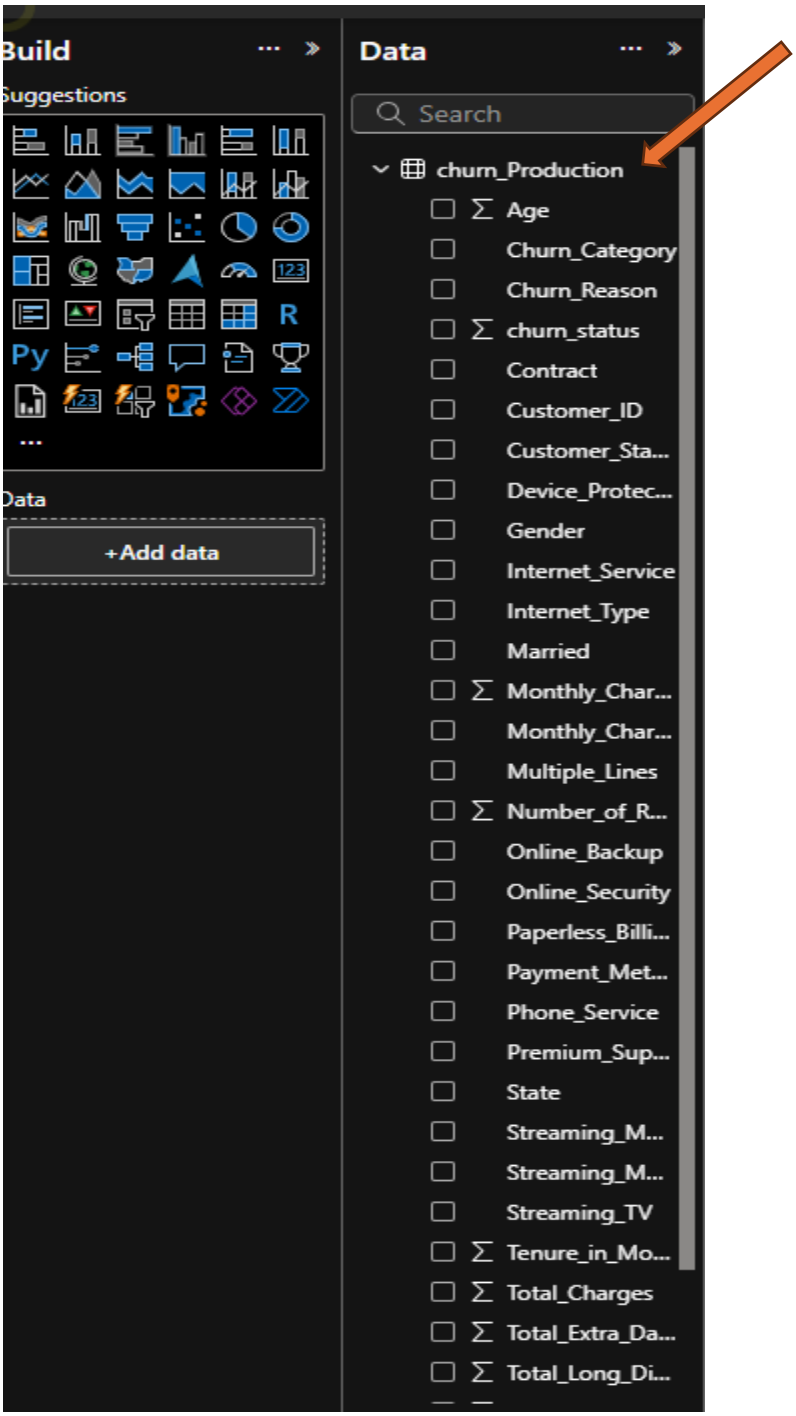
Now I will close and apply changes to the Data





Is loading into the worksheet

Now the Data pane by the right corner is showing the churn_Production



At this point I had to save my file by pressing CTRL S and inputting the name as Chun Analysis

//
//

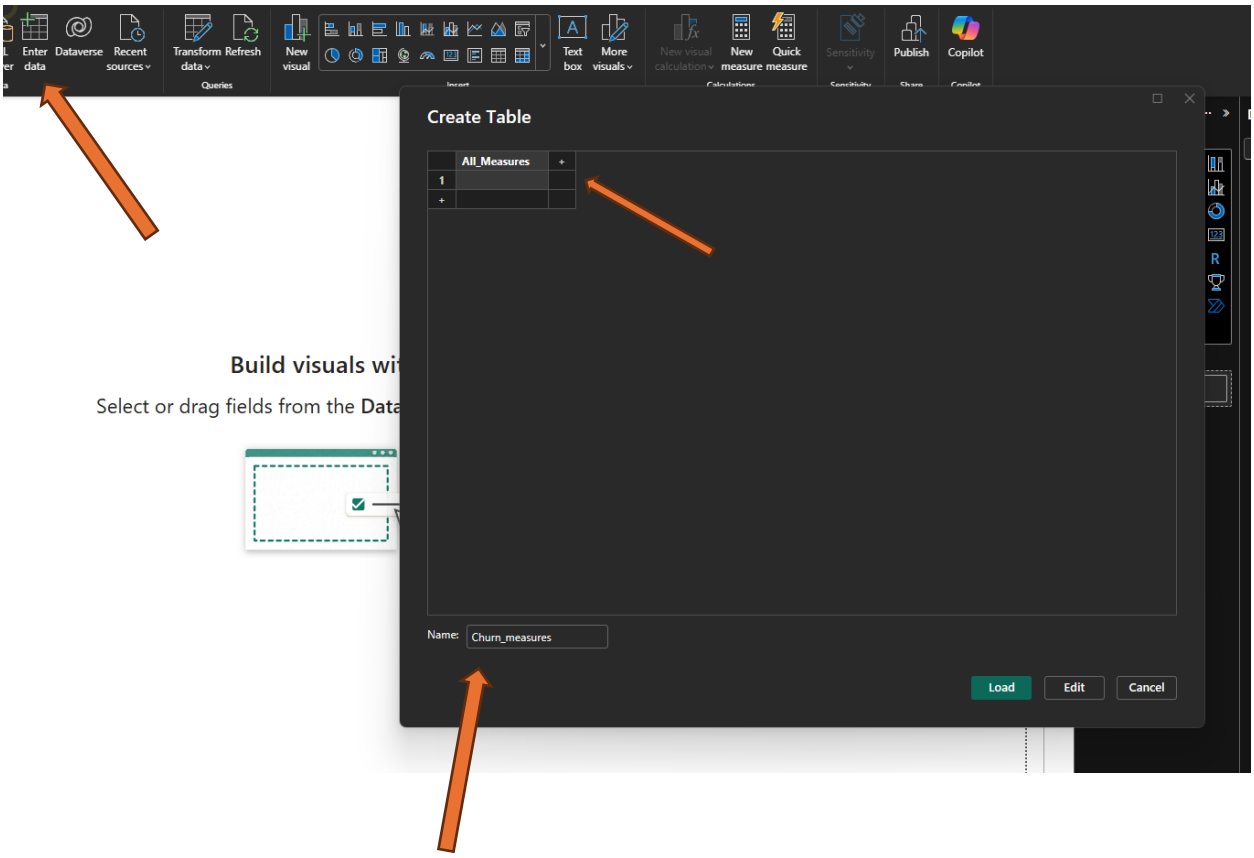
3.1 My Summary Page Blueprint

This is a list of visuals that will be on my summary page dashboard

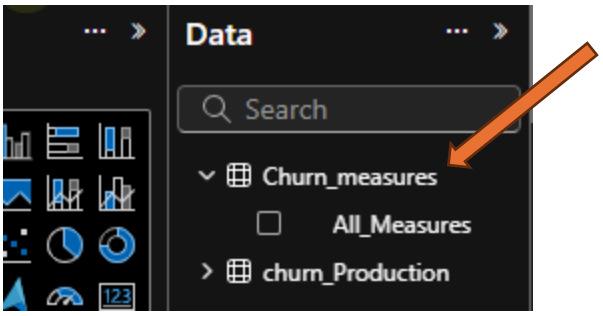
- Total Customers
- New joiners
- Total churn
- Churn Rate
- Gender
- Age
- Geographic
- Services
- Payment Method & Contract
- Tenure
- Churn Distribution
- Services

Before I start to create charts, I would like to create a few measures and keep them separate from the main tables, that way is easy for me to edit the measure in case is required in the future.

- I am going to create a dummy table where I will insert my measures manually to keep all my measures inside of it

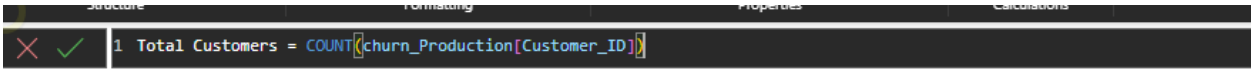


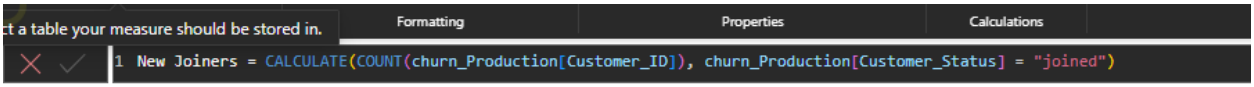
- Now I will click on Enter Data and this page will pop up, this is where I will input my table name as Churn_Measures and column as All_Measures

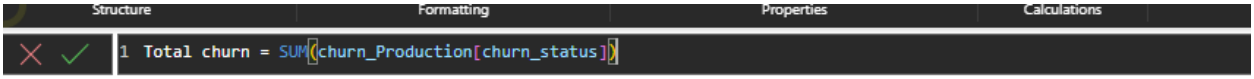


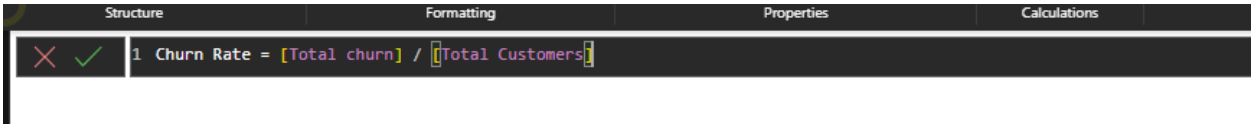
Now it has appeared on the Data pane.

I will create new different measures by clicking right beside the churn_measure and clicking on new measure and write several Dax formulae's









Now I am done with the measures I will be visualizing with; it is time for me to save my files so I wouldn't lose my project CTRL S

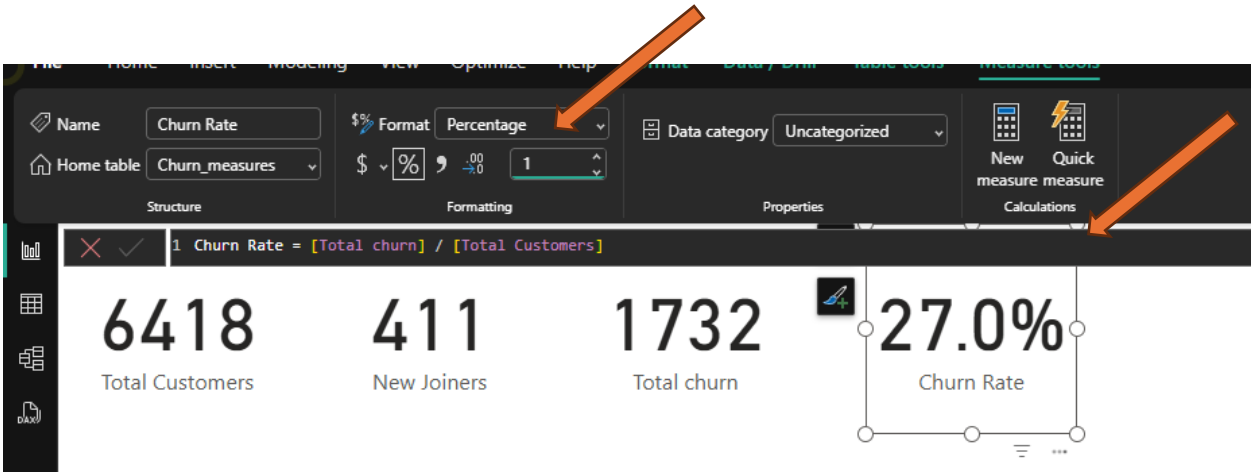
//
//

3.2 VISUALIZATION

First visuals I must create are

- Total Customers
- New joiners
- Total Churn
- Churn Rate

I will drop in few Cards from the visualization pane with its measures



I changed the churn rate to percentage 1 decimal value, and I did same with the comma as well for both 6418 and 1732.

Now I am done with the Metrics, I will be diving into the individual categories available which are

- Gender
- Age
- Geographic

- Payment Method & Contract
- Tenure
- Churn Distribution
- Services

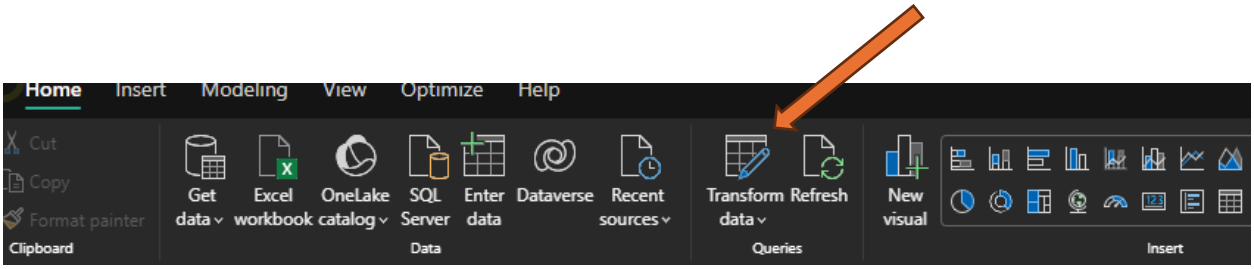
I am going to use the donut chart for gender by going to the churn_production table and click on gender and then the donut chart.

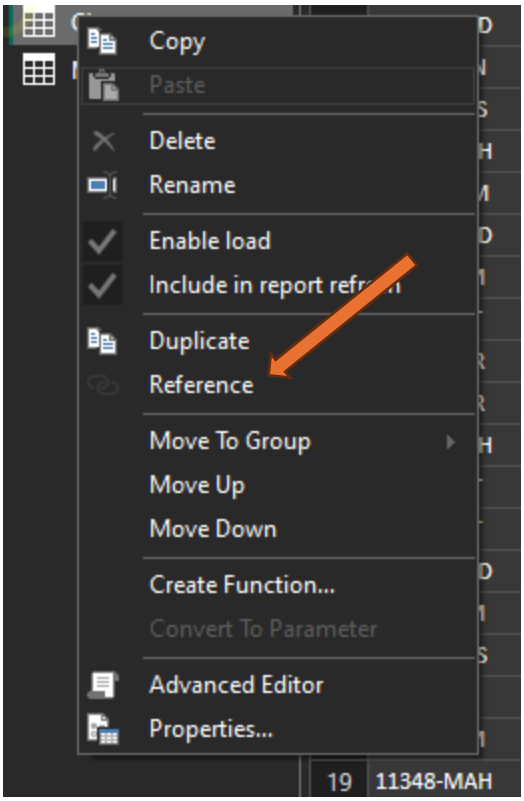
I will drag total churn from the All_Measure table into the donut chart

Now I will do some formatting

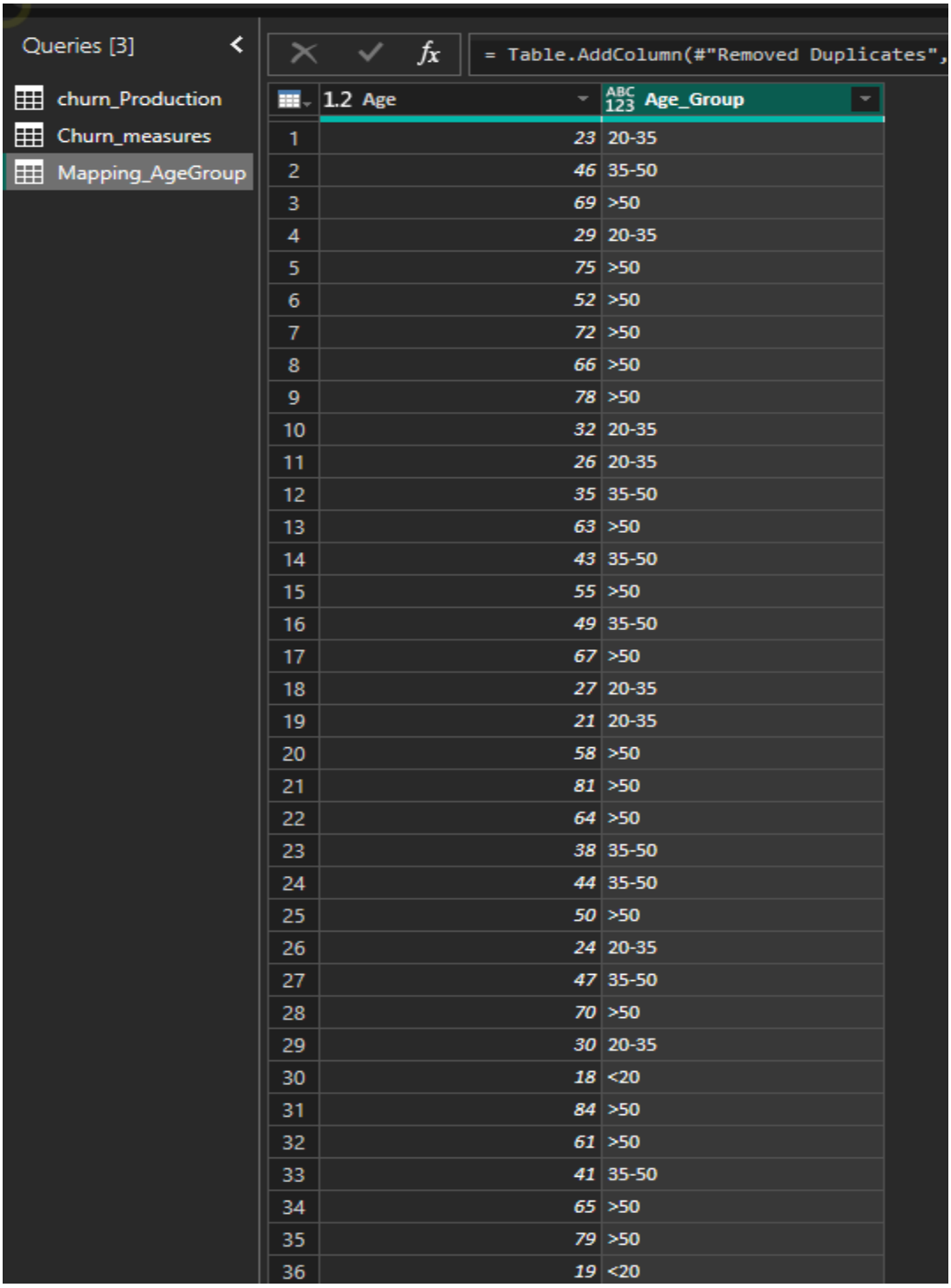
As for the Age column, since there are a lot of age digits clustered all around, it wouldn't be easy to understand it when it gets visualized.

So, I will head to transform Data which will take me to power query whereby I will reference the churn_production table.





Changed it to Mapping_AgeGroup



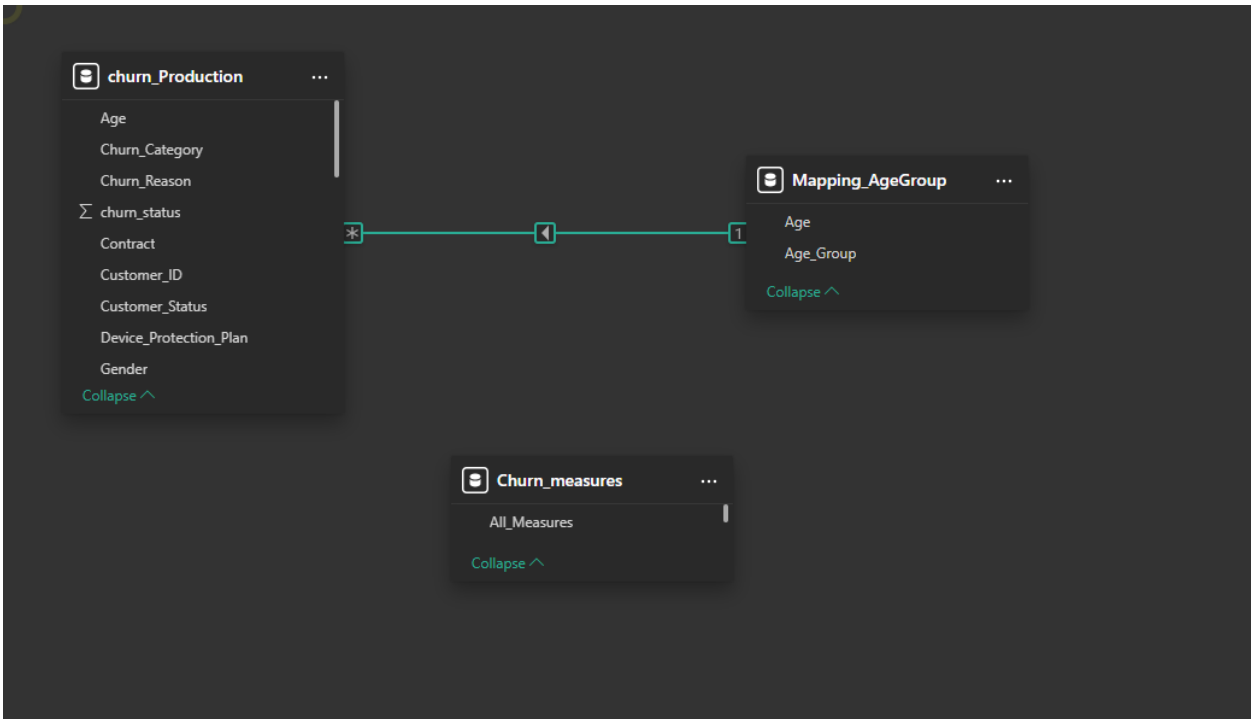
	1.2 Age	Age_Group
1	23	20-35
2	46	35-50
3	69	>50
4	29	20-35
5	75	>50
6	52	>50
7	72	>50
8	66	>50
9	78	>50
10	32	20-35
11	26	20-35
12	35	35-50
13	63	>50
14	43	35-50
15	55	>50
16	49	35-50
17	67	>50
18	27	20-35
19	21	20-35
20	58	>50
21	81	>50
22	64	>50
23	38	35-50
24	44	35-50
25	50	>50
26	24	20-35
27	47	35-50
28	70	>50
29	30	20-35
30	18	<20
31	84	>50
32	61	>50
33	41	35-50
34	65	>50
35	79	>50
36	19	<20

I created a new column as well with the formulae and I removed every other column beside Age

Age Group = if [Age] < 20 then “< 20” else if [Age] < 36 then “20 – 35” else if [Age] < 51 then “36 – 50” else “> 50”

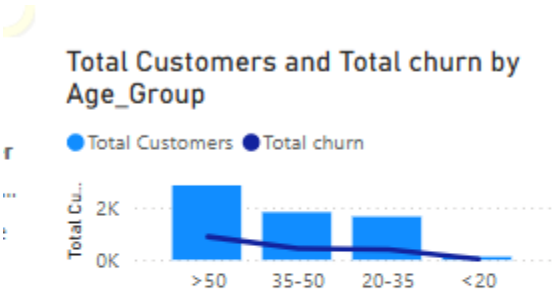
AgeGrpSorting = if [Age Group] = “< 20” then 1 else if [Age Group] = “20 – 35” then 2 else if [Age Group] = “36 – 50” then 3 else 4

Then I changed Change data type of AgeGrpSorting to Numbers



Now when I checked the model view section, I saw that my Mapping_AgeGroup is already linked to the churn_Production table.

This shows that I can easily use this Age inside of it to filter all columns on the churn_Production table



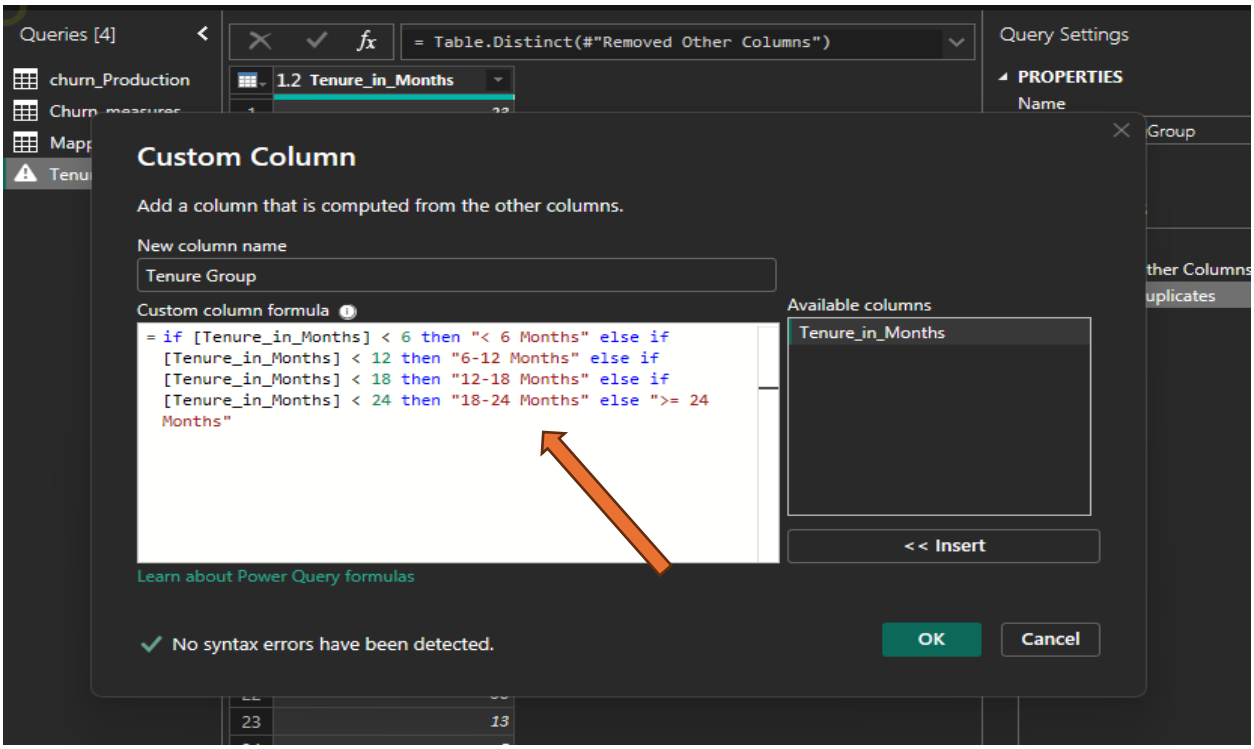
This is a line and stacked chart and my reason for using total customers and churn rate is because I would love to compare my churn rate with the number of customers that are available.

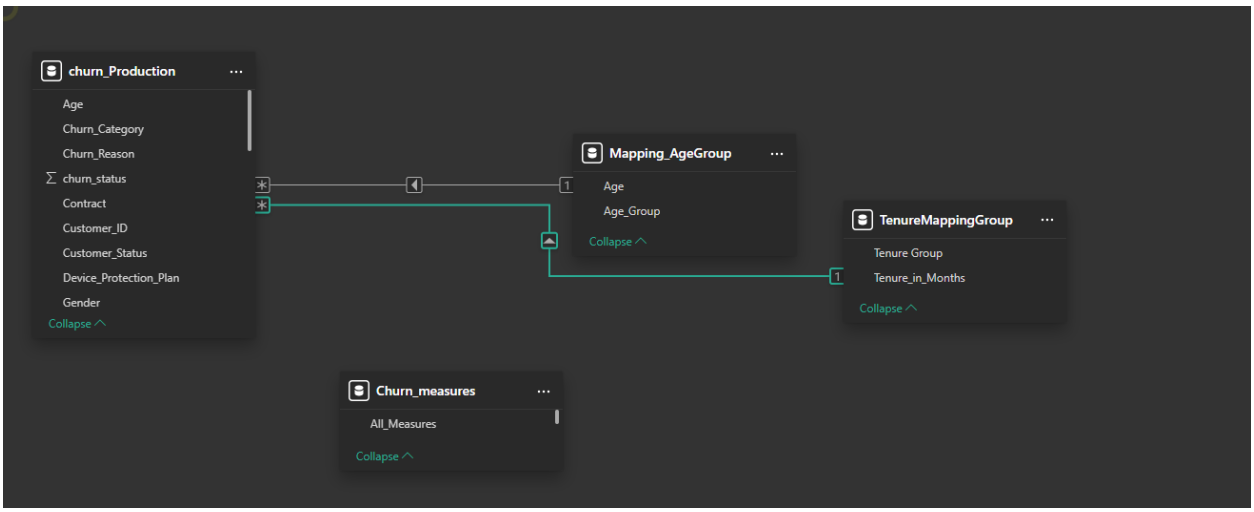
Now I am done with Demographics I will start with the Account information

I will be using the Clustered Bar chart for the Payment method By Total churn

I will be using the Clustered Bar chart for the Contract by Churn Rate

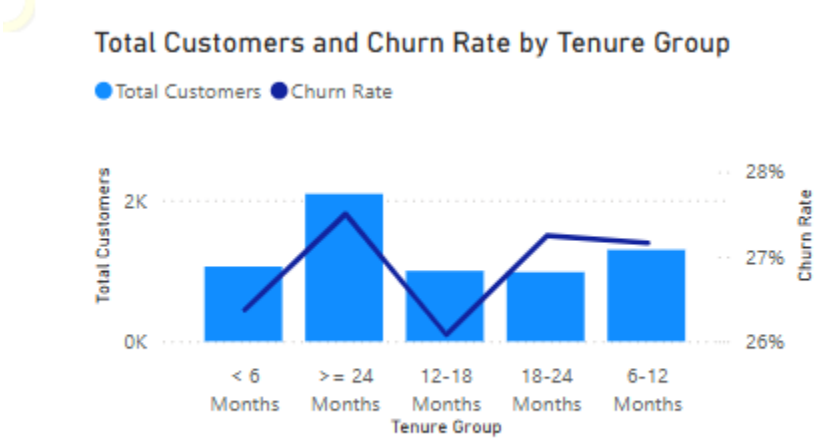
For Tenure I will have to Transform tenure since it has so many values which might affect my visualization, I will do the same as I did for Age.





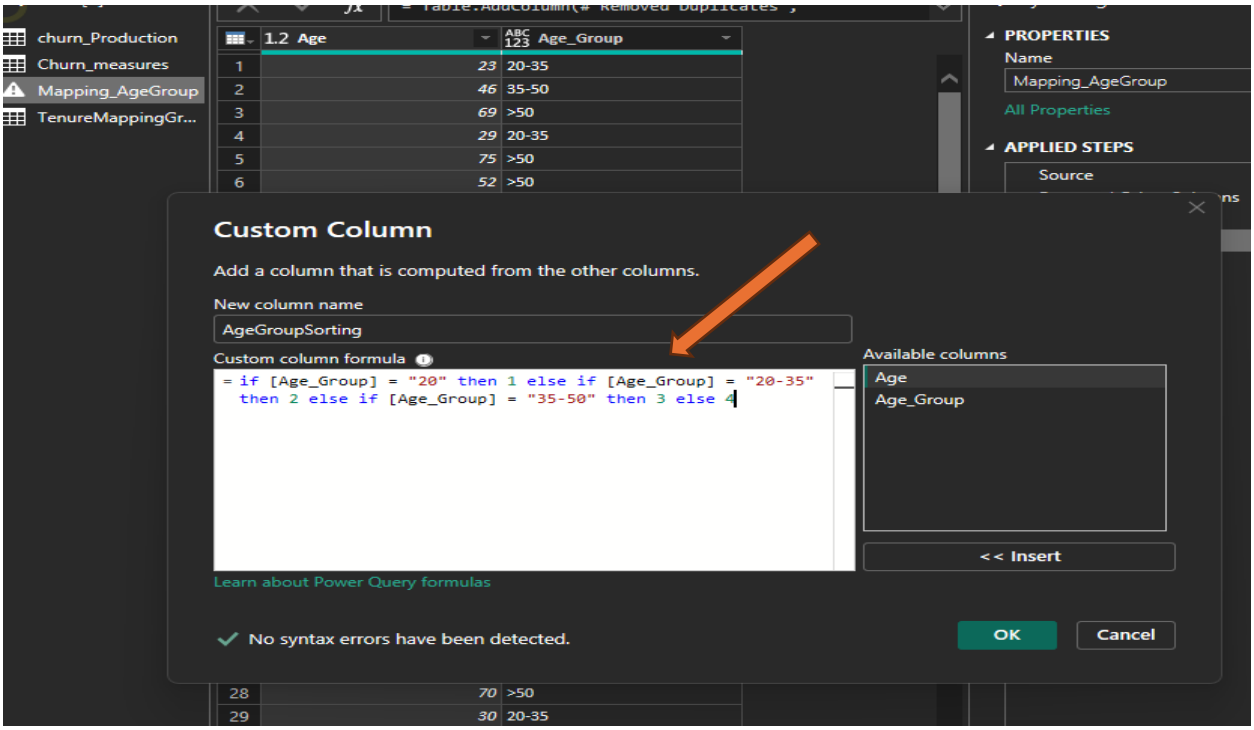
Power BI has already done the connection on the model view

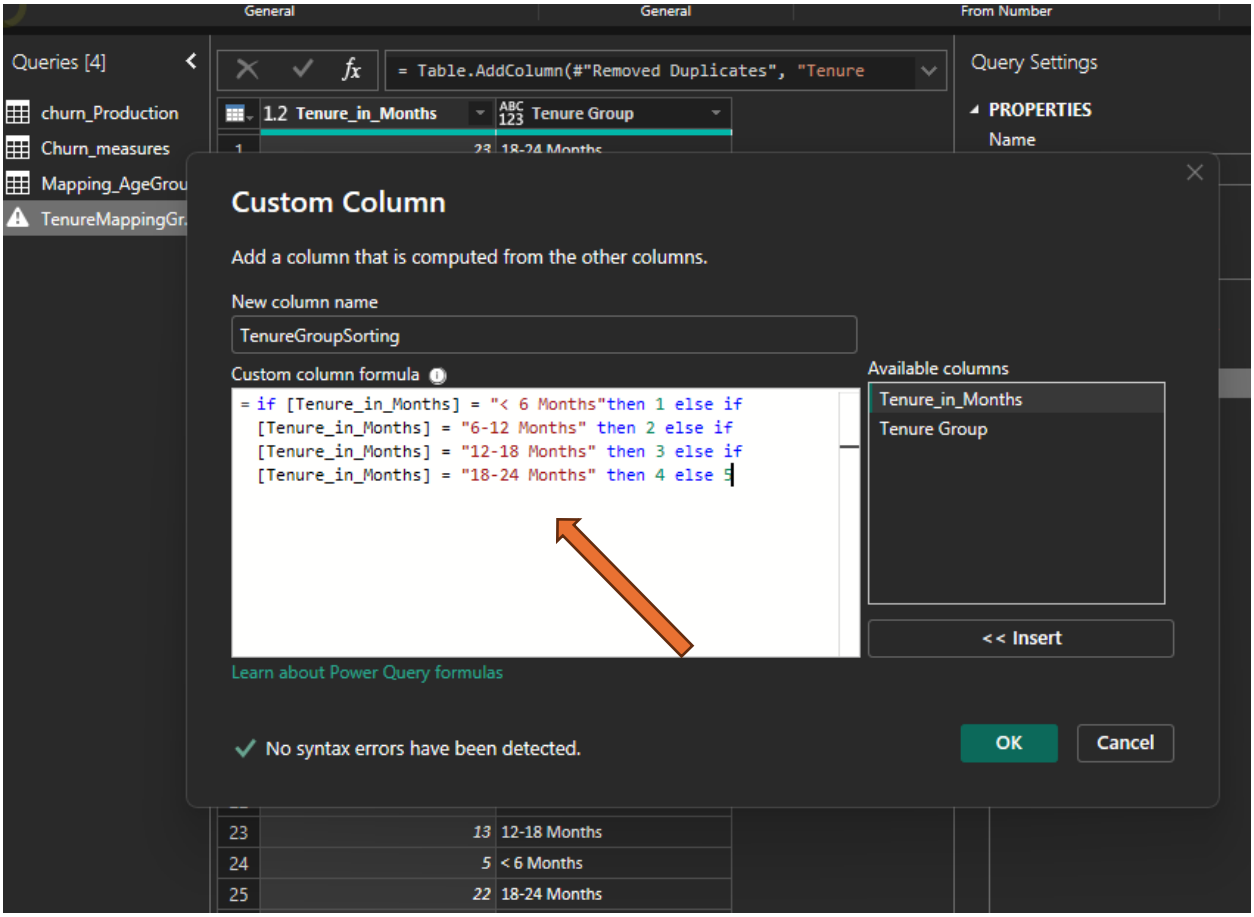
I will be using the Line & stacked Bar Chart for the Total Customers and Churn Rate by Tenure Group

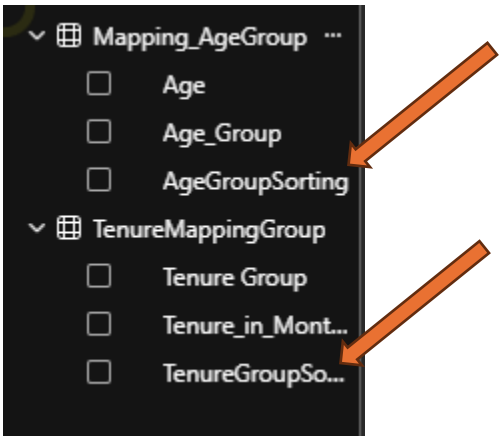


I dragged the churn rate from the x axis to the y axis to create that line

Looking at the x axis, it is not sorted in order so I will have to go back to transform and sort both the Mapping_AgeGroup and TenureMappingGroup

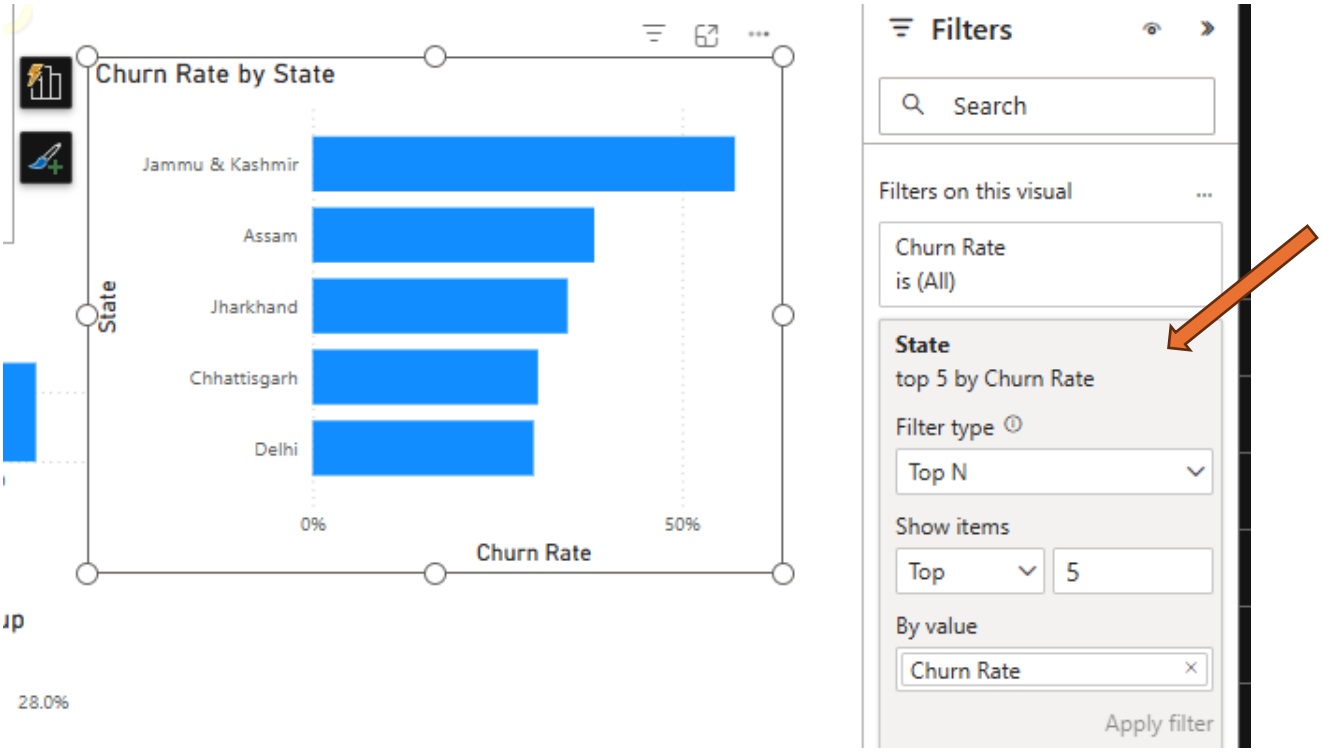






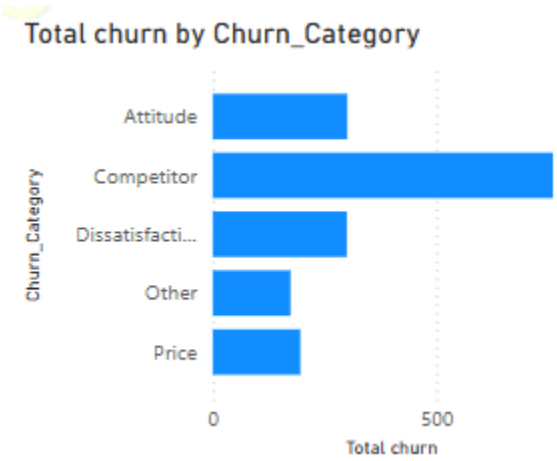
Is now showing on the Data pane

I will do the churn rate by state using the clustered bar chart, since there are a lot of states, I need the top 5 states with the highest churn rate.



I filtered it with the churn rate, which I dragged churn rate to the value box and from Basic filtering to Top N

- Now, the next step is to visualize **Churn Distribution** using the **churn_category** column with a **Clustered Bar Chart**.
- The reason I'm focusing on **Total Churn** instead of **Churn Rate** is because I'm analyzing the **churn_category,** and it's essential to include all categories in the analysis to get a complete picture.



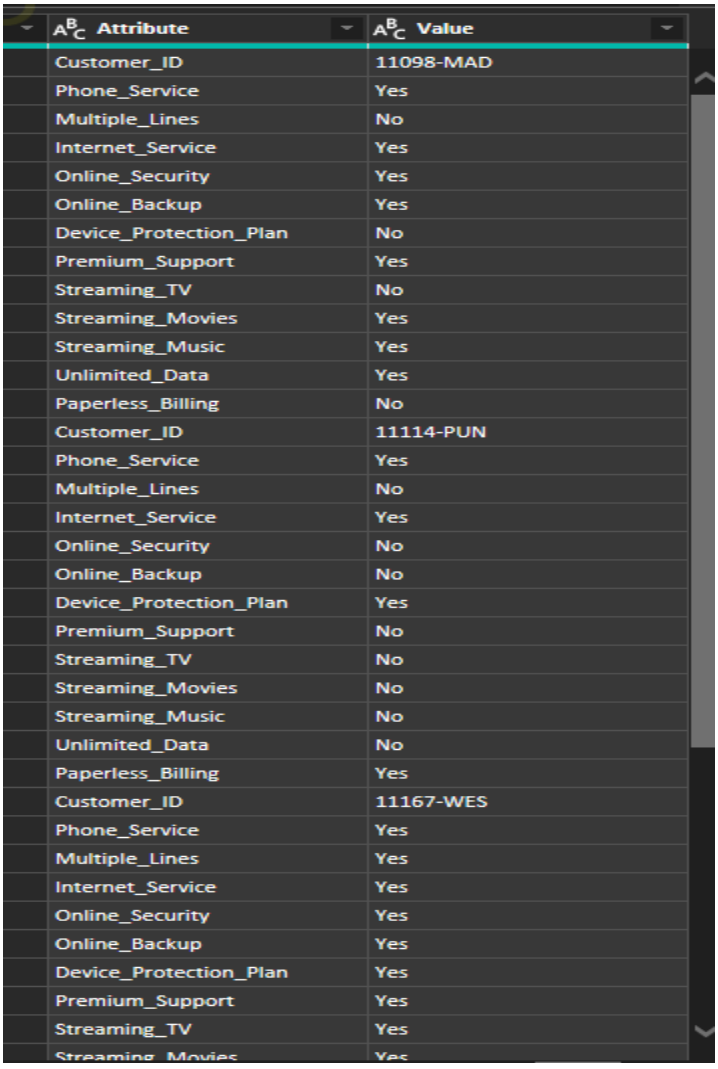
Now, I will perform the analysis for the **Internet_Type** column and visualize it using a **Clustered Bar Chart.** This time, I will include the **Churn Rate** to better understand how churn varies across different types of internet services.

Lastly, I want to create a visual that consolidates all the **telecom services** into a single view. This will provide a comprehensive overview of service usage or churn patterns across multiple services, making it easier to identify trends and insights immediately.

Phone_Service	Multiple_Lines	Internet_Service	Internet_Type	Online_Security	Online_Backup	Device_Protection_Plan	Premium_Support	Streaming_TV	Streaming_Movies	Streaming_Music	Unlimited_Data
No	No	Yes	DSL	No	Yes	Yes	No	No	No	No	Yes
No	No	Yes	DSL	Yes	No	Yes	No	No	No	No	Yes
No	No	Yes	DSL	No	Yes	No	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	Yes	No	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	Yes	Yes	Yes	No	No	No	Yes
No	No	Yes	DSL	No	Yes	No	No	No	No	No	Yes
No	No	Yes	DSL	Yes	No	No	Yes	No	Yes	Yes	Yes
No	No	Yes	DSL	Yes	No	Yes	No	No	No	No	Yes
No	No	Yes	DSL	No	No	No	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	No	No	Yes	No	No	No	Yes
No	No	Yes	DSL	No	No	No	No	No	No	No	Yes
No	No	Yes	DSL	Yes	No	Yes	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	No	Yes	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	Yes	No	Yes	No	Yes	Yes	Yes
No	No	Yes	DSL	No	No	No	No	Yes	Yes	Yes	Yes
No	No	Yes	DSL	No	Yes	No	No	No	Yes	Yes	Yes
No	No	Yes	DSL	Yes	Yes	No	No	No	No	No	Yes
No	No	Yes	DSL	Yes	Yes	No	No	No	No	Yes	Yes
No	No	Yes	DSL	Yes	Yes	Yes	Yes	No	No	No	Yes
No	No	Yes	DSL	No	Yes	No	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	No	No	Yes	No	Yes	Yes	Yes
No	No	Yes	DSL	No	Yes	No	No	No	No	No	Yes
No	No	Yes	DSL	No	No	No	No	No	No	No	Yes
No	No	Yes	DSL	No	Yes	Yes	No	No	Yes	Yes	Yes
No	No	Yes	DSL	No	No	No	No	No	No	No	Yes
No	No	Yes	DSL	Yes	No	Yes	Yes	No	No	No	Yes
No	No	Yes	DSL	No	Yes	No	Yes	No	No	No	Yes
No	No	Yes	DSL	Yes	Yes	No	No	No	No	No	Yes
No	No	Yes	DSL	No	No	No	No	No	No	No	Yes
No	No	Yes	DSL	No	Yes	Yes	No	No	Yes	Yes	Yes
No	No	Yes	DSL	No	No	No	No	No	No	No	Yes
No	No	Yes	DSL	No	Yes	Yes	No	Yes	No	No	Yes

Before I start, I need to rearrange all the telecom service columns into a grid format so that I can easily visualize and compare the services in one chart.

I clicked on **Transform Data,** which took me to the **churn_production** table. I created a **reference** for it and renamed it to **services_production.** Then, I used **Ctrl** to select all the columns related to services, went to the **Transform** tab above, and clicked on **Unpivot Columns** to rearrange the data for easier visualization.

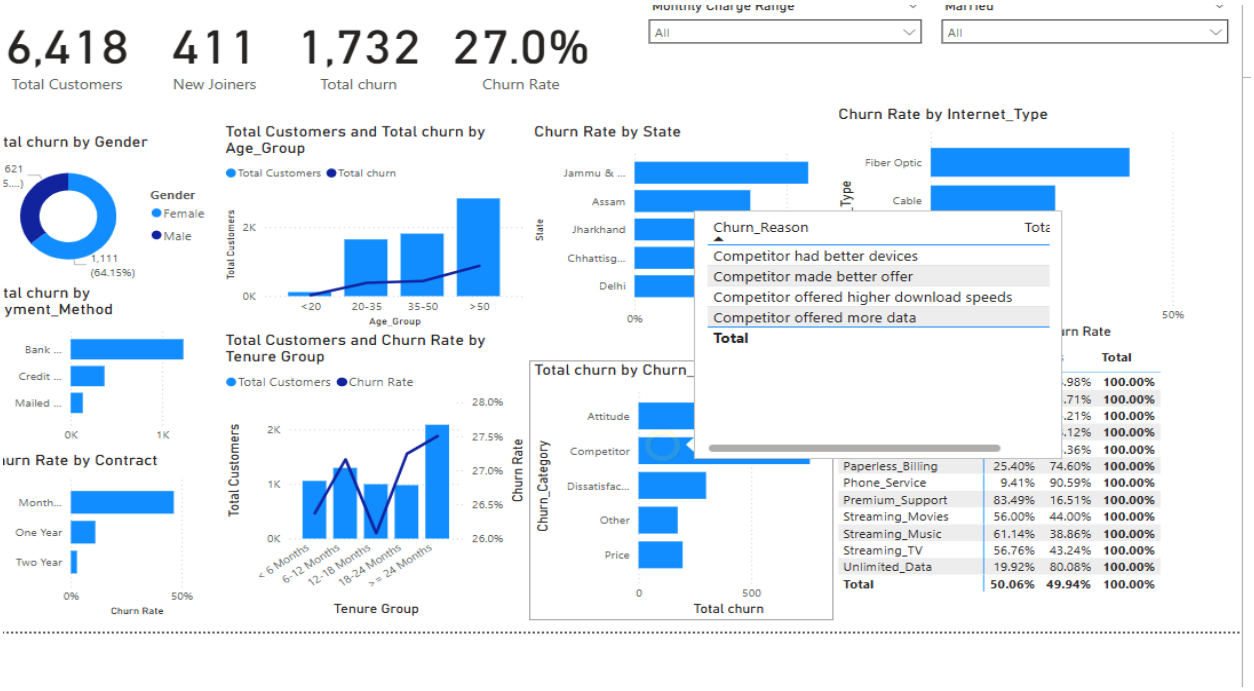


Attribute	Value
Customer_ID	11098-MAD
Phone_Service	Yes
Multiple_Lines	No
Internet_Service	Yes
Online_Security	Yes
Online_Backup	Yes
Device_Protection_Plan	No
Premium_Support	Yes
Streaming_TV	No
Streaming_Movies	Yes
Streaming_Music	Yes
Unlimited_Data	Yes
Paperless_Billing	No
Customer_ID	11114-PUN
Phone_Service	Yes
Multiple_Lines	No
Internet_Service	Yes
Online_Security	No
Online_Backup	No
Device_Protection_Plan	Yes
Premium_Support	No
Streaming_TV	No
Streaming_Movies	No
Streaming_Music	No
Unlimited_Data	No
Paperless_Billing	Yes
Customer_ID	11167-WES
Phone_Service	Yes
Multiple_Lines	Yes
Internet_Service	Yes
Online_Security	Yes
Online_Backup	Yes
Device_Protection_Plan	Yes
Premium_Support	Yes
Streaming_TV	Yes
Streaming_Movies	Yes

Now it added everything into 2 columns with the Attributes and Values which I changed the name to services and status

Now, I will click on **Service**, which will appear in the Fields pane. Then, I'll select the **Matrix** visual from the Visualizations pane. In the **Build** pane, I will drag **Status** to the **Columns** section and **Services** to the **Rows** section to display the data in a clear and organized grid format.

I will drag **Churn Status** to the **Values** section in the Build pane. By default, it shows the **sum of Churn Status,** but that's not what I want. I need to see the **percentage of each service** in terms of **Churn Status (Yes/No).** So, I will click the dropdown on the field in the Values section, go to **Show Values As,** and select **Percent of Row Total** to display the data as percentages.

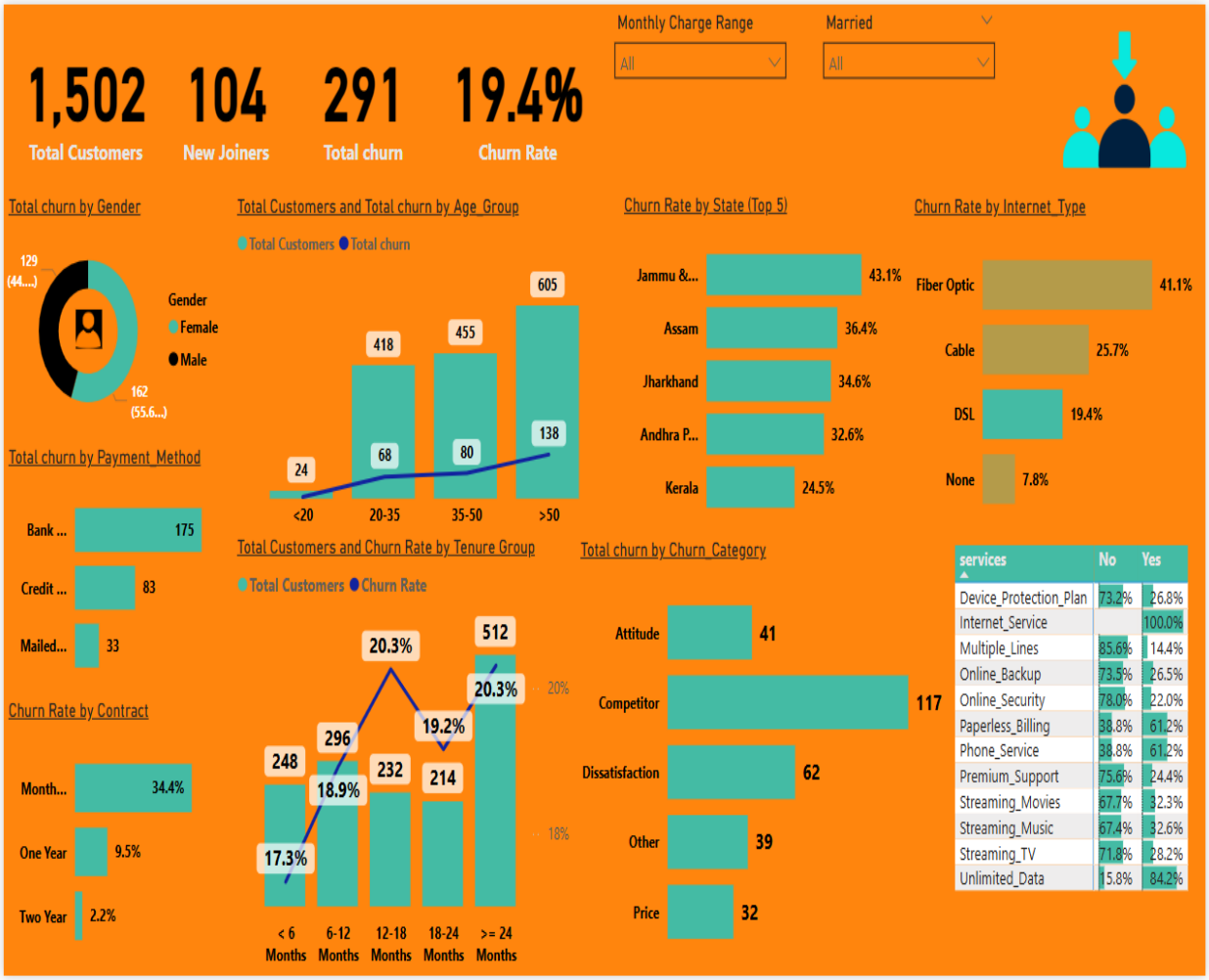


I analyzed churn by incorporating Monthly Charge Range, Marital Status, and Service Usage to understand churn patterns. I calculated churn rates for each service and visualized the distribution of churn reasons as a percentage of total churn. This helps identify which customer groups and services are most associated with churn and why customers are leaving.

//

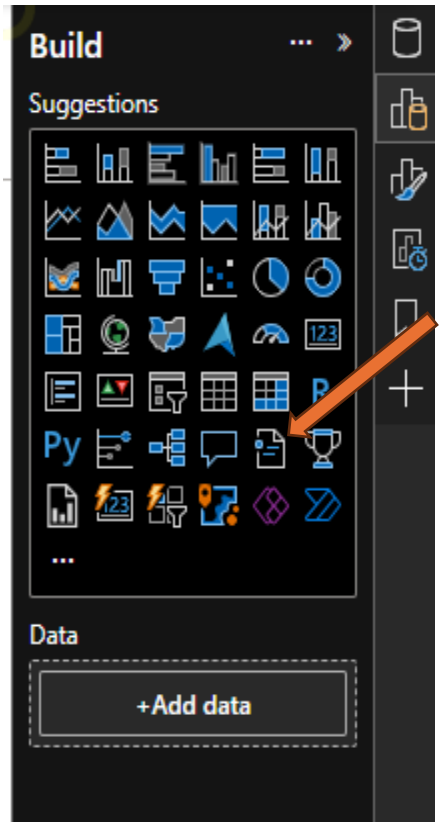
//

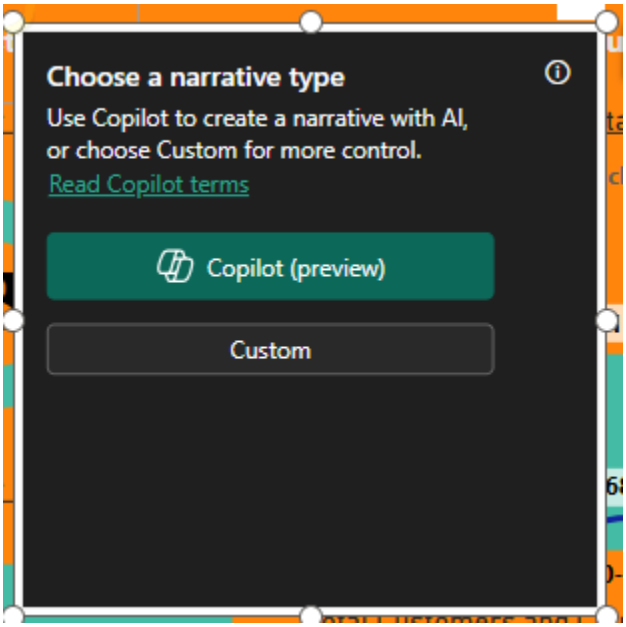
3.3 Enhance Visualization



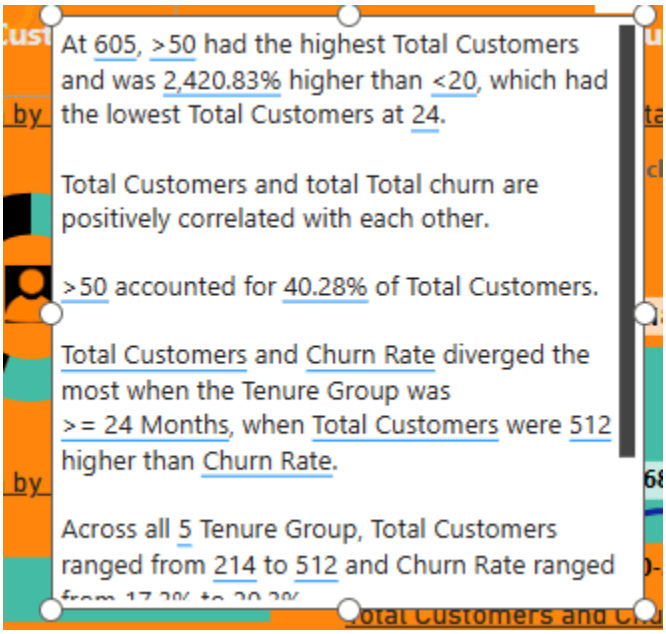
This interactive Power BI dashboard provides a comprehensive analysis of customer churn using demographic, service usage, and contract data. Key business insights are highlighted to help reduce churn and improve customer retention.

Now I will be using the AI Narrative tool on the Visualization Pane

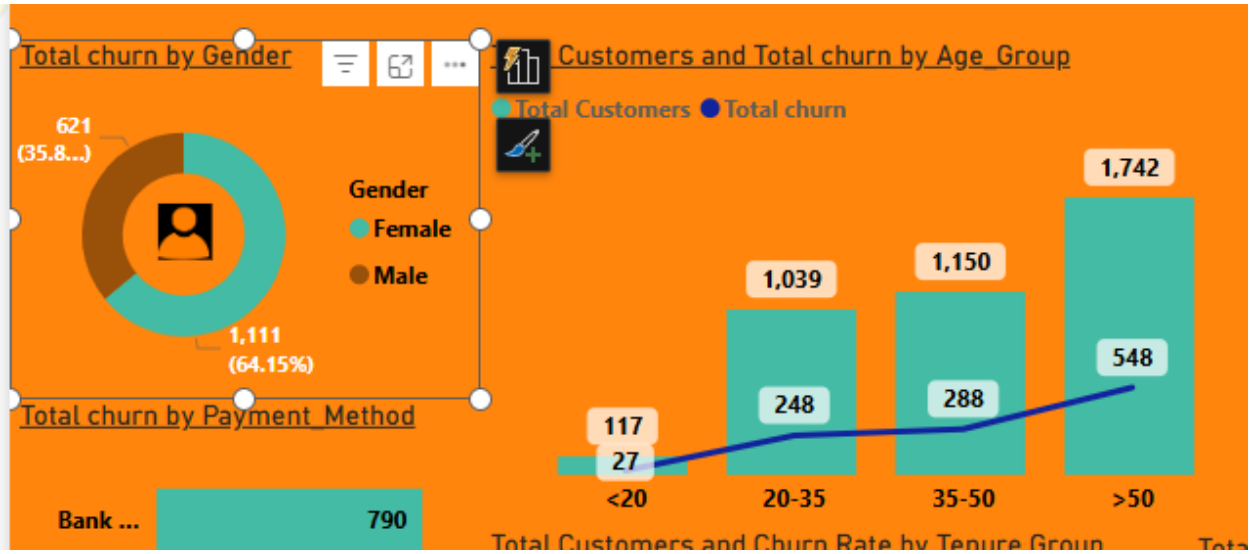




When I click on custom it will give me a summary of my entire visual in writing



If I filter the visuals on my dashboard this AI copilot gives me a brief explanation of it and what happened, and this is just a starting point for me to start my analysis of the Dashboard



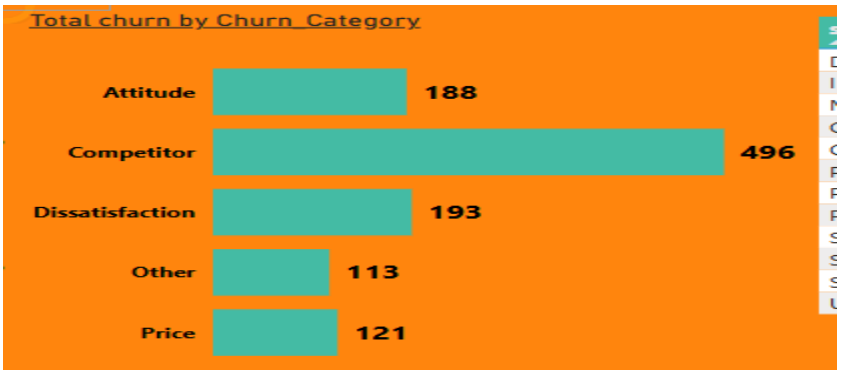
The **"Total Churn by Gender"** chart shows that **female customers** **account for 621 churned users,** representing **35.8%** of total churn. While male churn is higher overall, a deeper look into the **"Customers and Total Churn by Age Group"** chart reveals a strong issue:

- In the **50+ age group**, **548 customers churned,** which is the highest among all age brackets.
- Given the gender breakdown, it's clear that a significant portion of these churned users are **older female members.**

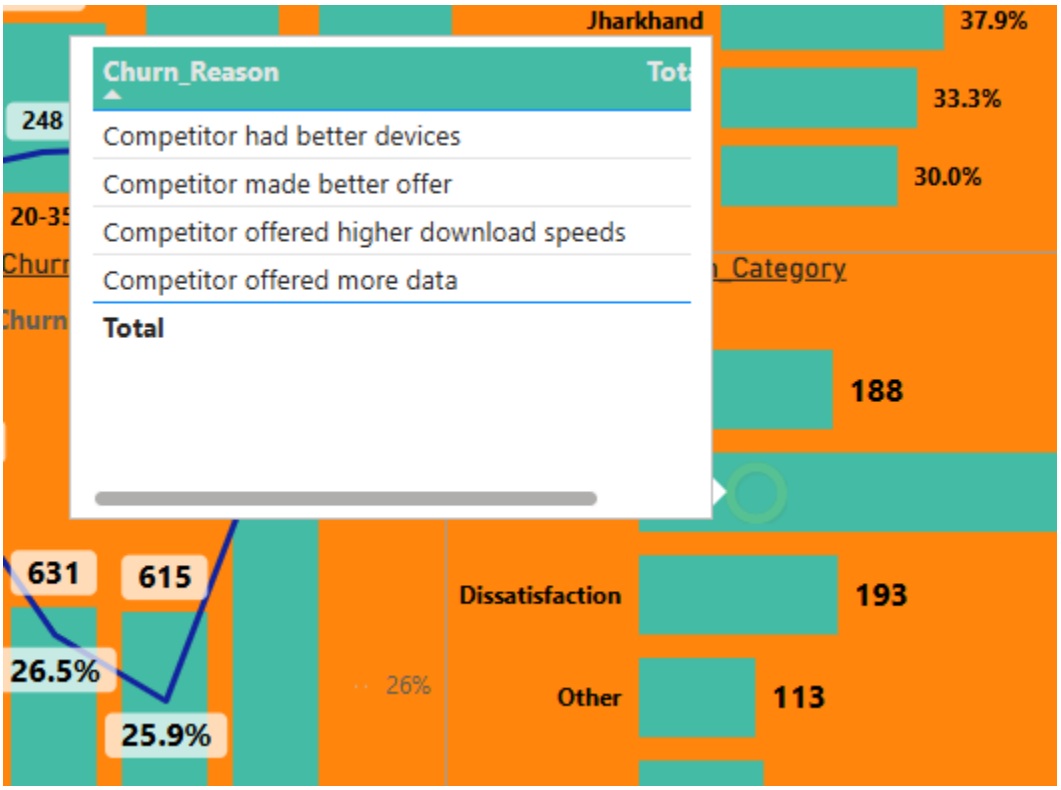
3.4 Marketing Solution

Goal

If I design a marketing campaign specifically targeting females aged over 50, I could help reduce churn significantly in this high-risk group. This segment accounts for a large portion of the 548 churned customers in the 50+ age group, as shown in the dashboard. By addressing their unique needs and offering personalized support, I can retain more loyal customers and improve overall retention.

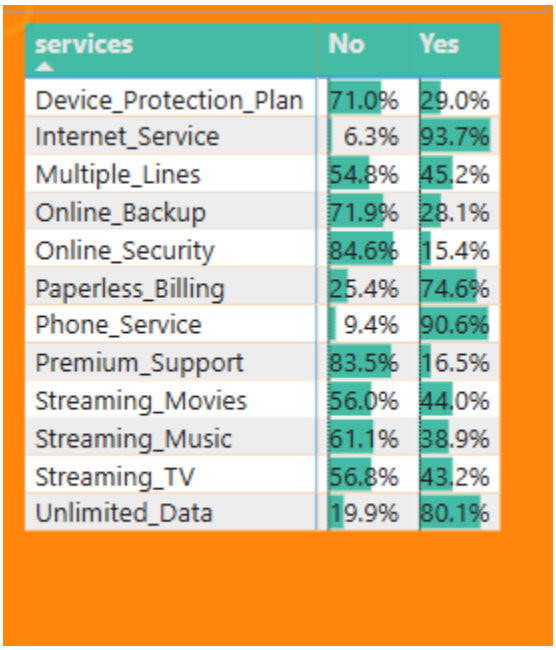


As I analyze further, I can see that most females that was churned was due to competitors



When I hovered over the **churn category data**, I found that many customers left due to **better devices and offers from competitors.** These two factors fall under the **"Competitor" churn reason**, which had the highest count.

Now looking closely at the Grid which is Churn by Services



services	No	Yes
Device_Protection_Plan	71.0%	29.0%
Internet_Service	6.3%	93.7%
Multiple_Lines	54.8%	45.2%
Online_Backup	71.9%	28.1%
Online_Security	84.6%	15.4%
Paperless_Billing	25.4%	74.6%
Phone_Service	9.4%	90.6%
Premium_Support	83.5%	16.5%
Streaming_Movies	56.0%	44.0%
Streaming_Music	61.1%	38.9%
Streaming_TV	56.8%	43.2%
Unlimited_Data	19.9%	80.1%

Using a personal benchmark of **60% non-subscription** as an indicator of poor service uptake, I analyzed the grid and found that several key services are underutilized by customers — particularly **females aged over 50,** who are already a high-churn demographic.

🚩 **Services with low adoption (over 60% saying "No"):**

- **Device Protection Plan** – 71%
- **Online Backup** – 71.9%
- **Online Security** – 84.6%
- **Premium Support** – 83.5%
- **Streaming Music** – 61.1%

CONCLUSION

High-Risk Demographics Identified

- I Discovered that **35.8% of churners are female**, with **50+ age group contributing 548 churned customers** which is a critical segment to prioritize.
- There were High Competitors with better offers and devices which dominated and exposed vulnerabilities in retention strategies.

Gap in Services Are Exposed

- This is the Alarming low adoption of services used my customers whereby:
- 84.6% lack Online Security
- 83.5% don't use Premium Support
- 71% skip Device Protection
- Looking at the gaps in % of services not used by customers, you will see the dissatisfaction and competitors switching.

Payment & Contract Weaknesses

- Month-to-month contracts have 3× higher churn risks than annual plans.

- Bank transfer users churn 22% more frequently than automatic payment users.

MY RECOMMENDATION

"Silver Shield" Loyalty Bundle for Women 50+

- With Total Package Deal of (Device Protection + Premium Support + Streaming Music) at a 40% discounted rate for this demographic.
- After 12 months of subscription, customers will be given Free device upgrade which is a strong counter to competitors offers.

Lock-In & Save Campaign Discount

- annual contracts: "Pay for 10 months, get 12" which is a 20% effective discount to reduce month-to-month volatility.

IMPACT I ENABLED

- With these Dashboard-Driven Decisions, Executives can now track real-time churn drivers like "Competitor Offers" (23% of churn) vs "Service Dissatisfaction" (18%).
- **Service Revenue Growth:** Closing adoption gaps could generate **$1.2M/year** from previously unused services.

THE FUTURE IS ABOUT STAYING AHEAD

This project shows that losing customers isn't unavoidable, we can fix it. By using smart data insights, personalized deals, and teaching customers about underused services, I've built a clear plan to turn customers who might leave into loyal fans. The dashboard I created isn't just for reports, it's a powerful tool to spot risks early and act fast.

My next Plan

Is to Test my ideas: Run two versions of campaigns (A/B tests) to see which ones work best. Track the "Total Churn" number and watch it shrink every month.