

# **THE NEOTIA UNIVERSITY**

## **A Project Report ON *SALES INSIGHT USING POWER BI***

Submitted To,  
The Neotia University

Submitted By,  
Kuntal Chaudhury,CSE(DA),02  
Subhankar Mandal,CSE(DA),01  
Ashish kr. Singh,CSE(DA),03  
Nikita Jha,CSE(DA),06

Project Guide,  
Prof. Jaydeb Mandal

# INDEX

- Acknowledgement
- Certificate
- Introduction
- Methodology
- Problem Statement
- Steps
- To Analyze the data using power BI
- Sales Insights Using Power BI
- Int Production
  - \*Purpose
  - \*Scope
  - \*Abbreviations
  - \*References
  - \*Overall Description

## \*Other requirements

- Microsoft Excel
- ETL Tool as Talend
- Power BI
- Analysis & Report of the Present Manual System
- Testing & Implimation

## **ACKNOWLEDGEMENT**

We gratefully acknowledge for the assistance, cooperation, guidance and clarifications provided by **THE NEOTIA UNIVERSITY** during the development of the **Sales Insight Using Power BI** . Our extreme gratitude to **Mr.Jaydeb Mandal** who guided us throughout the project. Without his willing disposition, spirit of accommodation, frankness, timely clarification and above all faith in us, this project could not have been completed in due time.

His readiness to discuss all important matters at work deserves special attention.

We would also like to thank whole of the faculty of the college for their cooperation and important support.

## **CERTIFICATE**

This is to certify that the project entitled “**Sales Insight Using Power BI**”has been developed by “**KUNTAL CHAUDHURY,SUBHANKAR MANDAL, ASISH KR. SINGH , NIKITA JHA ”**.

He has worked on the project Online Exam System. He has used MICROSOFT EXCEL , TALEND , POWER BI for the project. His work is satisfactory.

I wish him all the best for his bright future

**Dated:-19/DECEMBER/2022**

**Project Incharge**  
**Prof.Jaydeb Mandal**

## INTRODUCTION

Sales Data Analysis provides an understanding of the product that your customers are buying and help you dissect why they are behaving in a certain way. You can analyze this behaviour and lead the further process. Many global, industry-leading brand are now using their sales data in inventive ways to make better business decision, but any company can take advantages of insights and reporting tools to achieve data-driven sales success. This project is the outcome of a descriptive research on past, present, and future of sales industry and the application of business analytics in shaping appropriate marketing strategies with data sources, DAX query language through dashboard in Power BI. The project aim to show on how we can use the Power BI with sales analysis data using SQL query for data cleaning and DAX query language and its performance on presenting the dashboard to the end users. So in this project, I have created dashboard to know the trends and business performance and also to know which product sales most, which product sales most in particular regions and market performance. The project represents the large dataset into visualization form that help to take business decision.

Keywords: Sales Analysis, research, DAX query ,POWER BI, SQL ,dashboard, Data visualization, report generate.

Power BI is the technical and procedural representation of data. It is an infrastructure that collects, stores and analyzes the data produced by a company's activity. Power BI parses all the data generated by a business and presents easy-to-digest reports, performance measures and trends that inform management decisions. BI components and software come in a wide variety of Power query, Power map, Power pivot, Power view, Power Q&A, Power BI desktop. There are many others parts for Power BI as well such as Power BI.com Websites, Power BI Mobile Apps. Power BI is cloud-based data analysis, which can be used for reporting and data analysis from a wide range of data sources. Power BI is simple and user friendly enough that business analysts and power users can work with it and get benefits of it. On other hand Power BI is powerful and mature enough that can be used in enterprise systems by BI developers for complex data mashup and modelling scenarios.

## **METHODOLOGY**

- **Data Collection–**

Data Collection is the process of gathering and measuring information on targeted variables in an established systems, which then enables one to answer relevant questions and evaluate outcomes.

- **Data Storage –**

Power BI uses two primary repositories for storing and managing data: Data that is uploaded from users is sent to excel sheet.

- **Data Cleaning -**

Once the data is stored into power BI it is important to clean and transform data before you build any visualizations or reporting .This is essential step in building quality visualization. Cleaning and transforming data enables to build visualizations from sound and clean data. If it is not done diligently visualization will not behave as we expect.

- **Data analysis –**

Data analysis is a process of inspecting, cleaning, transforming and modelling data with the goal of discovering useful information, informing conclusions and supporting decision making .

We use DAX (Data Analysis Expressions) for Data Manipulation and working on the data.

- **Visualization –**

In this process we convert manually data into visualization is to visually display collected data by using various charts, graphs or other visualization type

## **PROBLEM STATEMENT**

A hardware is a company in India which supplies computer hardware and peripheral devices across India only. The have many stores across India such as surge stores, Nomad stores etc. The head office of the company is situated in Delhi.

- **Scenario —**

The sales manager of the company is facing many challenges. He is facing issues in tracking sales in dynamically growing market. He is having issues with the insights of his business.

In order to this he has some of the regional managers in North, south and central India working for the company. So, he calls them and ask about the insights he wants to know. They tell him about the sales in last quarter and the growth in that quarter.

So, the problem is that the conversations that are happening are verbal. Hence, the regional managers are sugar coating the facts and the manager of the company does not get the clear picture of the facts. Even after knowing that the sales are declining, he cannot do anything because he does not have the clear picture of the sales. Asking for the records the regional manager provides him with excel files. But by this he cannot figure out small things.

All what the manager wants is a view of the weakest area the company need to focus to increase the sales and improvise the



declination. He is interested in simple, understandable and digestive insight. So, he is more interested in a dashboard which he can go and look at the real data because data speaks the truth. All he wants is a simple data visualization tool which he can access on daily basis.

Hence, by using such tools and technology one can make data driven decisions which helps to increase the sales of the company.

So, in this project we will help a company make its own sales related dashboard using PowerBI.

### **Data Discovery**

#### **Project planning using AIMS grid –**

AIMS grid: It is a project management tool which consists of four components to it.

- 1) Purpose (what to do exactly)
- 2) Stack holders (who will be involved)
- 3) End result (what do you want to achieve)
- 4) Success criteria (cost optimization and time save)

In our case the end result will be the dashboard created and success criteria will be bumping up the sales using cost optimization and save the time of the manager of the company.

## AIMS GRID

<p><b>PURPOSE</b></p> <p>To unlock sales insights that are not visible before for the sales team for decision support &amp; automate them to reduced manual time spent in data gathering.</p>	<p><b>STAKEHOLDERS</b></p> <ul style="list-style-type: none"> <li>• Sales Director</li> <li>• Marketing Team</li> <li>• Customer Service Team</li> <li>• Data &amp; Analytics Team</li> <li>• IT</li> </ul>
<p><b>END RESULT</b></p> <p>An automated dashboard providing quick &amp; latest sights in order to support data driven decision making.</p>	<p><b>SUCCESS CRITERIA</b></p> <ul style="list-style-type: none"> <li>• Dashboard uncovering sales order insights with latest data available.</li> <li>• Sales team able to take better decisions &amp; prove 10% cost saving of total spend</li> <li>• Sales Analysts stop data gathering manually in order to save 20% of business time and reinvest it value added activity.</li> </ul>

## Flowchart of the project execution –

- Data collection @talend -> Data Transformation( XML to csv file/ JSON to csv file)



## How will the company work Using Power BI –

*There is a team of software engineers (falcons) which owns sale*

*management system. The records of this system are stored in Power BI database.*

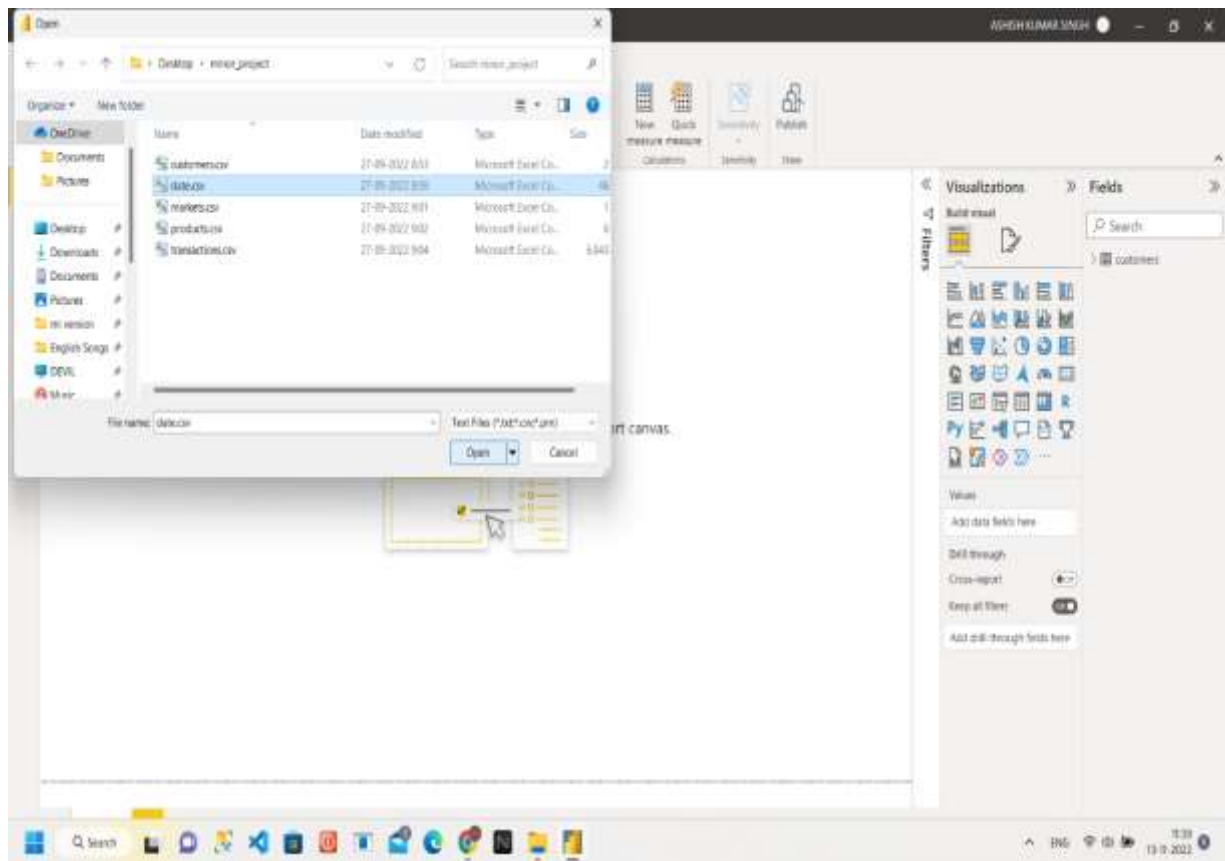
*The team of Data Analyst (Data masters) reaches out to the software engineers to get an access to data base which they can use to create the dashboard in PowerBI.*

In this same manner our project is going to be executed. We are going to fetch the data from the database from company's website and then we are going to transform and load the data in the PowerBI to build the dashboard.

### **Step 1: Collecting All The Company Sales Files.**

All the files are collecting from the google drive or company files and save into the desktop or dwonloads . All files are save into XML , JSON , csv File.

So, All the files are convert into csv. File to analyze the data. To find the sales insights the company data for growth of the company we get all the files into csv.file .All the files are import to power BI in csv format because all the files are analyze in Power BI in very easily that's why we need all the file in csv format.



## Step 2: . Importing Data to Talend workbench For converting any file to csv file.

All the collecting files are change into csv. Format for analyze the data into Power BI . To find the sales insights the company data for growth of the company we get all the files into csv.file .All the files are import to power BI in csv format because all the files are analyze in Power BI in very easily that's why we need all the file in csv format.

Talend is an ETL tool for Data Integration. It provides software solutions for data preparation, data quality, data integration, application integration, data management and big data. Talend has a separate product for all these solutions. Data integration and big data products are widely used. This tutorial helps you to learn all the fundamentals of Talend tool for data integration and big data with examples.

## **Audience**

Talend is also ideal for Big Data professionals who are looking to use an ETL tool with Big Data ecosystem.

## **Prerequisites**

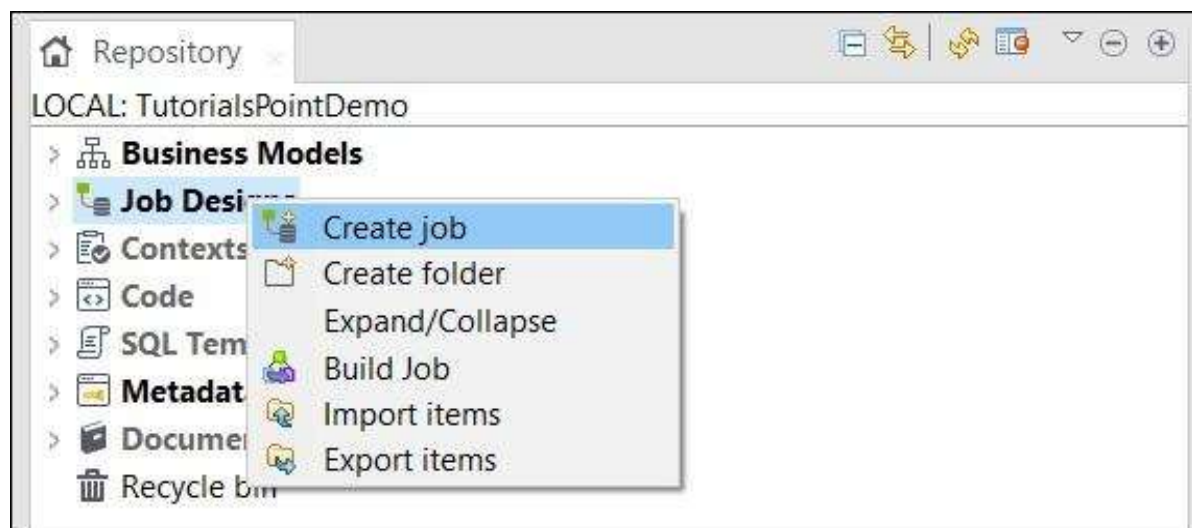
You should be familiar with basic Data warehousing concepts as well as fundamentals of ETL (Extract, Transform, Load).

The steps of the converting data into csv format to analyze the data-

This is the technical implementation/graphical representation of the business model. In this design, one or more components are connected with each other to run a data integration process. Thus, when you drag and drop components in the design pane and connect them with connectors, a job design converts everything to code and creates a complete runnable program which forms the data flow

## Creating a Job

In the repository window, right click the Job Design and click Create Job.



- Provide the name, purpose and description of the job and click Finish.

**New Job**  
Add a job in the repository

Name: NewJob

Purpose: Creating a new job example

Description: test example

Author: user@talend.com

Locker:

Version: 0.1

Status:

Path:

Finish Cancel

- You can see your job has been created under Job Design.

Talend Open Studio for Big Data (7.0.1.20180411\_1414) | TutorialsPointDemo (Connection: Local)

File Edit View Window Help

Repository: LOCAL: TutorialsPointDemo

- Business Models
  - Job Designs
    - t01\_Compo
    - t02\_Concatenating
    - t03\_DemoRoutines
    - t04\_Context
    - beforeRunJobs 0.1
    - NewJob 0.1**
    - priorTest 0.1
    - readme 0.1
  - Contexts
  - Code
  - SQL Templates
  - Metadata
  - Documentation
  - Recycle bin

Job: NewJob 0.1

Designer/Code

Job: NewJob 0.1

Main

Name: NewJob

Author: user@talend.com

Version: 0.1

Creation: 18/8/18 1:40 PM

Modification: 18/8/18 1:40 PM

Purpose: Creating a new job example

Status:

Description: test example

Find compone

Favorites

Recently Used

Big Data

Business Intelli...

Business

Cloud

Custom Code

Data Quality

Databases

Databases NoS...

DotNET

ELT

ESB

File

Internet

Logs & Errors

Misc

Orchestration

Processing

System

Talend MDM

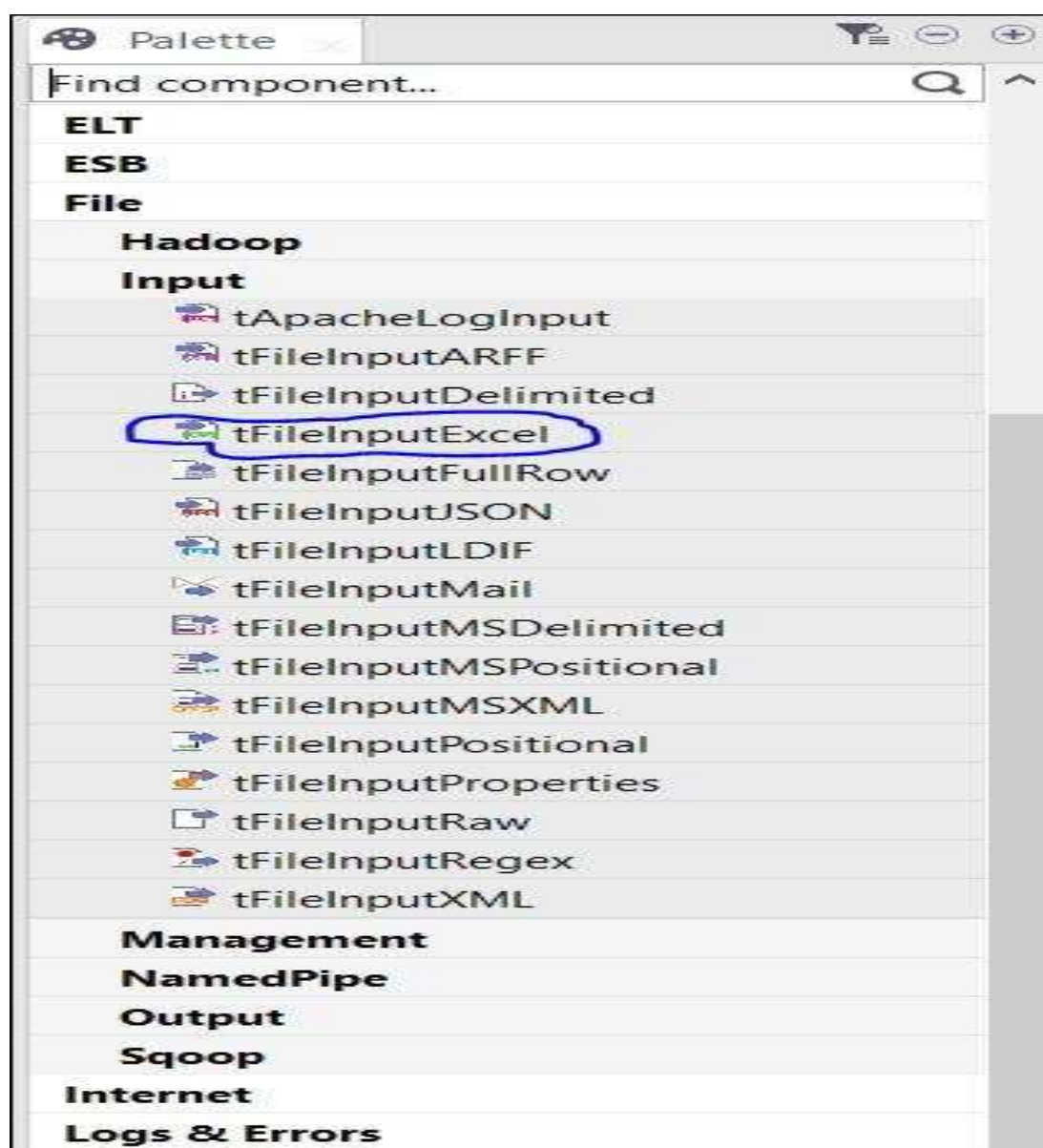
Unstructured

XML

Now, let us use this job to add components, connect and configure them. Here, we will take an excel file as an input and produce an excel file as an output with same data.

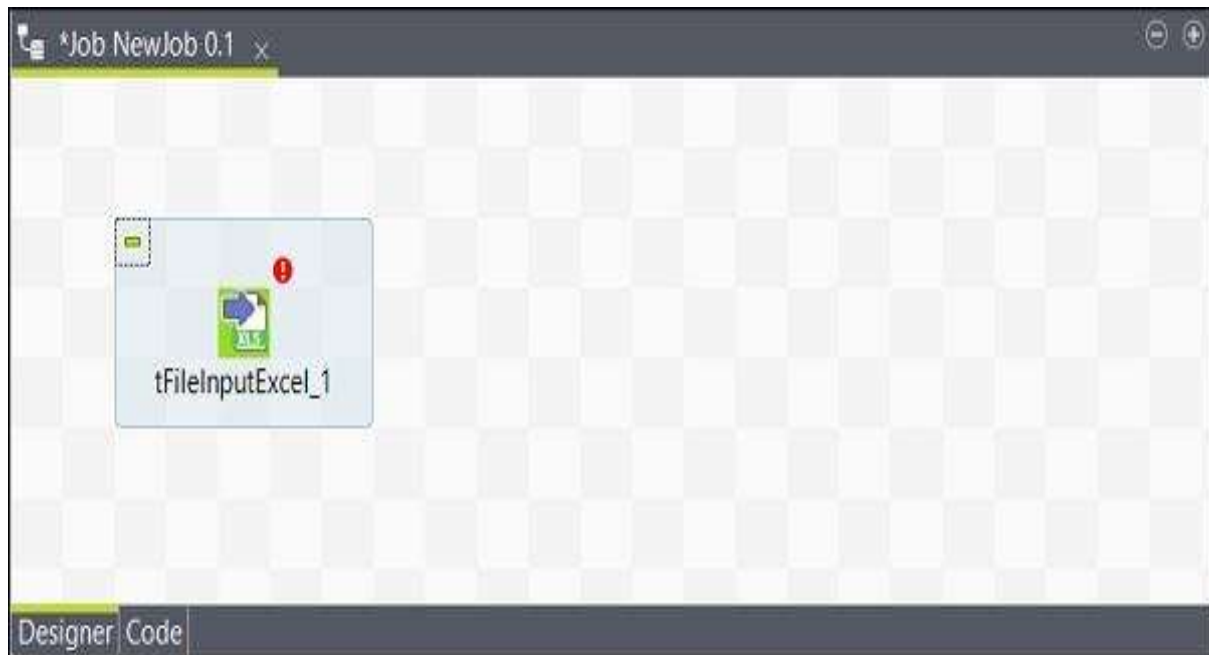
## Adding Components to a Job

There are several components in the palette to choose. There is a search option also, in which you can enter the name of the component to select it.

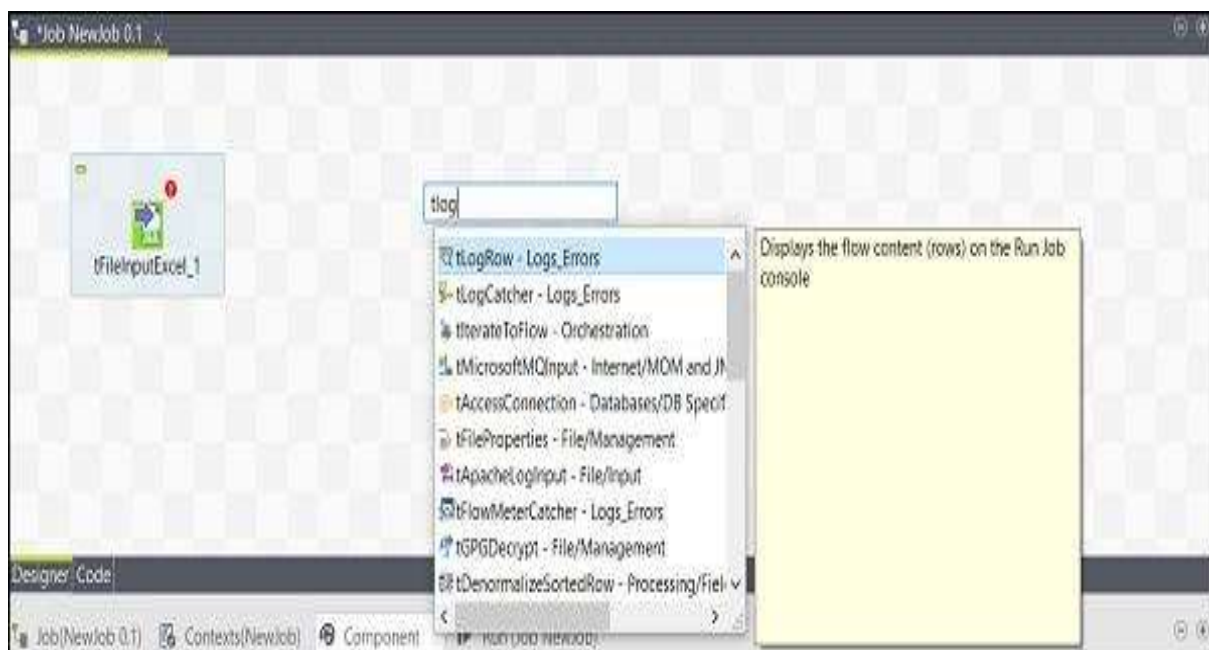




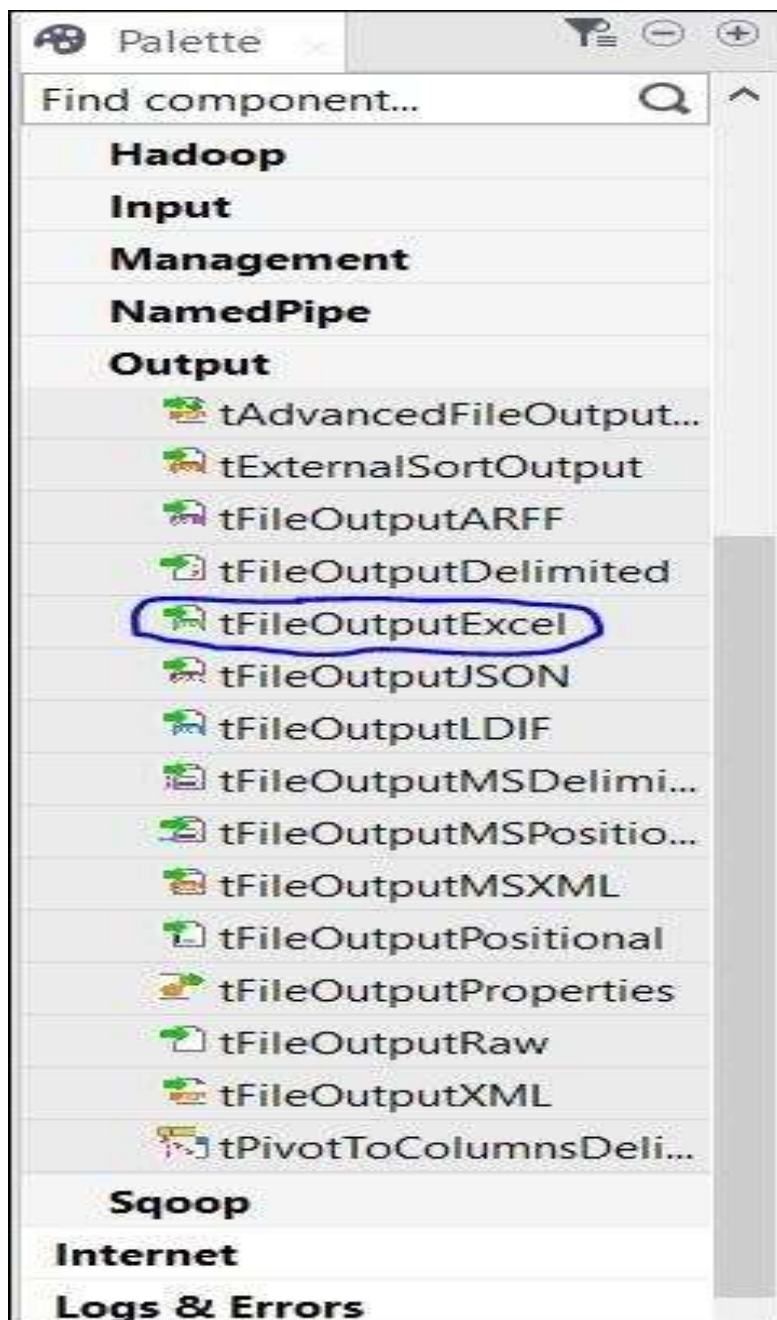
Since, here we are taking an excel file as an input, we will drag and drop tFileInputExcel component from the palette to the Designer window.



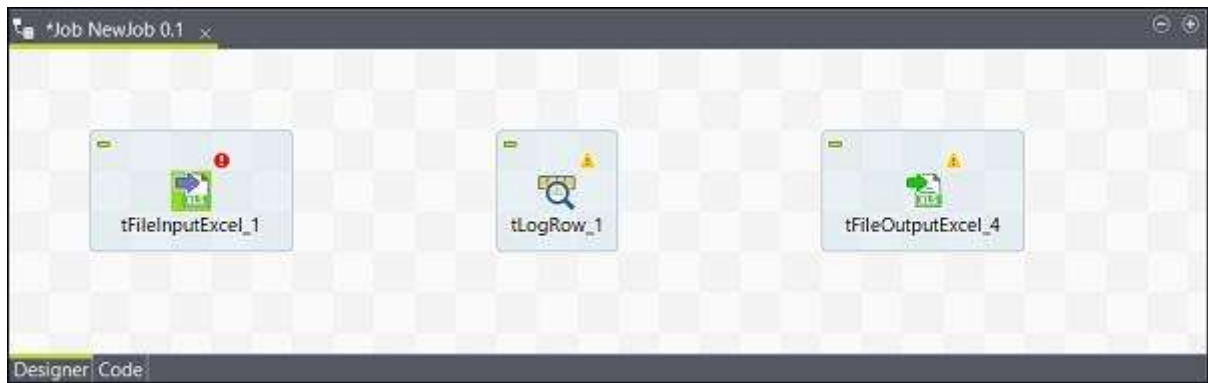
- Now if you click anywhere on the designer window, a search box will appear. Find tLogRow and select it to bring it in the designer window.



Finally, select tFileOutputExcel component from the palette and drag drop it in designer window.

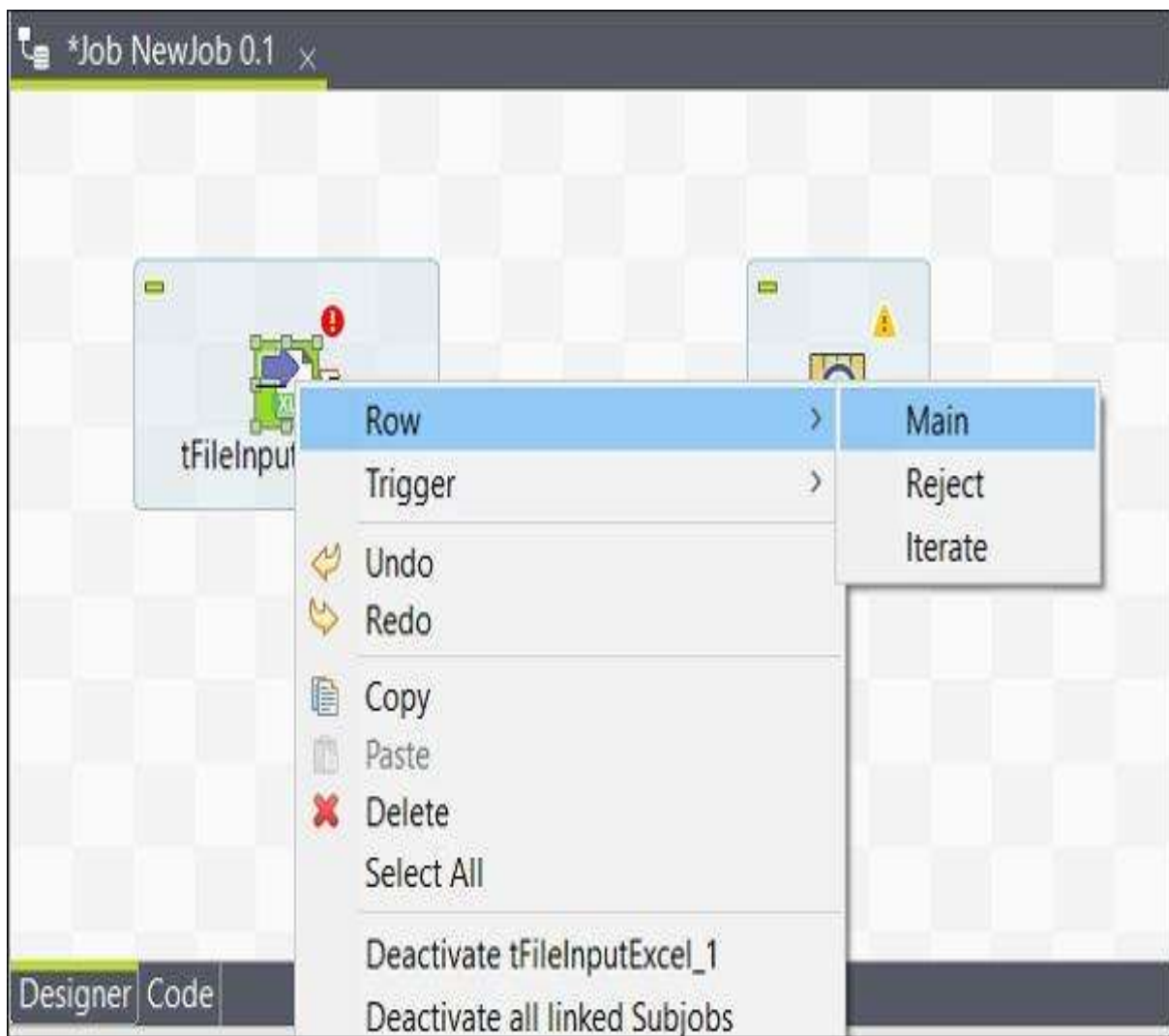


Now, the adding of the components is done.

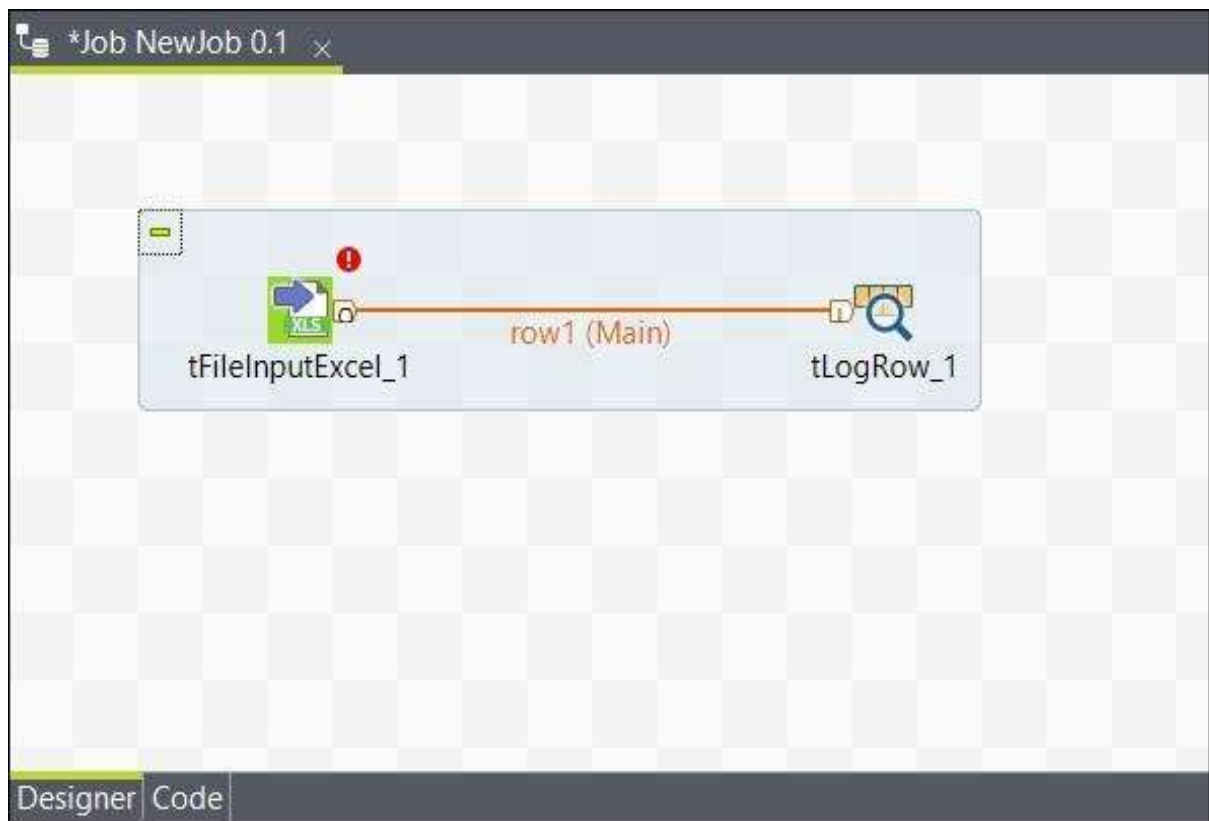


## Connecting the Components

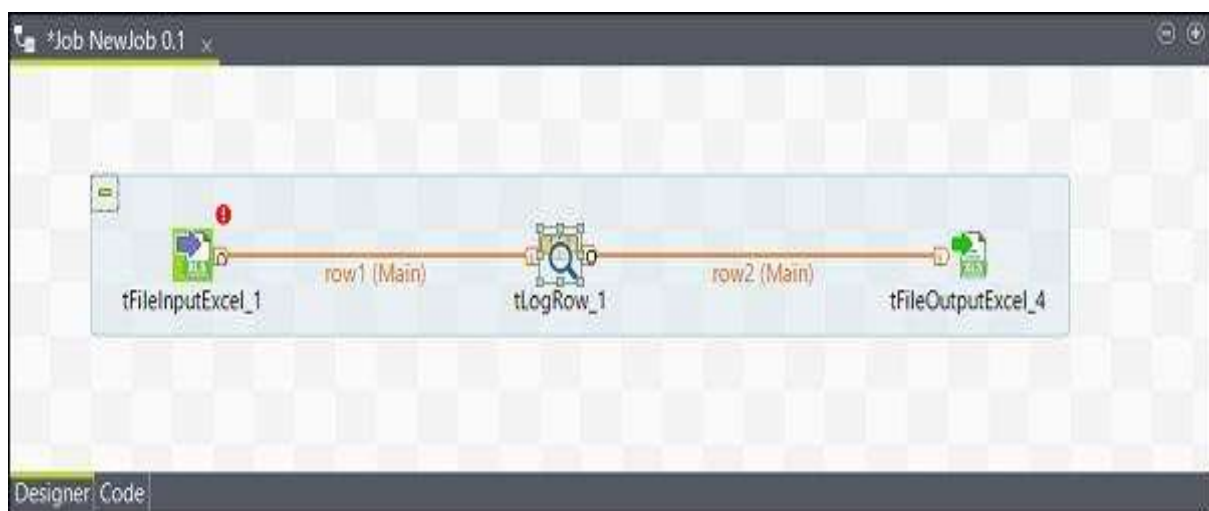
After adding components, you must connect them. Right click the first component tFileInputExcel and draw a Main line to tLogRow as shown below.



Similarly, right click tLogRow and draw a Main line on tFileOutputExcel. Now, your components are connected.



- Now added the tFileOutputExcel.



## Configuring the components

After adding and connecting the components in the job, you need to configure them. For this, double click the first component tFileInputExcel to configure it. Give the path of your input file in File name/stream as shown below.

If your 1<sup>st</sup> row in the excel is having the column names, put 1 in the Header option.

**tFileInputExcel\_1**

Property Type: Built-In

**Basic settings**

Advanced settings: ☒ Read excel2007 file format(xlsx)

Dynamic settings: File name/Stream: "D:/input\_file.xlsx"

View: ☒ All sheets

Documentation: Header: 1 Footer: 0 Limit:

☐ Affect each sheet(header&footer)

☐ Die on error

First column: 1 Last column:

Schema: Built-In Edit schema

Click Edit schema and add the columns and its type according to your input excel file. Click Ok after adding the schema.

**Schema of tFileInputExcel\_1**

tFileInputExcel\_1

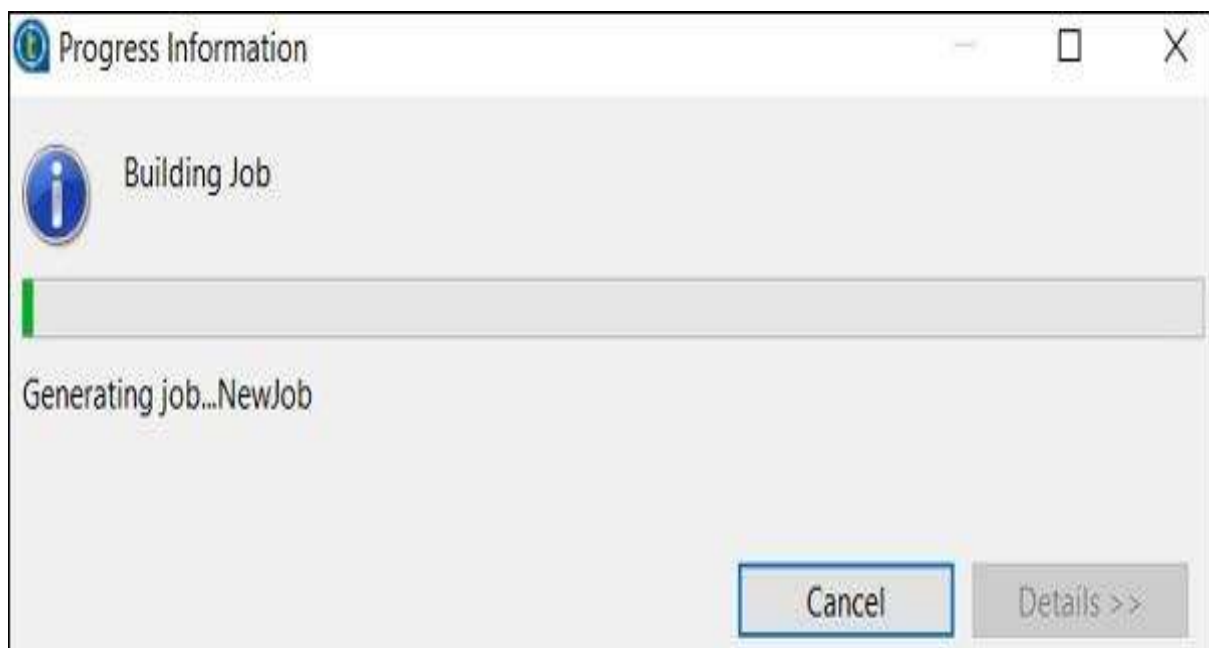
Column	K	Type	<input checked="" type="checkbox"/> N.	Date Pattern ...	Length	Precisi	Defa	Comm
id	<input checked="" type="checkbox"/>	int	<input type="checkbox"/>					
name	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>					
gender	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>					
salary	<input type="checkbox"/>	Float	<input checked="" type="checkbox"/>					
department	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>					
city	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>					

OK Cancel

Click Yes.



In tLogRow component, click on sync columns and select the mode in which you want to generate the rows from your input. Here we have selected Basic mode with “,” as field separator.



Finally, in tFileOutputExcel component, give the path of file name where you want to store.

**tFileOutputExcel\_4**

Property Type: Built-In

**Basic settings**

Advanced settings: ☐ Write excel2007 file format(xlsx)

Dynamic settings: ☐ Use Output Stream

View: File Name: "D:/Output\_file.xls"

Documentation: Sheet name: "Sheet1"

☐ Include header

☐ Append existing file

☐ Is absolute Y pos.

Font: Default

☐ Define all columns auto size

Define column auto size:

Column	Auto size
id	<input type="checkbox"/>
name	<input type="checkbox"/>
gender	<input type="checkbox"/>
salary	<input type="checkbox"/>
department	<input type="checkbox"/>
city	<input type="checkbox"/>

Schema: Built-In Edit schema Sync columns

your output excel file with the sheet name. **Click on sync columns.**

## Executing the Job

Once you are done with adding, connecting and configuring your components, you are ready to execute your Talend job. Click Run button to begin the execution.

Job(NewJob 0.1) Contexts(NewJob) Component Run (Job NewJob)

**Job NewJob**

Basic Run Execution Run Kill Clear

Debug Run

**tLogRow\_1**

Schema: Built-In Edit schema Sync columns

**Basic settings**

Advanced settings: Mode ☒ Basic

Dynamic settings: ☐ Table (print values in cells of a table)

View: ☐ Vertical (each row is a key/value list)

Documentation: Field Separator: ","

☐ Print header

☐ Print component unique name in front of each output row

☐ Print schema column name in front of each value

☐ Use fixed length for values

☒ Print content with log4j



You will see the output in the basic mode with “,” separator.

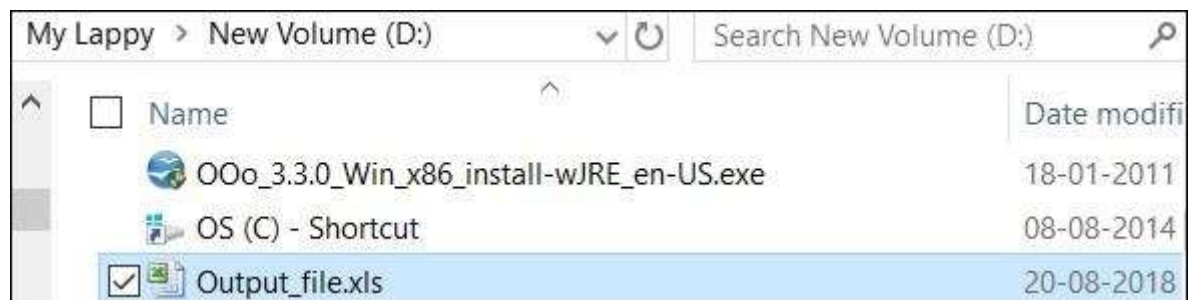
The screenshot shows the SAP Job Designer interface for a job named 'Job NewJob 0.1'. The job is composed of three steps: 'tfileInputExcel\_1', 'tLogRow\_1', and 'tfileOutputExcel\_4'. The job has been executed, and the output is displayed in the 'Basic Run' tab. The output shows a list of 200 rows of data, formatted as a CSV with a comma separator. The data is as follows:

Name	Value
180.Fanechko,Female,3032.62,Human Resources,san francisco	
181.Corby,Male,3578.24,Marketing,new york	
182.Ethelbert,Male,5529.83,Marketing,new york	
183.Court,Male,3115.47,Product Management,new york	
184.Rich,Male,4123.35,Training,san francisco	
185.Brocky,Male,5451.75,Marketing,new york	
186.Tabitha,Female,3470.8,Human Resources,san francisco	
187.Stirling,Male,4596.85,Services,san francisco	
188.Euell,Male,4623.17,Research and Development,Washington dc	
189.Rione,Female,5460.88,Marketing,new york	
190.Leonardo,Male,5219.56,Human Resources,chicago	
191.Freeland,Male,5186.41,Research and Development,los angeles	
192.Horacio,Male,4238.18,Marketing,atlanta	
193.Berk,Male,4539.3,Training,new york	
194.Siegfried,Male,3002.52,Business Development,atlanta	
195.Emanuel,Male,5048.82,Sales,Washington dc	
196.Lily,Female,3071.47,Marketing,atlanta	
197.Felita,Female,3122.53,Product Management,atlanta	
198.Cora,Female,5213.28,Marketing,new york	
199.Ruth,Female,4479.62,Human Resources,new york	
200.Bevon,Male,3931.6,Legal,chicago	
[statistics] disconnected	

The job ended at 00:54 20/08/2010. [exit code=0]



You can also see that your output is saved as an excel at the output path you mentioned.



- In this way all XML file, Json file, excel file and other file convert into .csv file to analyze the data.

- Importing Transaction Table.

minio\_project - Power BI Desktop

ASHISH KUMAR SINGH

File Home Help Table tools Column tools

Name

sales\_amount

Format

Currency

Summarize by

Date

Data category

Uncategorized

Sort by

column

Data groups

Groups

Manage relationships

Relationships

New columns

Calculations

Structure

Formatting

Properties

product\_code

customer\_code

market\_code

order\_date

order\_qty

currency

sales\_amount

unit

Prod006	cus000	Mar011	29-12-2019	1	INR	9,100.00	1010
Prod006	cus000	Mar011	30-12-2019	1	INR	9,100.00	1010
Prod006	cus000	Mar011	31-12-2019	1	INR	9,100.00	1010
Prod006	cus000	Mar011	01-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	02-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	03-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	04-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	05-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	06-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	07-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	08-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	09-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	10-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	11-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	12-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	13-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	14-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	15-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	16-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	17-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	18-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	19-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	20-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	21-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	22-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	23-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	24-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	25-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	26-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	27-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	28-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	29-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	30-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	31-01-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	01-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	02-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	03-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	04-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	05-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	06-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	07-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	08-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	09-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	10-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	11-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	12-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	13-02-2020	1	INR	9,100.00	1010
Prod006	cus000	Mar011	14-02-2020	1	INR	9,100.00	1010

Fields

Search

customers

data

markets

products

transactions

currency

customer\_code

market\_code

order\_code

product\_code

sales\_amount

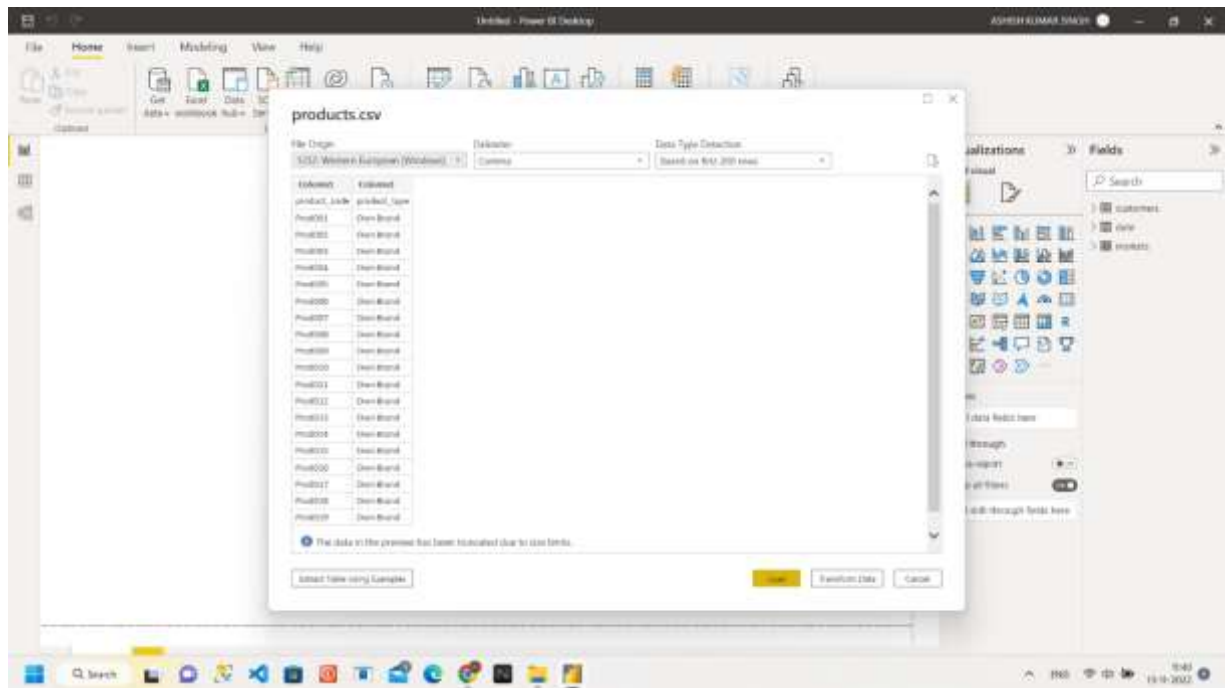
sales\_qty

unit

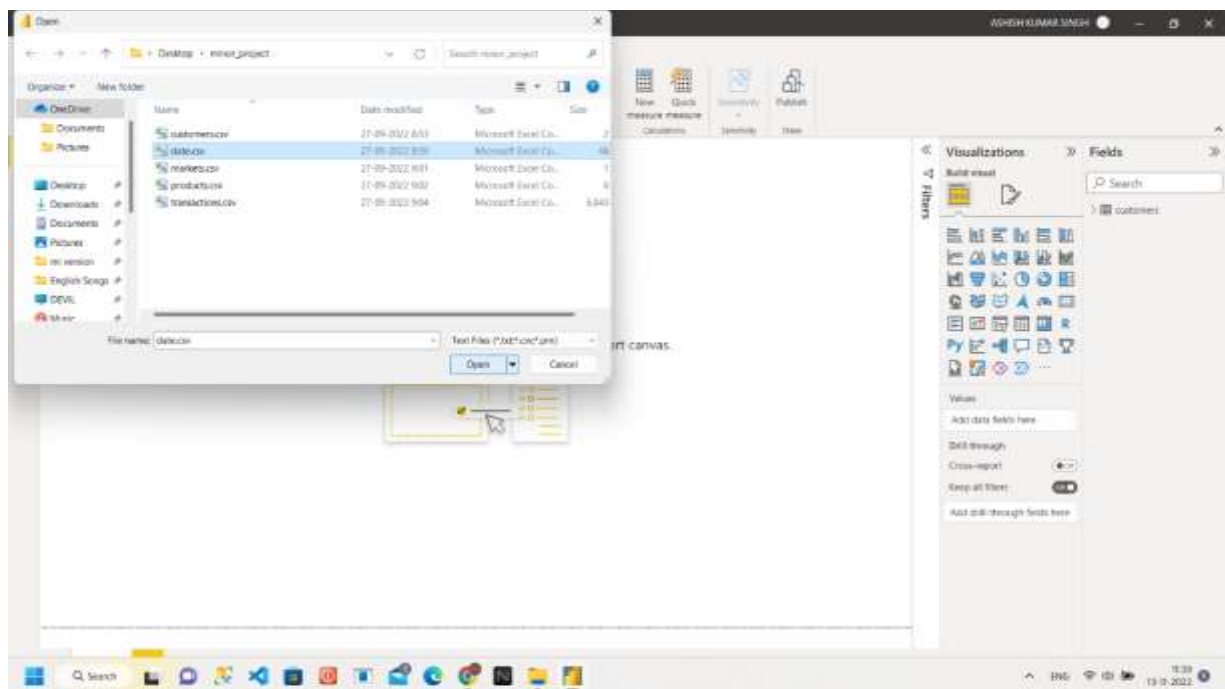
- Importing Product Table In Power BI.

[illegible]

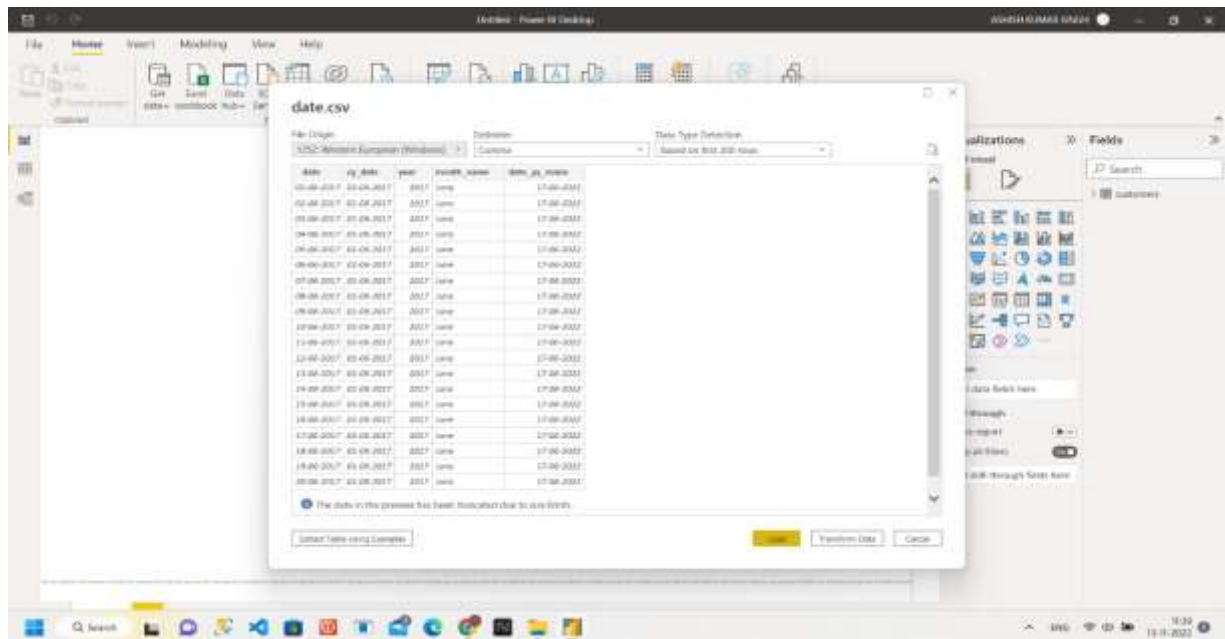
- Load the product table.



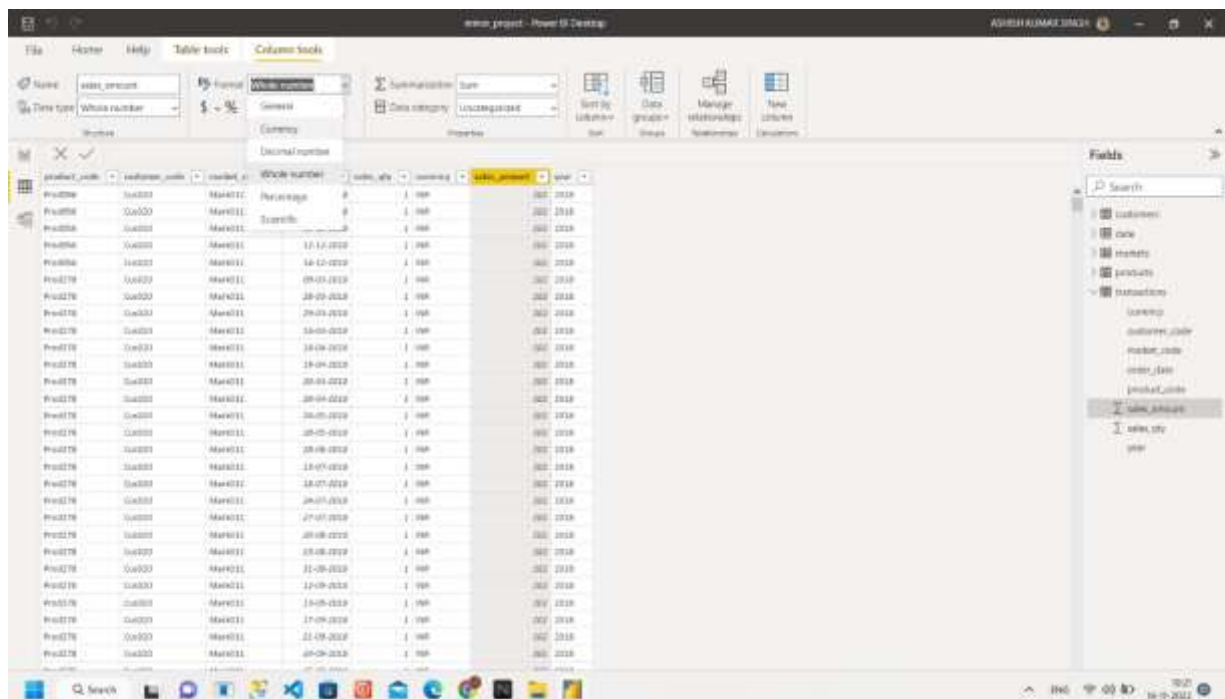
- Importing The date.csv file.



- Load the date.csv file.



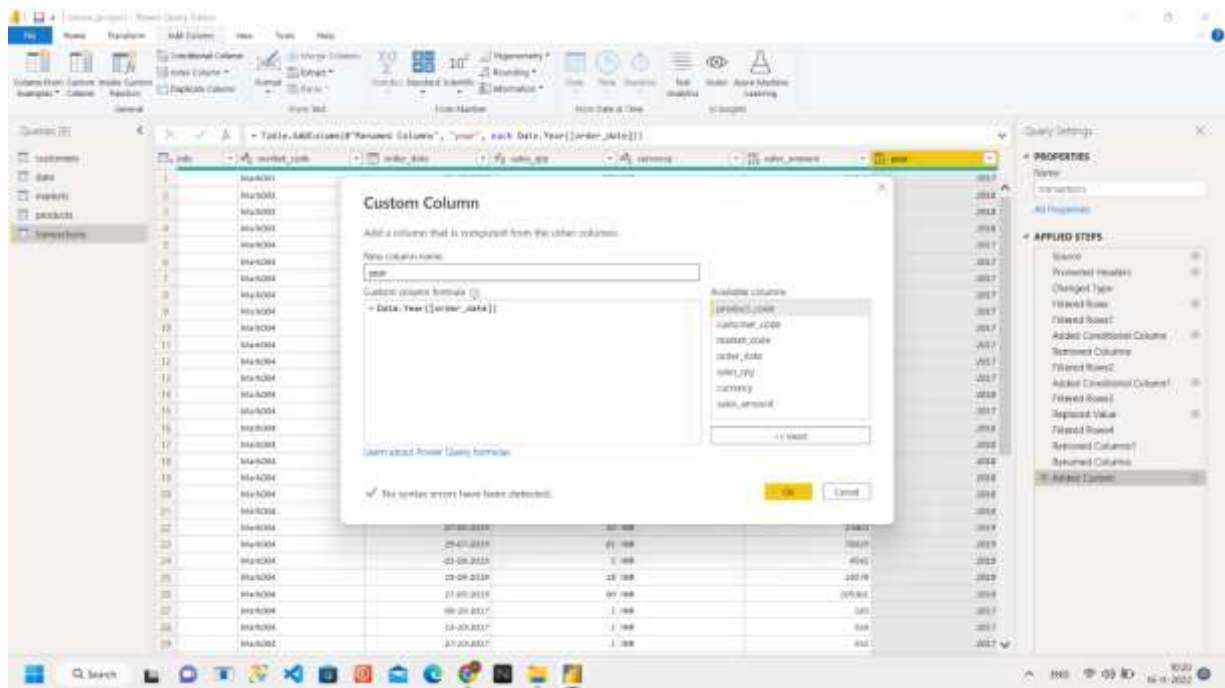
- Importing Market Table In Power BI.



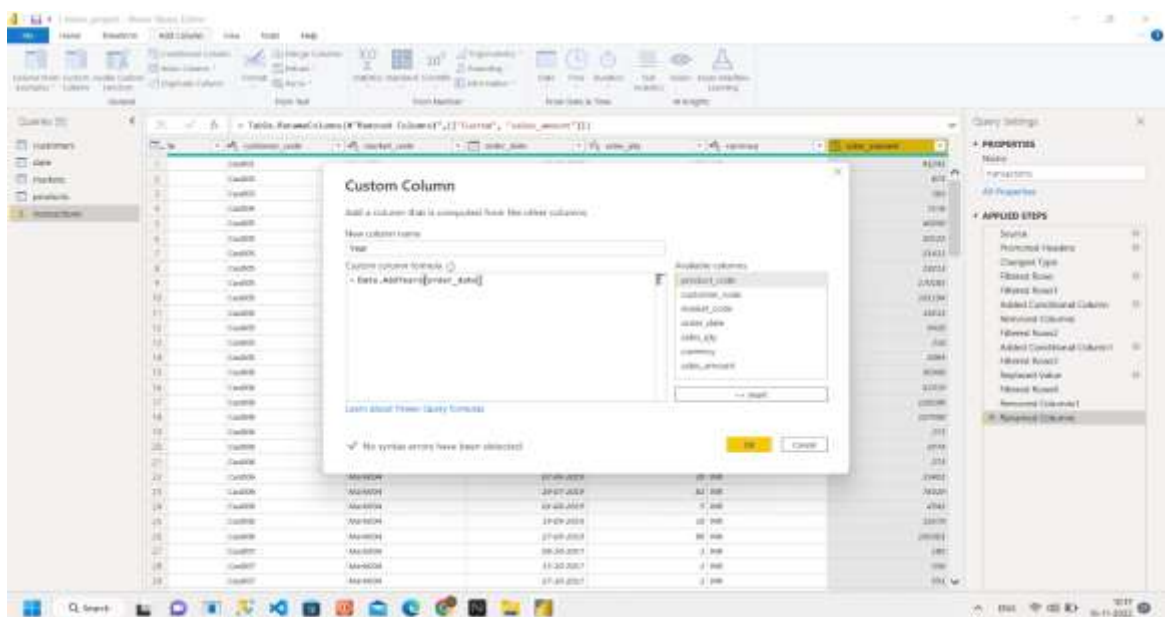
- 
- The screenshot shows the Microsoft Access application window. The title bar indicates the file is 'Untitled - Power BI Desktop'. The 'Table Tools' ribbon is active, displaying various options for working with tables. The 'Fields' task pane on the right shows a list of fields for the 'customers' table. The fields are listed in a table with three columns: 'Field Name', 'Field Type', and 'Field Size'.
- | Field Name    | Field Type | Field Size |
|---------------|------------|------------|
| customer_id   | AutoNumber | 4          |
| customer_name | Text       | 50         |
| customer_type | Text       | 50         |
| large_store   | Yes/No     |            |
| small_store   | Yes/No     |            |
| branch        | Text       | 50         |
| address       | Text       | 255        |

-

- **Customize The Transaction Table In Power BI.**

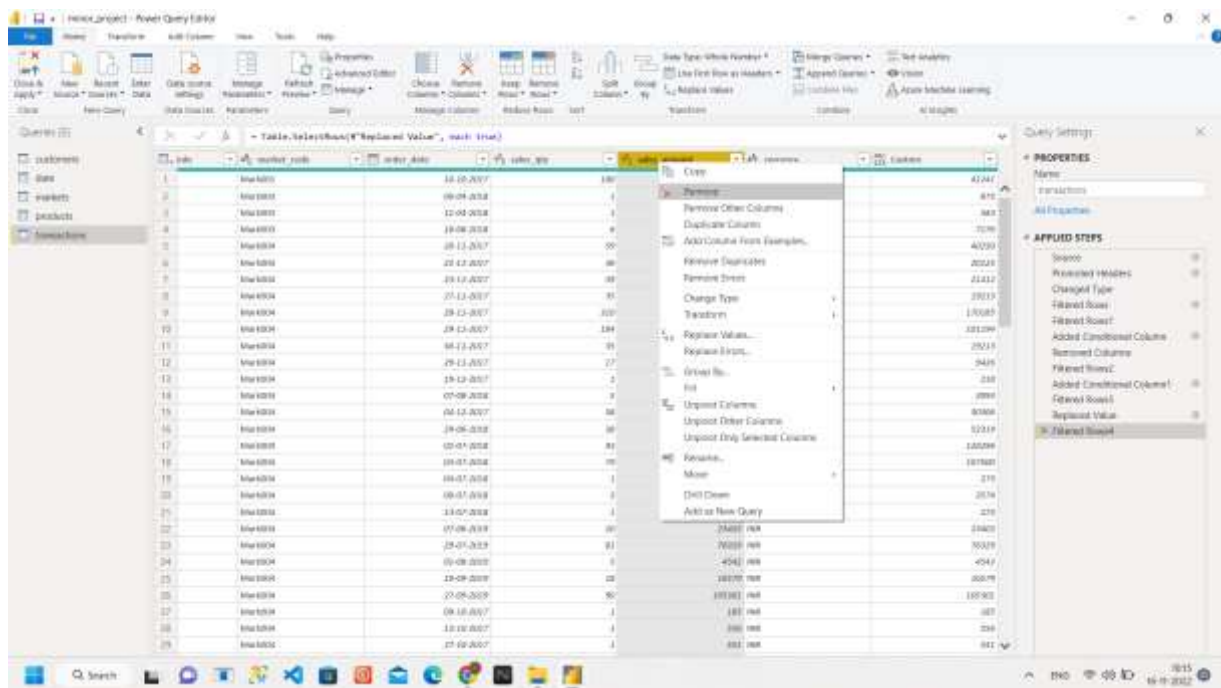


- **Customize date In Transaction Table.**



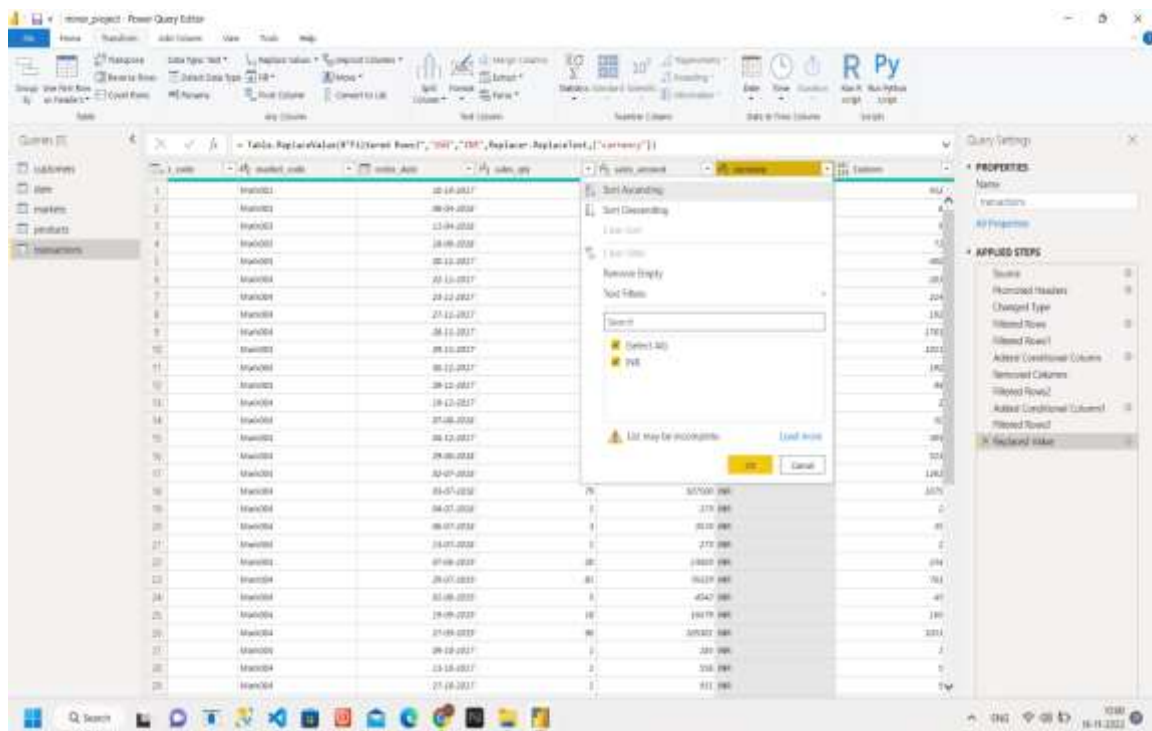
- **Remove one table In Transaction Table.**



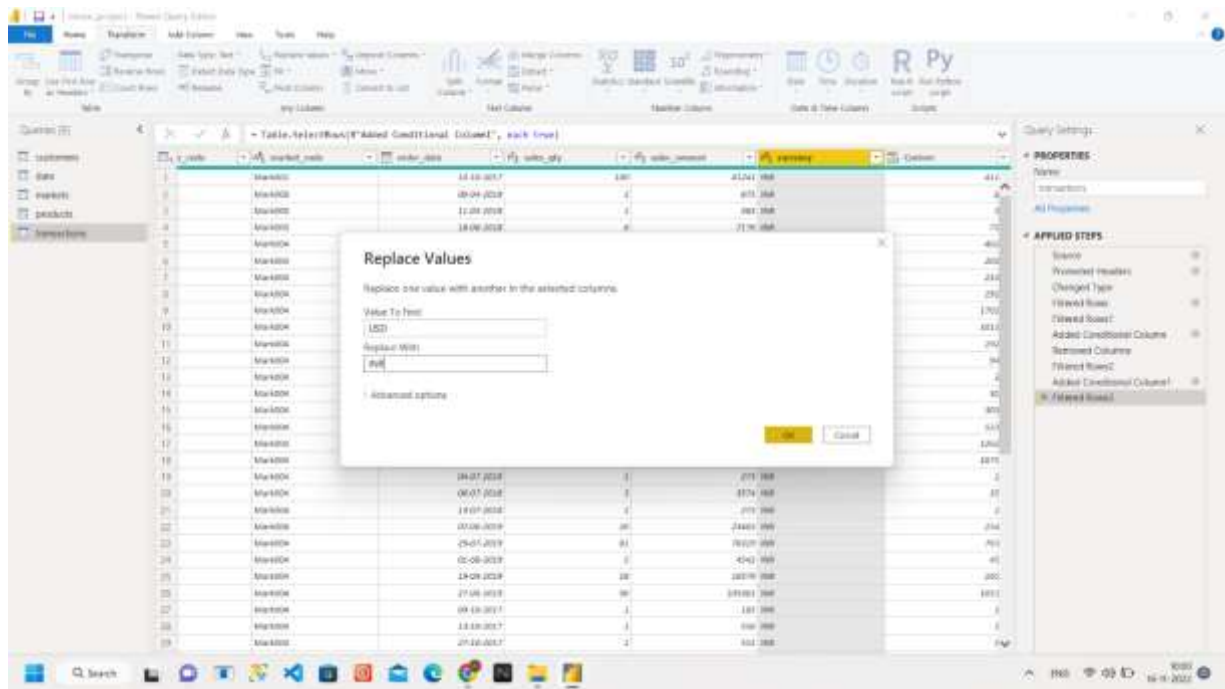


- **Change Sales amount In Transaction Table.**

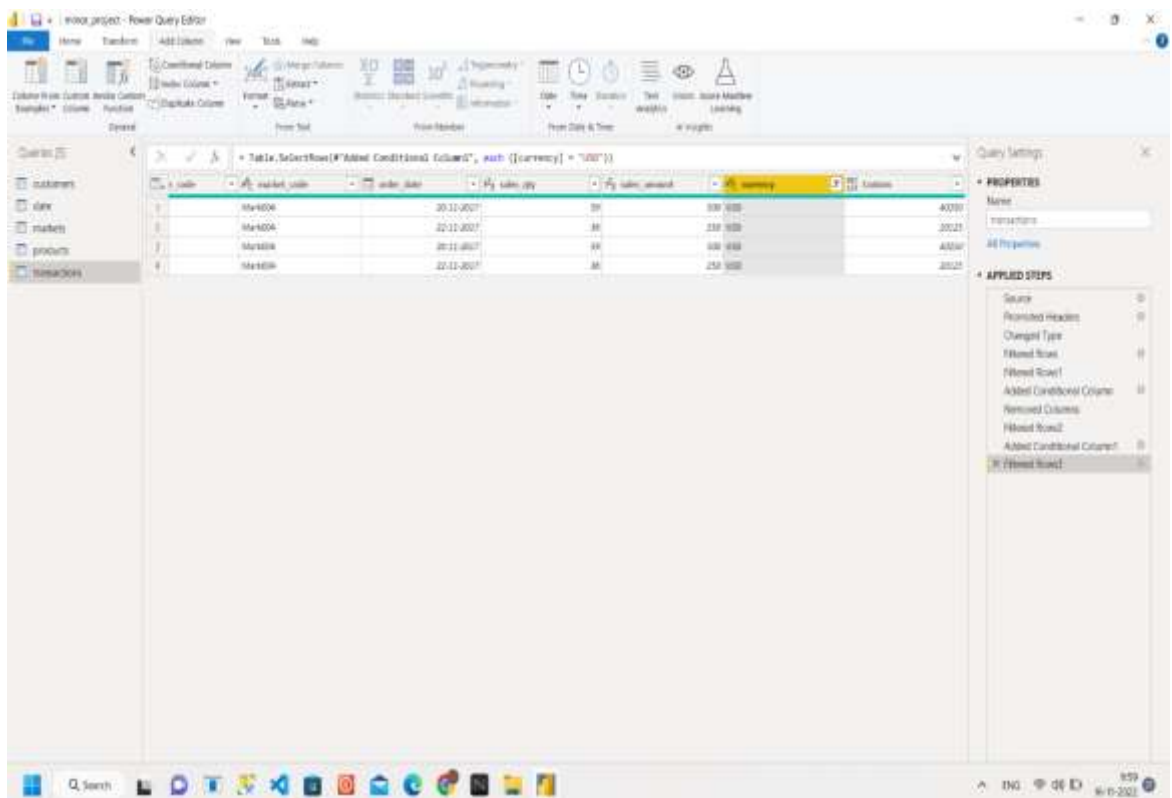
Which amounts are in INR currency this amounts are remaining same in the Transaction table in sales amount Column.



- Which amount values are in USD currency we change them into INR currency with condition statement and INR Currency remaining same.



- Identify the USD currency In Sales amount column.



- Apply the condition statement to change the USD values are in INR values.



Table: AddColumn("Filtered Row1", "Currency", Add If [currency] = "USD" then (order\_amount)/68.3 else (order\_amount))

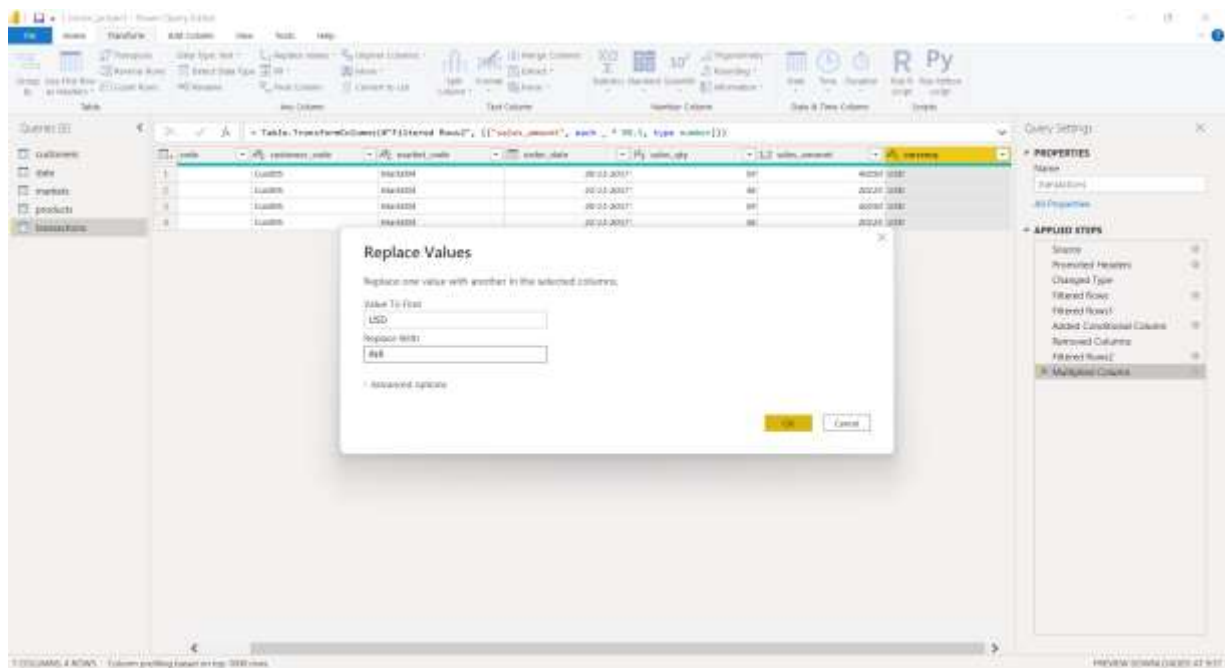
id	market_code	order_date	order_qty	order_amount	currency
1	usa000	18-09-2017	180	41241 INR	41241
2	usa000	09-09-2018	1	817 INR	817
3	usa000	12-09-2018	1	981 INR	981
4	usa000	19-09-2018	8	7176 INR	7176
5	usa000	28-12-2017	55	980 INR	40330
6	usa000	23-12-2017	86	2987 INR	20228
7	usa000	23-12-2017	38	31612 INR	21212
8	usa000	27-12-2017	31	25021 INR	23013
9	usa000	28-12-2017	109	170187 INR	150185
10	usa000	28-12-2017	124	233184 INR	221298
11	usa000	30-12-2017	31	25011 INR	23013
12	usa000	29-12-2017	27	9400 INR	9400
13	usa000	18-12-2017	2	219 INR	219
14	usa000	07-09-2018	2	9281 INR	9281
15	usa000	04-12-2017	54	30350 INR	30350
16	usa000	28-05-2018	38	52220 INR	52219
17	usa000	02-01-2018	81	126796 INR	126796
18	usa000	09-07-2018	95	127580 INR	127580
19	usa000	09-07-2018	1	219 INR	219
20	usa000	09-07-2018	3	3574 INR	3574
21	usa000	14-07-2018	1	219 INR	219
22	usa000	07-08-2018	30	23499 INR	23499
23	usa000	25-07-2018	31	70339 INR	70339
24	usa000	09-08-2018	5	4542 INR	4542
25	usa000	18-09-2018	28	18870 INR	18870
26	usa000	27-09-2018	80	189381 INR	189381
27	usa000	08-10-2017	1	181 INR	181
28	usa000	18-10-2017	1	586 INR	586
29	usa000	27-09-2017	1	591 INR	591

- Detect the all INR amounts.

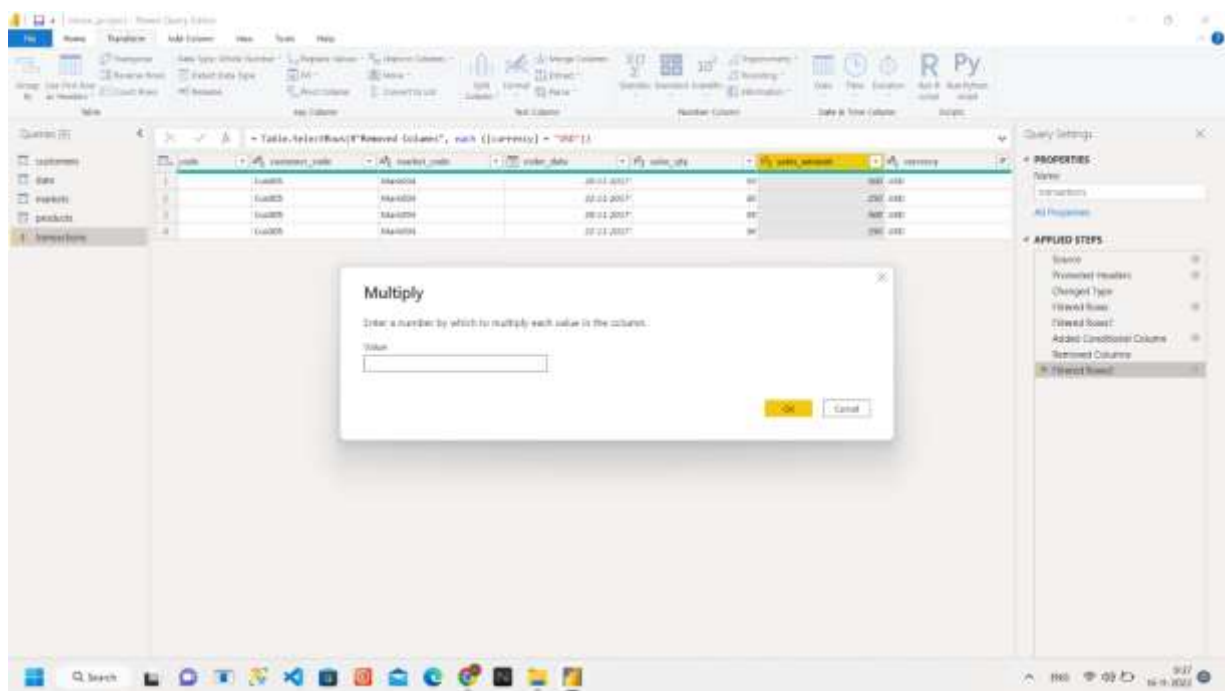
Table: ReplaceValue("Filtered Row1", "USD", "INR", Replace.ReplaceText, "currency")

id	market_code	order_date	order_qty	order_amount	currency
1	usa000	18-09-2017	180	40330 INR	INR
2	usa000	09-09-2018	1	2817 INR	INR
3	usa000	12-09-2018	1	4081 INR	INR
4	usa000	19-09-2018	8	27176 INR	INR

- Detect the all USD amount values replace In INR amount values.



- To change Into all USD amount In INR amount multiply USD amount with 80.5.



**Multiply**

Enter a number by which to multiply each value in the column.

Value:

10.5

OK Cancel

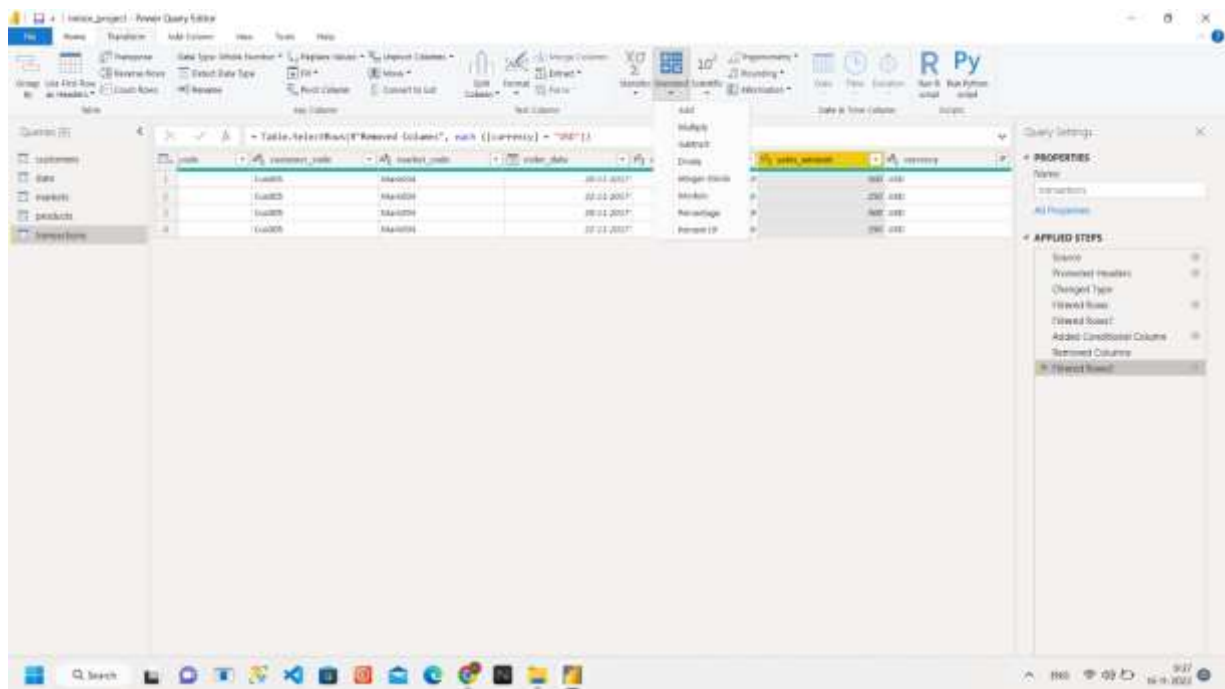
code	customer_code	market_code	order_date	sales_qty	sales_amount	currency
1	cus005	mar0004	20-11-2027	80	40250 USD	USD
2	cus005	mar0004	22-11-2027	20	20125 USD	USD
3	cus005	mar0004	20-11-2027	80	40250 USD	USD
4	cus005	mar0004	22-11-2027	20	20125 INR	INR

- All USD amount values successfully change into INR values.

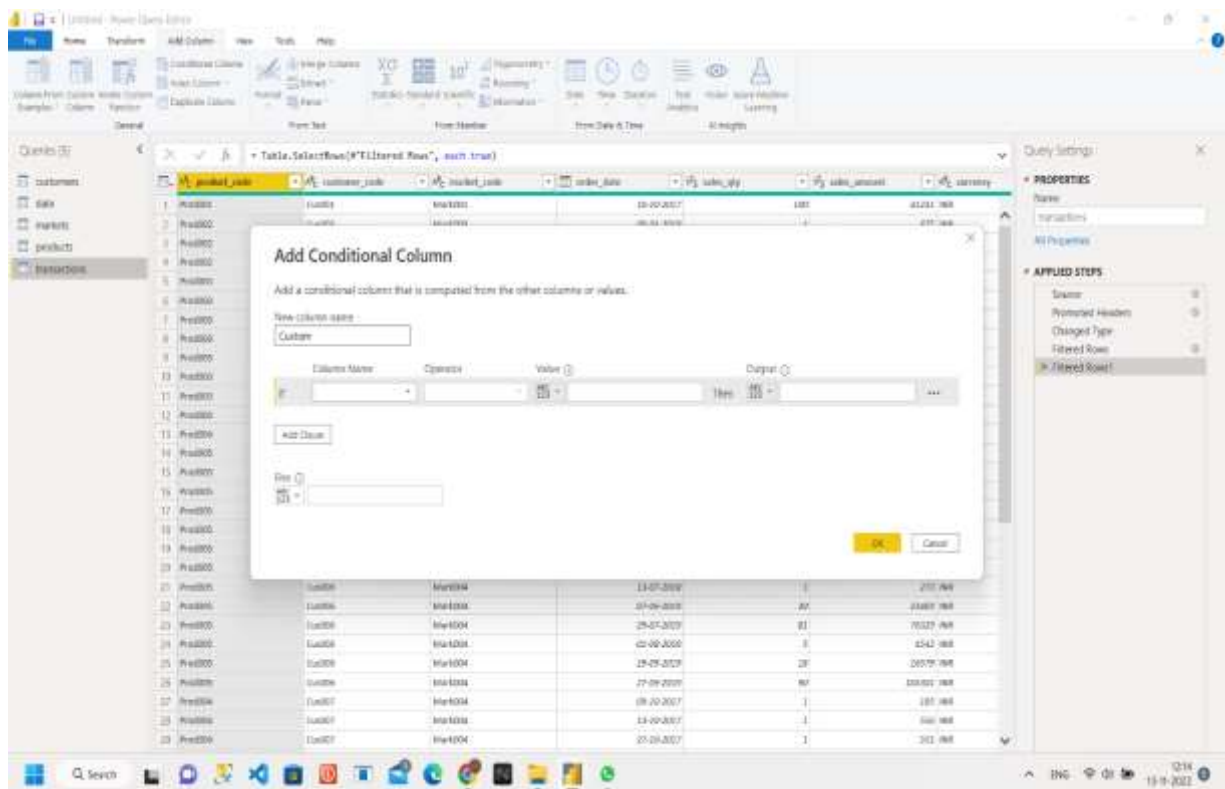
**Table.SelectRows("Filtered Rows", each ([currency] = "INR"))**

code	customer_code	market_code	order_date	sales_qty	sales_amount	currency
1	cus005	mar0004	20-11-2027	80	422625 INR	INR
2	cus005	mar0004	22-11-2027	20	211312 INR	INR
3	cus005	mar0004	20-11-2027	80	422625 INR	INR
4	cus005	mar0004	22-11-2027	20	211312 INR	INR

- Add one column In Transaction Table.



- Added one conditonal Column In Transaction Table.



- Fixed or marked the all the Tables First Rows which Indicates all the amount codes etc.

The screenshot shows the Power Query Editor interface. The main area displays a table with the following columns: product\_code, customer\_code, market\_code, order\_date, order\_qty, order\_amount, and currency. The table contains 29 rows of data. The first row is highlighted in yellow. The right sidebar shows the 'Query Settings' pane with the 'Properties' tab selected, showing the query name 'Table.SelectRows("Filtered Rows", each true)' and the 'Applied Steps' list.

product_code	customer_code	market_code	order_date	order_qty	order_amount	currency
Prod001	Cus001	Mar001	19-10-2017	100	81291 /MR	
Prod001	Cus001	Mar001	09-04-2018	1	875 /MR	
Prod001	Cus001	Mar001	11-04-2018	1	547 /MR	
Prod001	Cus001	Mar001	19-06-2018	6	7100 /MR	
Prod001	Cus001	Mar001	28-11-2017	89	500 /USD	
Prod001	Cus001	Mar001	27-11-2017	86	297 /USD	
Prod001	Cus001	Mar001	29-11-2017	39	21612 /MR	
Prod001	Cus001	Mar001	27-11-2017	81	28217 /MR	
Prod001	Cus001	Mar001	28-11-2017	200	17030 /MR	
Prod001	Cus001	Mar001	29-11-2017	184	18119 /MR	
Prod001	Cus001	Mar001	28-11-2017	37	28217 /MR	
Prod001	Cus001	Mar001	29-11-2017	17	8400 /MR	
Prod001	Cus001	Mar001	19-12-2017	1	218 /MR	
Prod001	Cus001	Mar001	07-06-2018	5	8067 /MR	
Prod001	Cus001	Mar001	04-12-2017	88	30900 /MR	
Prod001	Cus001	Mar001	29-06-2018	36	51214 /MR	
Prod001	Cus001	Mar001	03-07-2018	93	126289 /MR	
Prod001	Cus001	Mar001	03-07-2018	79	187583 /MR	
Prod001	Cus001	Mar001	09-07-2018	1	278 /MR	
Prod001	Cus001	Mar001	09-07-2018	7	1574 /MR	
Prod001	Cus001	Mar001	13-07-2018	1	278 /MR	
Prod001	Cus001	Mar001	07-08-2018	20	23483 /MR	
Prod001	Cus001	Mar001	29-07-2019	81	76309 /MR	
Prod001	Cus001	Mar001	01-08-2018	9	4547 /MR	
Prod001	Cus001	Mar001	19-09-2019	18	18570 /MR	
Prod001	Cus001	Mar001	27-09-2019	80	108361 /MR	
Prod001	Cus001	Mar001	09-10-2017	1	187 /MR	
Prod001	Cus001	Mar001	13-10-2017	1	596 /MR	
Prod001	Cus001	Mar001	27-10-2017	1	593 /MR	

## Step 04 : Change The Type Of The Coumns Of All the Tables.

- Change The Type Of The Product Code In Transaction Table Column.

Table: TransactionTypes[\*Promoted Headers, {"product\_code", type text}, {"customer\_code", type text}, {"market\_code", type text}]

	product_code	customer_code	market_code	order_date	order_qty	order_amount	currency
1	P00000	C00001	M00001	18-10-2017	100	41,001.000	
2	P00000	C00001	M00001	19-10-2017	5	1.700	
3	P00000	C00001	M00001	19-04-2018	1	4,517.000	
4	P00000	C00001	M00001	11-04-2018	1	180.000	
5	P00000	C00004	M00001	19-09-2018	0	71,707.000	
6	P00000	C00005	M00001	28-12-2017	80	3307.1250	
7	P00000	C00005	M00001	28-12-2017	80	280.1450	
8	P00000	C00005	M00004	25-12-2017	35	214,117.000	
9	P00000	C00005	M00004	25-12-2017	31	10,211.000	
10	P00000	C00005	M00004	28-12-2017	200	1,703,817.000	
11	P00000	C00005	M00004	28-12-2017	180	1,811,790.000	
12	P00000	C00005	M00004	28-12-2017	25	10,211.000	
13	P00004	C00005	M00004	28-12-2017	2,1	940,000.000	
14	P00004	C00005	M00004	28-12-2017	1	2,137.000	
15	P00000	C00005	M00004	07-08-2018	0	40,000.000	
16	P00000	C00005	M00004	04-12-2017	10	30,000.000	
17	P00000	C00005	M00004	29-09-2018	50	52,539.000	
18	P00000	C00005	M00004	02-07-2018	50	1,64,780.000	
19	P00000	C00005	M00004	01-07-2018	10	1,07,500.000	
20	P00000	C00005	M00004	04-07-2018	1	270.000	
21	P00000	C00005	M00004	04-07-2018	0	50,000.000	
22	P00000	C00005	M00004	13-07-2018	1	270.000	
23	P00000	C00005	M00004	07-08-2017	100	21,401.000	
24	P00000	C00005	M00004	28-07-2018	11	10,000.000	
25	P00000	C00005	M00004	02-08-2018	0	4,94,7.000	
26	P00000	C00005	M00004	19-08-2018	10	1,05,171.000	
27	P00000	C00005	M00004	27-08-2018	60	1,03,381.000	
28	P00000	C00007	M00004	08-10-2017	1	180.000	
29	P00000	C00007	M00004	23-10-2017	1	180.000	
30	P00000	C00007	M00004	12-10-2017	1	101.000	

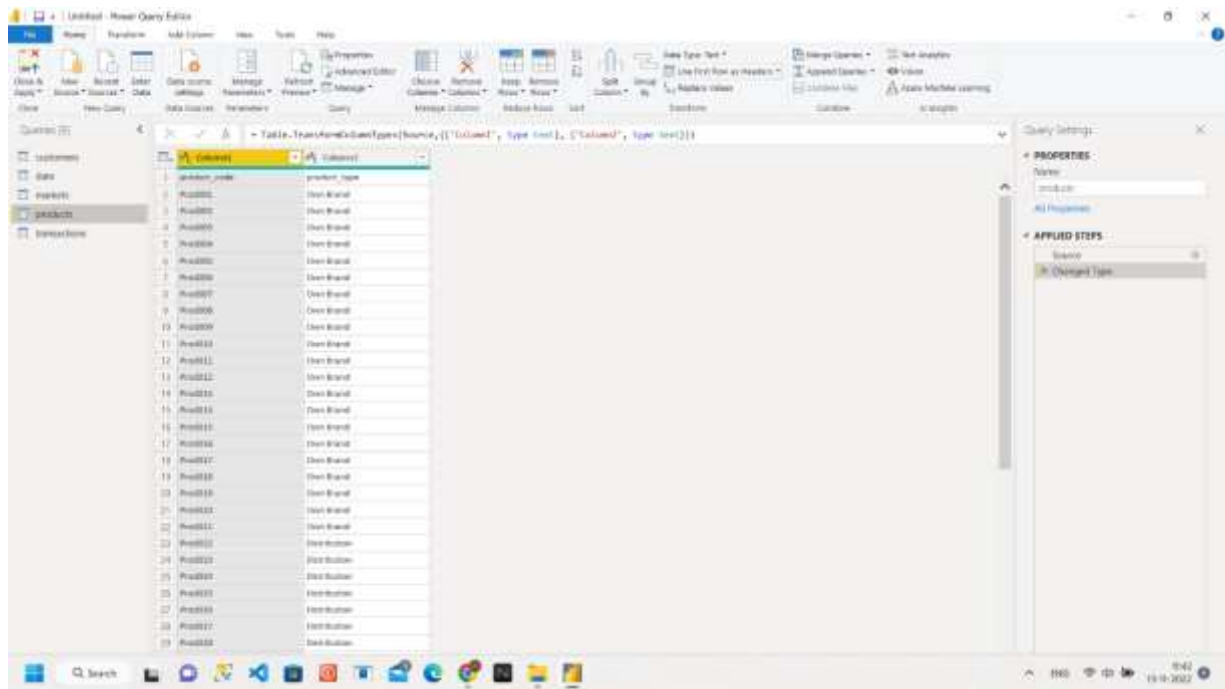
- Successfully Change The type of the product code in Transaction Table.

Table: TransactionTypes[\*Promoted Headers, {"product\_code", type text}, {"product\_type", type text}]

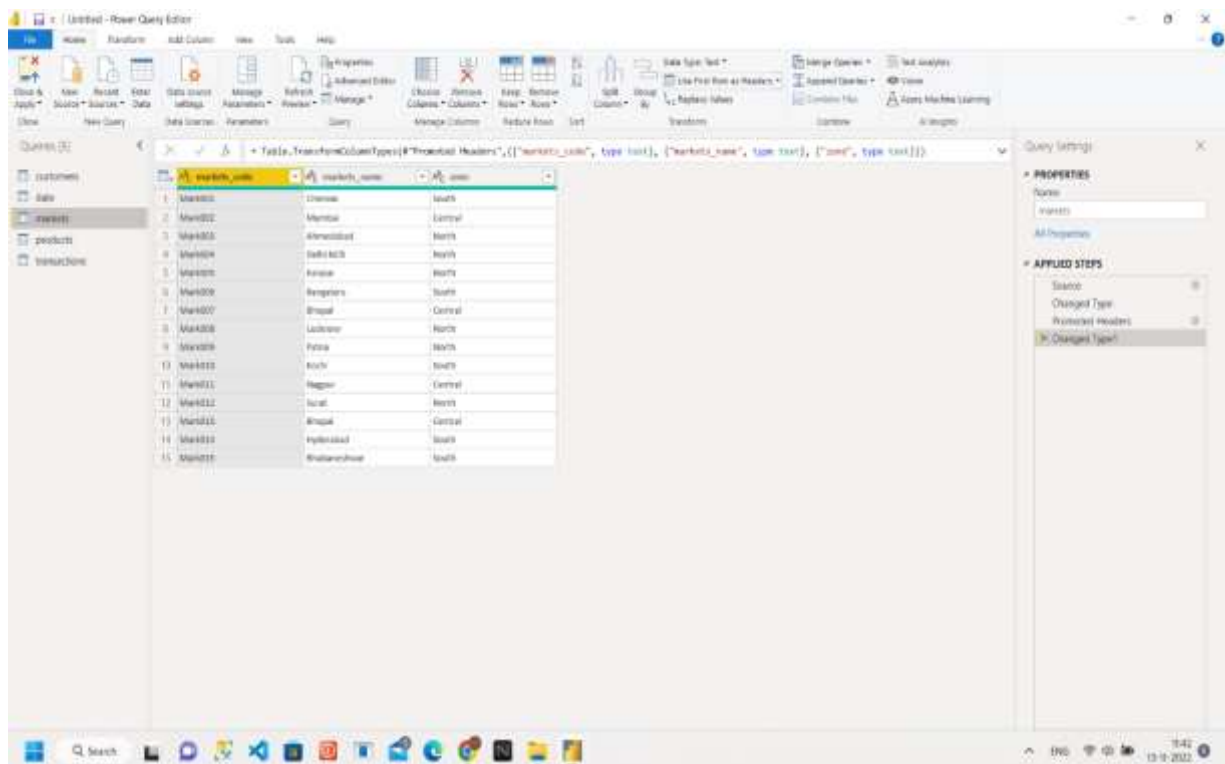
	product_code	product_type
1	P00000	Own Brand
2	P00000	Own Brand
3	P00000	Own Brand
4	P00004	Own Brand
5	P00000	Own Brand
6	P00000	Own Brand
7	P00007	Own Brand
8	P00000	Own Brand
9	P00000	Own Brand
10	P00000	Own Brand
11	P00011	Own Brand
12	P00012	Own Brand
13	P00013	Own Brand
14	P00014	Own Brand
15	P00015	Own Brand
16	P00016	Own Brand
17	P00017	Own Brand
18	P00018	Own Brand
19	P00019	Own Brand
20	P00020	Own Brand
21	P00021	Own Brand
22	P00022	Distribution
23	P00023	Distribution
24	P00024	Distribution
25	P00025	Distribution
26	P00026	Distribution
27	P00027	Distribution
28	P00028	Distribution
29	P00029	Distribution

- Change the type of the product code in Products column.

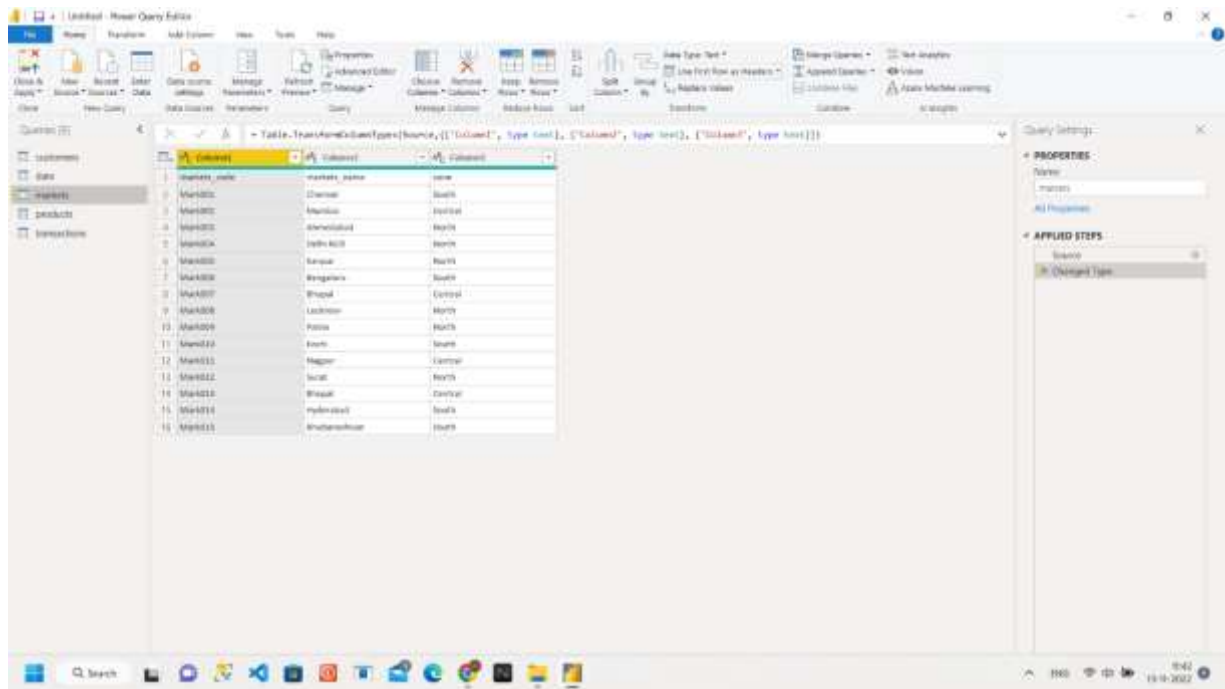




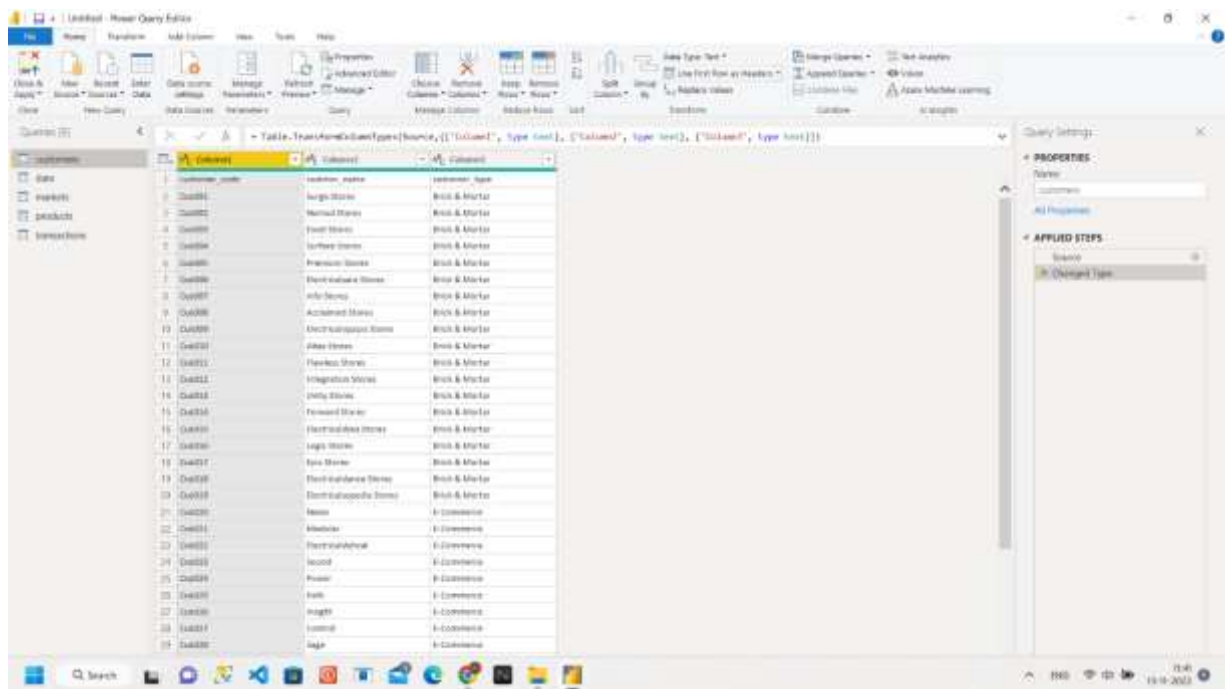
- Change the type of the market code in markets column.



- Successfully change the type of market code column in markets data.

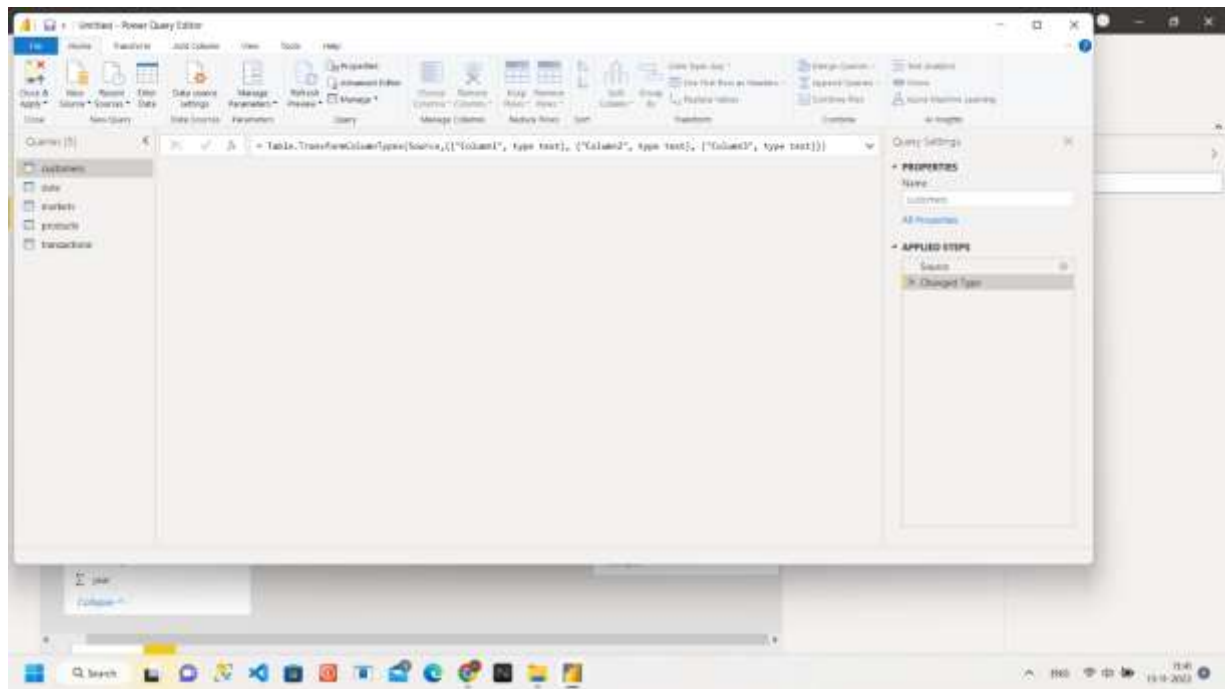


- Change the type of the customer code column in customers csv file.



- Successfully change the type of the column.





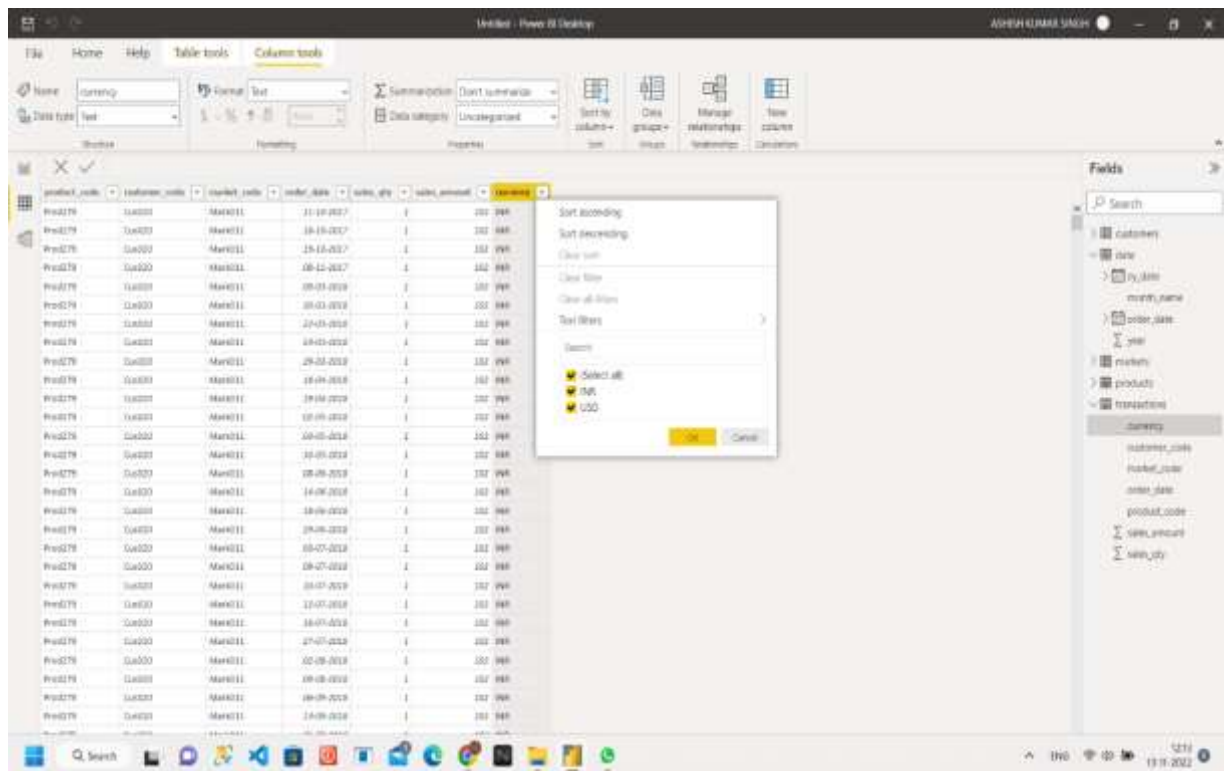
**Step 05 : Delete The Column.**

- Delete the dd\_yy\_mm column in Transaction Column.

The screenshot shows the 'transactions.csv' data source window in Microsoft Power BI Desktop. The window displays a table with the following columns: product\_code, customer\_code, market\_code, order\_date, sales\_qty, sales\_amount, and currency. The data is loaded from a CSV file, and the window includes a 'Data Type Detection' section indicating the data is based on 612,203 rows. The background shows the Power BI interface with the 'Fields' pane on the right.

product_code	customer_code	market_code	order_date	sales_qty	sales_amount	currency
Prod001	Cus001	Mar001	05-10-2017	830	62,041	INR
Prod001	Cus001	Mar001	06-05-2018	8	-1	INR
Prod001	Cus001	Mar001	09-09-2018	7	251	INR
Prod001	Cus001	Mar001	11-08-2018	7	261	INR
Prod001	Cus001	Mar001	20-08-2018	6	1170	INR
Prod001	Cus001	Mar001	20-11-2017	35	560	INR
Prod001	Cus001	Mar001	21-11-2017	35	280	INR
Prod001	Cus001	Mar001	21-11-2017	35	31611	INR
Prod001	Cus001	Mar001	21-11-2017	35	19011	INR
Prod001	Cus001	Mar001	26-11-2017	310	1,70,991	INR
Prod001	Cus001	Mar001	28-11-2017	194	15,119	INR
Prod001	Cus001	Mar001	30-11-2017	35	19011	INR
Prod001	Cus001	Mar001	29-11-2017	17	9401	INR
Prod001	Cus001	Mar001	29-12-2017	1	101	INR
Prod001	Cus001	Mar001	01-08-2018	9	9301	INR
Prod001	Cus001	Mar001	04-12-2017	35	30,991	INR
Prod001	Cus001	Mar001	29-05-2018	35	11,011	INR
Prod001	Cus001	Mar001	02-07-2018	83	1,20,981	INR
Prod001	Cus001	Mar001	02-07-2018	79	1,81,780	INR
Prod001	Cus001	Mar001	04-07-2018	1	111	INR

**Step 06 : We place the Tables in Ascending order or Descending order.**



## Step 07 : After that we fixed the all tables First rows.

Power Query Editor - Query Settings

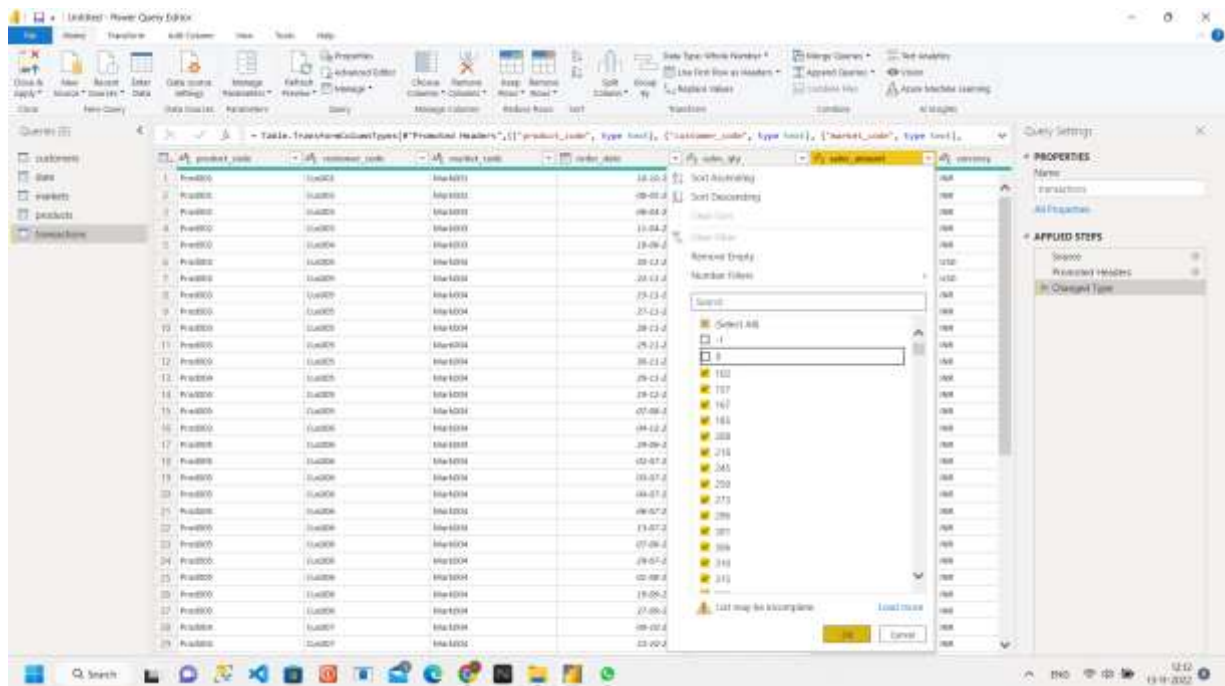
Query: Table.SelectRows("Filtered Rows", each true)

	product_code	customer_code	market_code	order_date	order_qty	sales_amount	currency
1	Prod001	Cus001	Mar001	19-10-2017	100	41081	INR
2	Prod002	Cus001	Mar003	06-04-2018	1	875	INR
3	Prod002	Cus001	Mar009	12-04-2019	1	949	INR
4	Prod002	Cus004	Mar003	18-06-2018	9	7139	INR
5	Prod001	Cus001	Mar004	20-11-2017	59	580	USD
6	Prod001	Cus005	Mar004	22-11-2017	46	290	USD
7	Prod001	Cus001	Mar004	21-11-2017	59	2141	INR
8	Prod001	Cus001	Mar004	27-11-2017	37	2821	INR
9	Prod001	Cus001	Mar003	28-11-2017	330	17000	INR
10	Prod001	Cus001	Mar004	29-11-2017	134	11174	INR
11	Prod001	Cus001	Mar004	08-11-2017	81	2821	INR
12	Prod004	Cus001	Mar004	28-11-2017	27	9405	INR
13	Prod004	Cus005	Mar004	19-11-2017	1	218	INR
14	Prod001	Cus001	Mar004	07-01-2018	9	895	INR
15	Prod001	Cus004	Mar004	04-11-2017	59	3000	INR
16	Prod001	Cus006	Mar004	29-06-2018	38	11339	INR
17	Prod001	Cus006	Mar004	03-07-2018	67	16290	INR
18	Prod001	Cus006	Mar004	03-07-2018	79	20750	INR
19	Prod005	Cus006	Mar004	04-07-2018	1	270	INR
20	Prod001	Cus006	Mar004	09-07-2018	1	654	INR
21	Prod001	Cus006	Mar004	13-07-2018	1	271	INR
22	Prod005	Cus006	Mar004	07-08-2019	20	23405	INR
23	Prod001	Cus006	Mar004	29-07-2019	41	26120	INR
24	Prod001	Cus006	Mar004	01-08-2019	9	4542	INR
25	Prod001	Cus006	Mar004	18-08-2019	18	18579	INR
26	Prod001	Cus006	Mar004	27-08-2019	90	10000	INR
27	Prod004	Cus007	Mar004	08-10-2017	1	135	INR
28	Prod004	Cus007	Mar004	13-10-2017	1	599	INR
29	Prod004	Cus007	Mar004	27-10-2017	1	551	INR

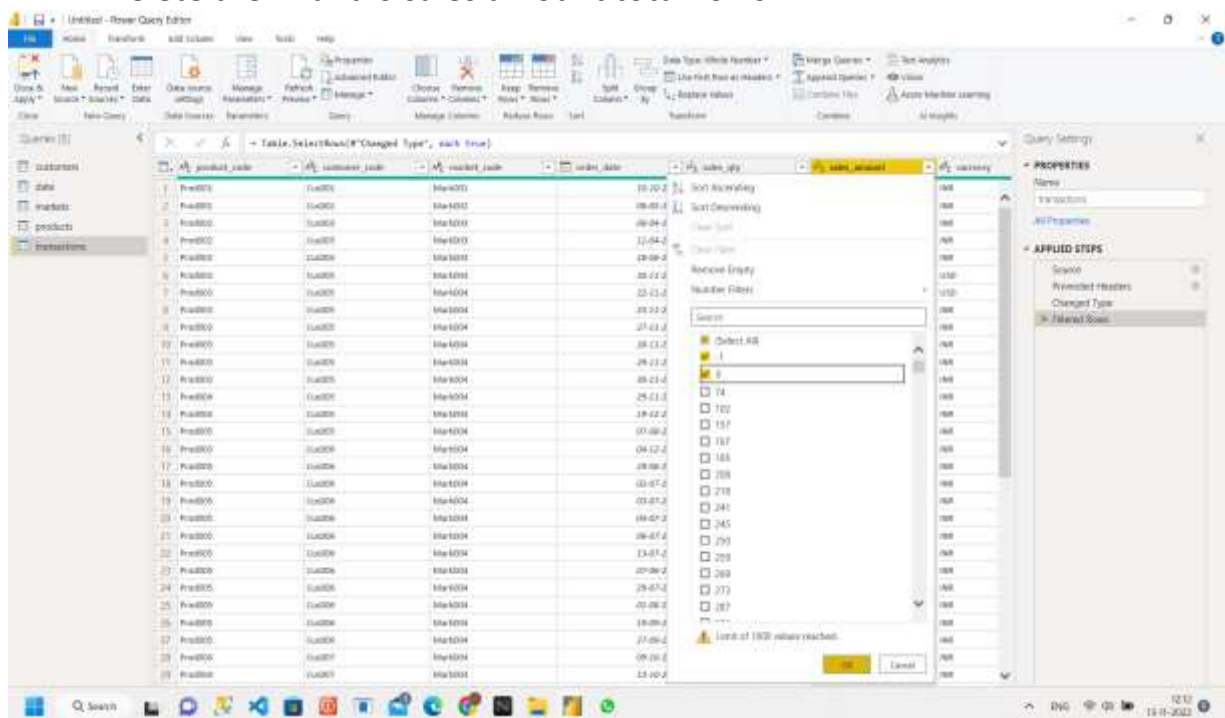
Applied Steps: Source, Promoted Headers, Changed Type, Filtered Rows, Filtered Rows

## Step 8 : Filtering The Sales amount.

After that we go into transaction tables and go into sales amount and filter the rows with respect of sales amount. We delete the 0 and -1 sales amount rows to analyze the data make it very easier.



- Delete the -1 and 0 sales amount total rows.



- Sorting the sales amount rows in Ascending order.

The screenshot displays the Microsoft Power Query Editor interface. The main data table is titled "Table.SelectRows(\*'Changed Type', each true)" and contains columns: product\_code, customer\_code, market\_code, order\_date, order\_qty, sales\_amount, and currency. A context menu is open over the "sales\_amount" column, with the "Sort Ascending" option selected. The "Sort" dialog box is also visible, showing "Sort By" set to "sales\_amount" and "Sort Order" set to "Ascending". The "APPLIED STEPS" pane on the right shows the sequence of transformations: Source, Expanded Headers, Changed Type, and Filtered Rows.

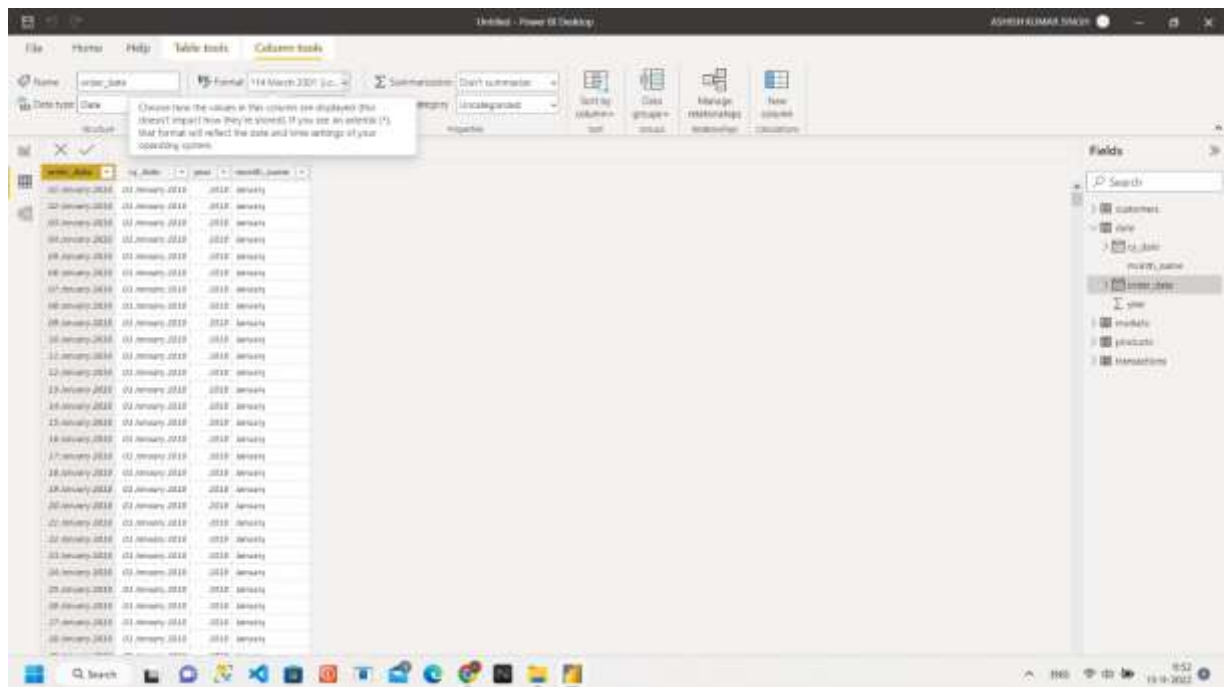
product_code	customer_code	market_code	order_date	order_qty	sales_amount	currency
Prod001	Cus001	Mar1001	22-07-2	1	100	EUR
Prod001	Cus002	Mar1002	08-09-2	1	100	EUR
Prod002	Cus003	Mar1003	08-04-2	1	100	EUR
Prod002	Cus004	Mar1004	17-04-2	1	100	EUR
Prod002	Cus005	Mar1005	18-06-2	1	100	EUR
Prod003	Cus006	Mar1006	08-11-2	1	100	EUR
Prod003	Cus007	Mar1007	22-11-2	1	100	EUR
Prod003	Cus008	Mar1008	25-11-2	1	100	EUR
Prod003	Cus009	Mar1009	27-11-2	1	100	EUR
Prod004	Cus010	Mar1010	28-11-2	1	100	EUR
Prod004	Cus011	Mar1011	29-11-2	1	100	EUR
Prod004	Cus012	Mar1012	29-11-2	1	100	EUR
Prod004	Cus013	Mar1013	29-11-2	1	100	EUR
Prod004	Cus014	Mar1014	29-11-2	1	100	EUR
Prod004	Cus015	Mar1015	29-11-2	1	100	EUR
Prod004	Cus016	Mar1016	29-11-2	1	100	EUR
Prod004	Cus017	Mar1017	29-11-2	1	100	EUR
Prod004	Cus018	Mar1018	29-11-2	1	100	EUR
Prod004	Cus019	Mar1019	29-11-2	1	100	EUR
Prod004	Cus020	Mar1020	29-11-2	1	100	EUR
Prod004	Cus021	Mar1021	29-11-2	1	100	EUR
Prod004	Cus022	Mar1022	29-11-2	1	100	EUR
Prod004	Cus023	Mar1023	29-11-2	1	100	EUR
Prod004	Cus024	Mar1024	29-11-2	1	100	EUR
Prod004	Cus025	Mar1025	29-11-2	1	100	EUR
Prod004	Cus026	Mar1026	29-11-2	1	100	EUR
Prod004	Cus027	Mar1027	29-11-2	1	100	EUR
Prod004	Cus028	Mar1028	29-11-2	1	100	EUR
Prod004	Cus029	Mar1029	29-11-2	1	100	EUR
Prod004	Cus030	Mar1030	29-11-2	1	100	EUR

### Step 9 : Make Changes Into Order Date In Transaction Table.

The screenshot displays the Microsoft Excel interface with a PivotTable. The PivotTable is set to show data by Product Code, Customer Code, and Market Code, with Order Date as the column label. The data is organized into a grid with 10 columns and 20 rows. The right-hand pane shows the 'Fields' task pane with a search bar and a list of available fields including Customer, Date, Product, and Transaction.

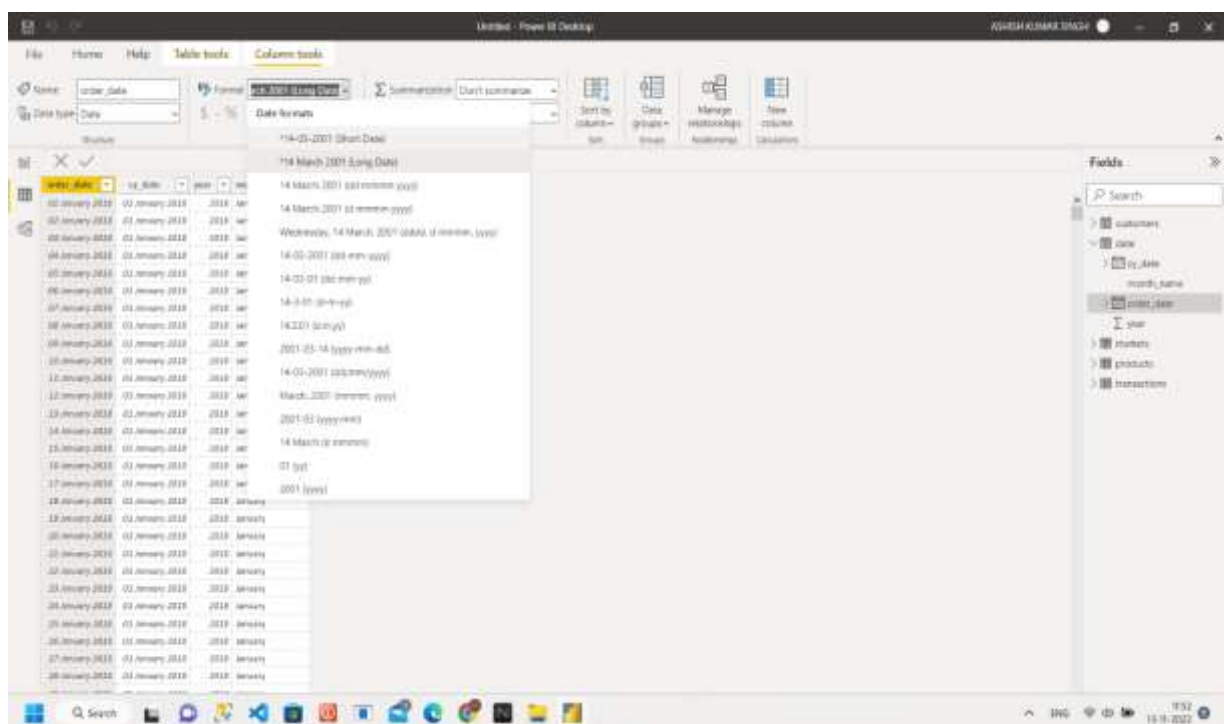
Product Code	Customer Code	Market Code	Order Date	Order Qty	Sales Amount	Revenue
Prod078	Cus001	Mar011	11-10-2027	1	101.994	
Prod078	Cus001	Mar011	18-10-2027	1	101.994	
Prod078	Cus001	Mar011	25-10-2027	1	101.994	
Prod078	Cus001	Mar011	08-12-2027	1	101.994	
Prod078	Cus001	Mar011	09-01-2028	1	101.994	
Prod078	Cus001	Mar011	16-01-2028	1	101.994	
Prod078	Cus001	Mar011	23-01-2028	1	101.994	
Prod078	Cus001	Mar011	29-01-2028	1	101.994	
Prod078	Cus001	Mar011	05-02-2028	1	101.994	
Prod078	Cus001	Mar011	12-02-2028	1	101.994	
Prod078	Cus001	Mar011	19-02-2028	1	101.994	
Prod078	Cus001	Mar011	26-02-2028	1	101.994	
Prod078	Cus001	Mar011	05-03-2028	1	101.994	
Prod078	Cus001	Mar011	12-03-2028	1	101.994	
Prod078	Cus001	Mar011	19-03-2028	1	101.994	
Prod078	Cus001	Mar011	26-03-2028	1	101.994	
Prod078	Cus001	Mar011	02-04-2028	1	101.994	
Prod078	Cus001	Mar011	09-04-2028	1	101.994	
Prod078	Cus001	Mar011	16-04-2028	1	101.994	
Prod078	Cus001	Mar011	23-04-2028	1	101.994	
Prod078	Cus001	Mar011	30-04-2028	1	101.994	
Prod078	Cus001	Mar011	07-05-2028	1	101.994	
Prod078	Cus001	Mar011	14-05-2028	1	101.994	
Prod078	Cus001	Mar011	21-05-2028	1	101.994	
Prod078	Cus001	Mar011	28-05-2028	1	101.994	
Prod078	Cus001	Mar011	04-06-2028	1	101.994	
Prod078	Cus001	Mar011	11-06-2028	1	101.994	
Prod078	Cus001	Mar011	18-06-2028	1	101.994	
Prod078	Cus001	Mar011	25-06-2028	1	101.994	
Prod078	Cus001	Mar011	02-07-2028	1	101.994	
Prod078	Cus001	Mar011	09-07-2028	1	101.994	
Prod078	Cus001	Mar011	16-07-2028	1	101.994	
Prod078	Cus001	Mar011	23-07-2028	1	101.994	
Prod078	Cus001	Mar011	30-07-2028	1	101.994	
Prod078	Cus001	Mar011	06-08-2028	1	101.994	
Prod078	Cus001	Mar011	13-08-2028	1	101.994	
Prod078	Cus001	Mar011	20-08-2028	1	101.994	
Prod078	Cus001	Mar011	27-08-2028	1	101.994	
Prod078	Cus001	Mar011	03-09-2028	1	101.994	
Prod078	Cus001	Mar011	10-09-2028	1	101.994	
Prod078	Cus001	Mar011	17-09-2028	1	101.994	
Prod078	Cus001	Mar011	24-09-2028	1	101.994	
Prod078	Cus001	Mar011	01-10-2028	1	101.994	

## Step 10 : Changes Into Date Column .



- Make change Into order date.

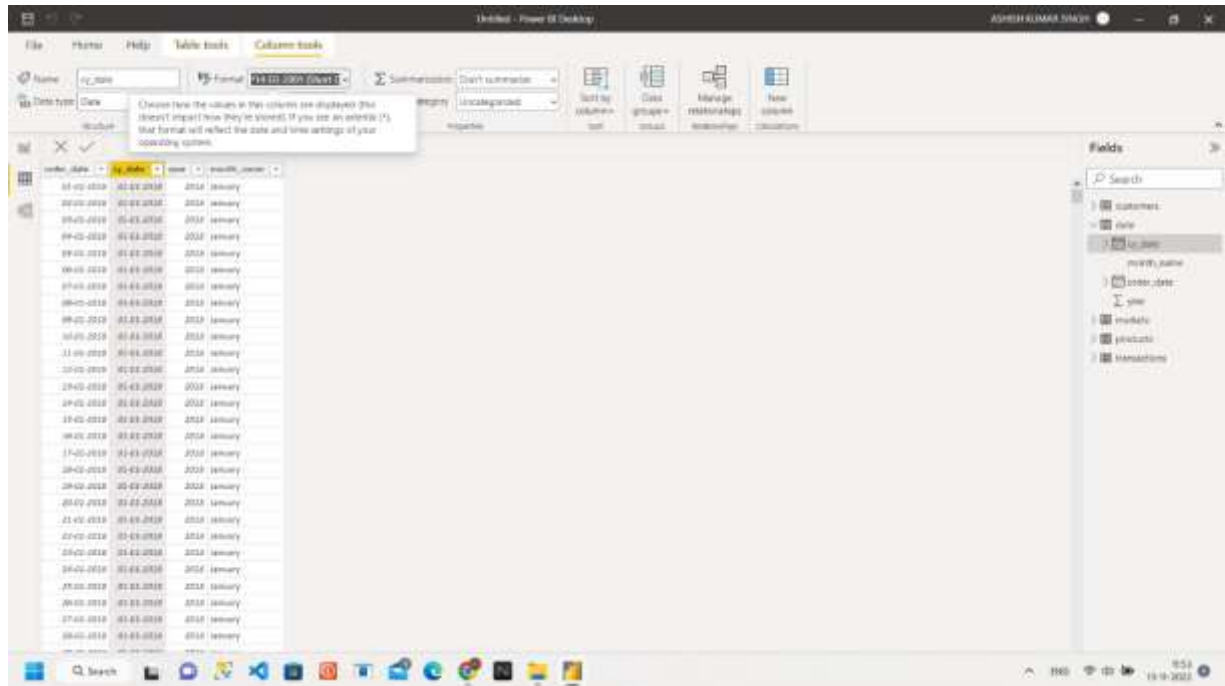
In order date column all the month date are in categorical data at first we detect this type of data.



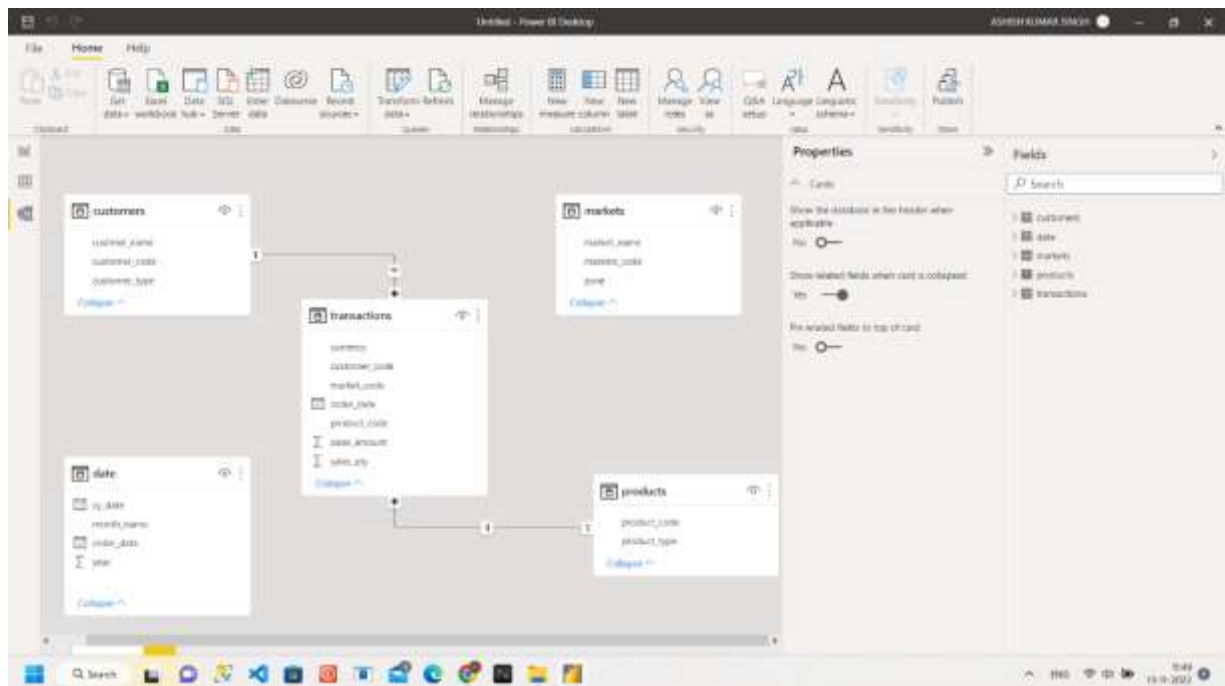


- Change Into Numerical Data.

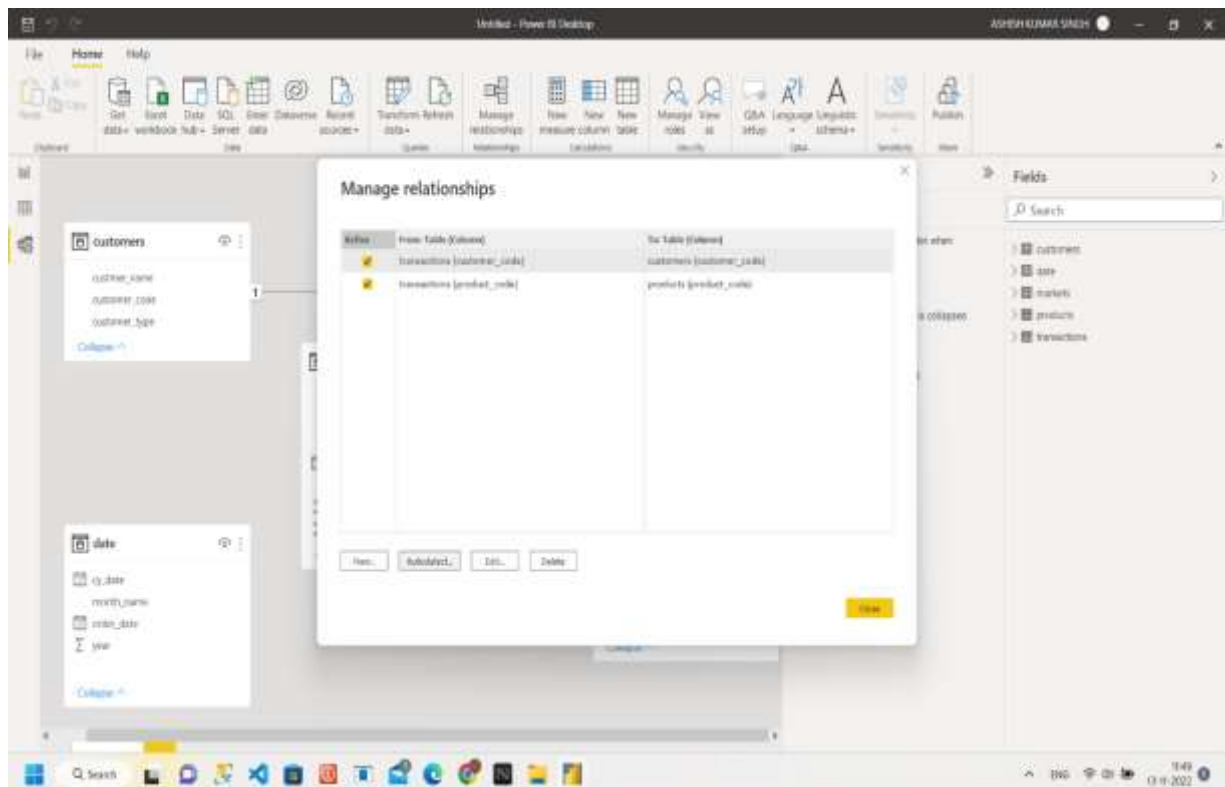
All the Month data change Into numerical data after condition statement apply.



**Step 11 : After that Check The Properties And Relationships.**

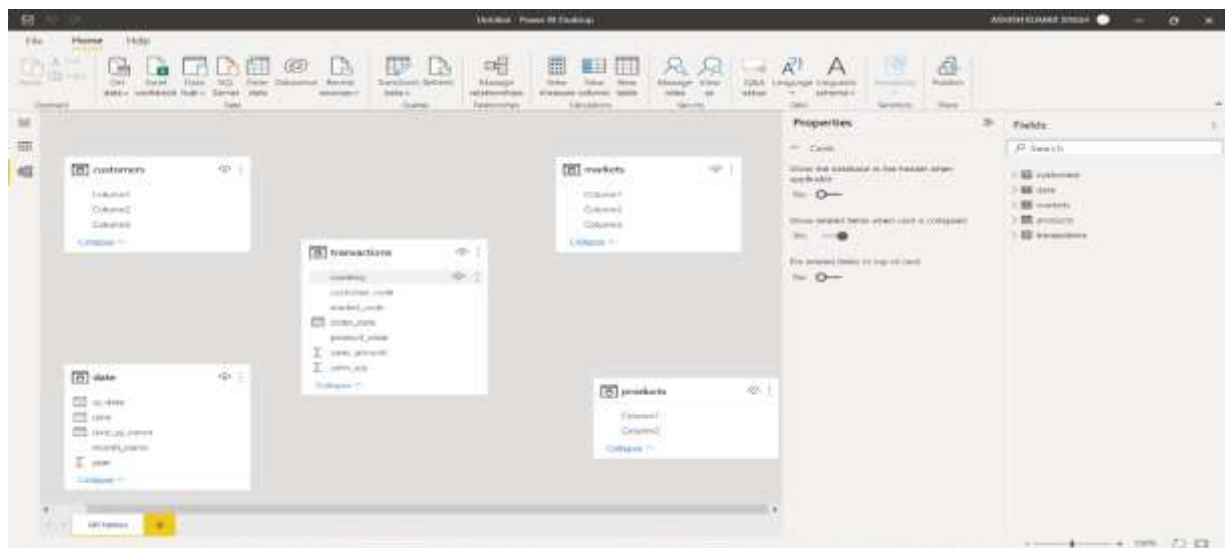


**Step 12 : Closing the window.**

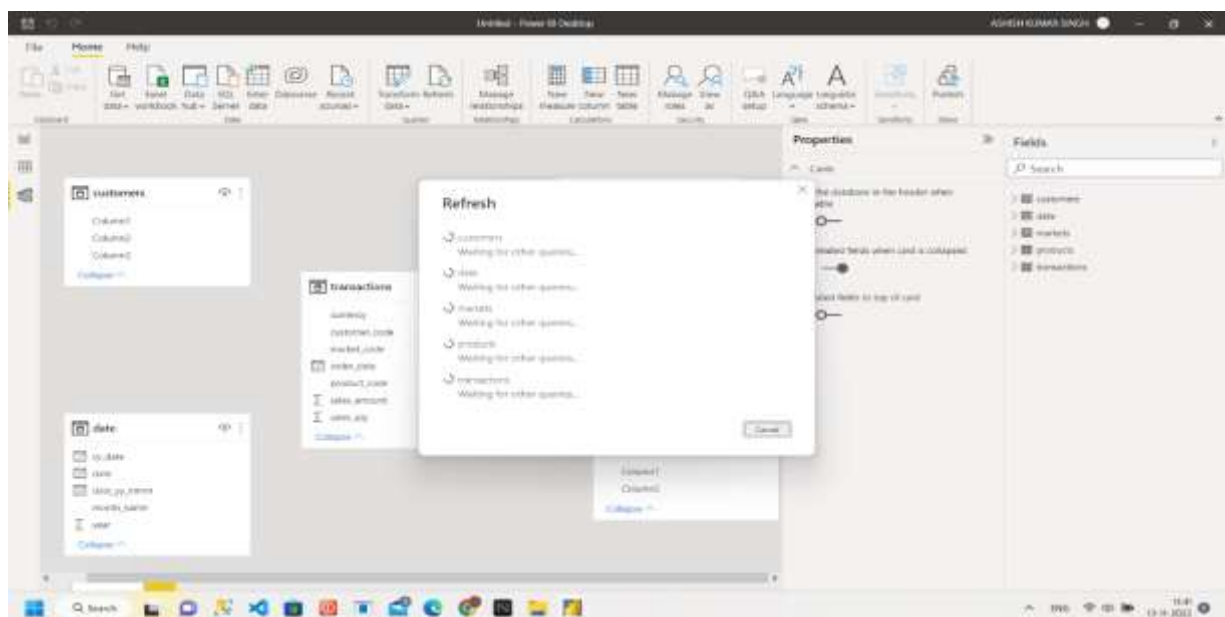


## Step 13 : Creating The Relationship Between All The Tables.

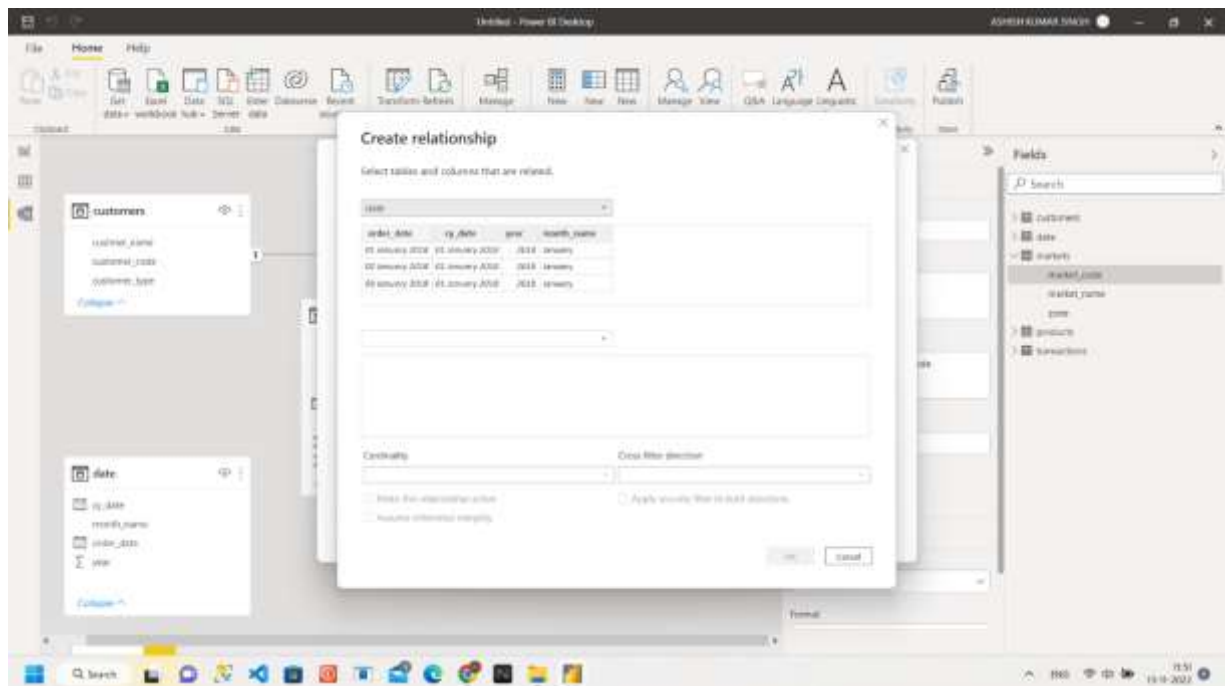
- After successfully importing all the tables and makes changes in few tables , we make a relationship between all the tables , In Power BI there is a major benefit that Power BI automatically generates a relationship between all the tables which tables are importing in Power BI.



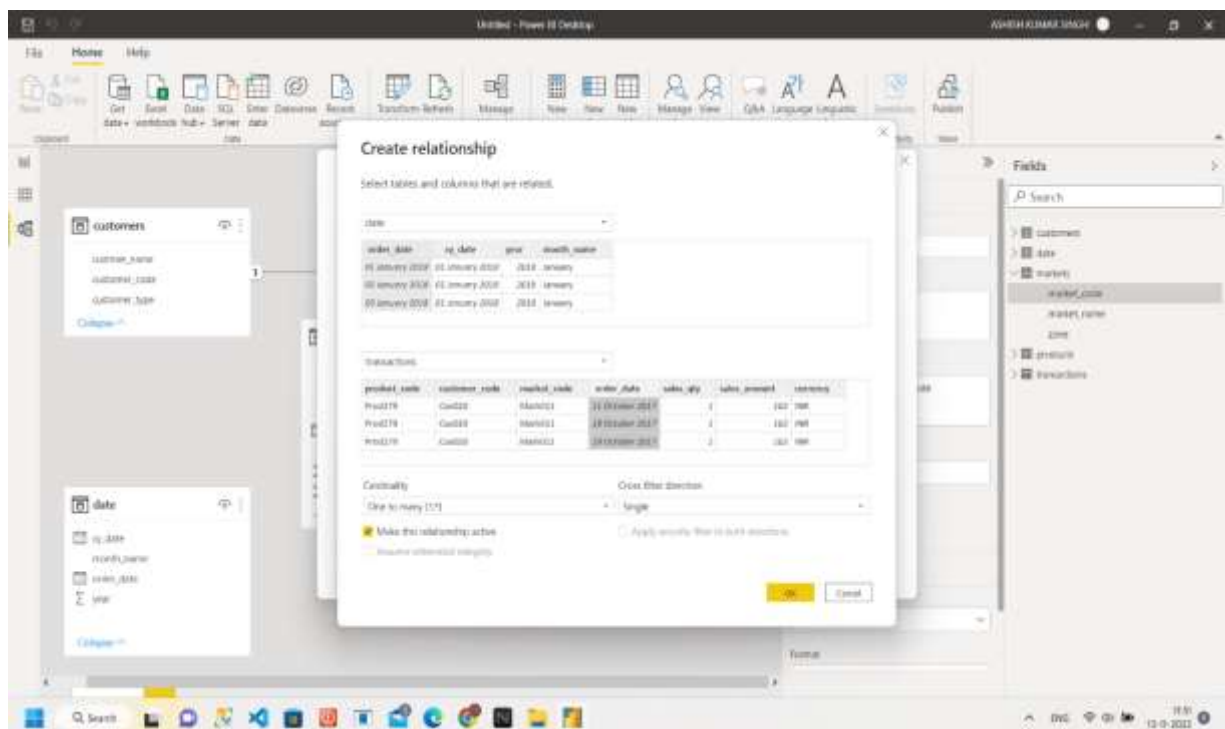
- Refreshing all the tables In power BI and all changes make visible .



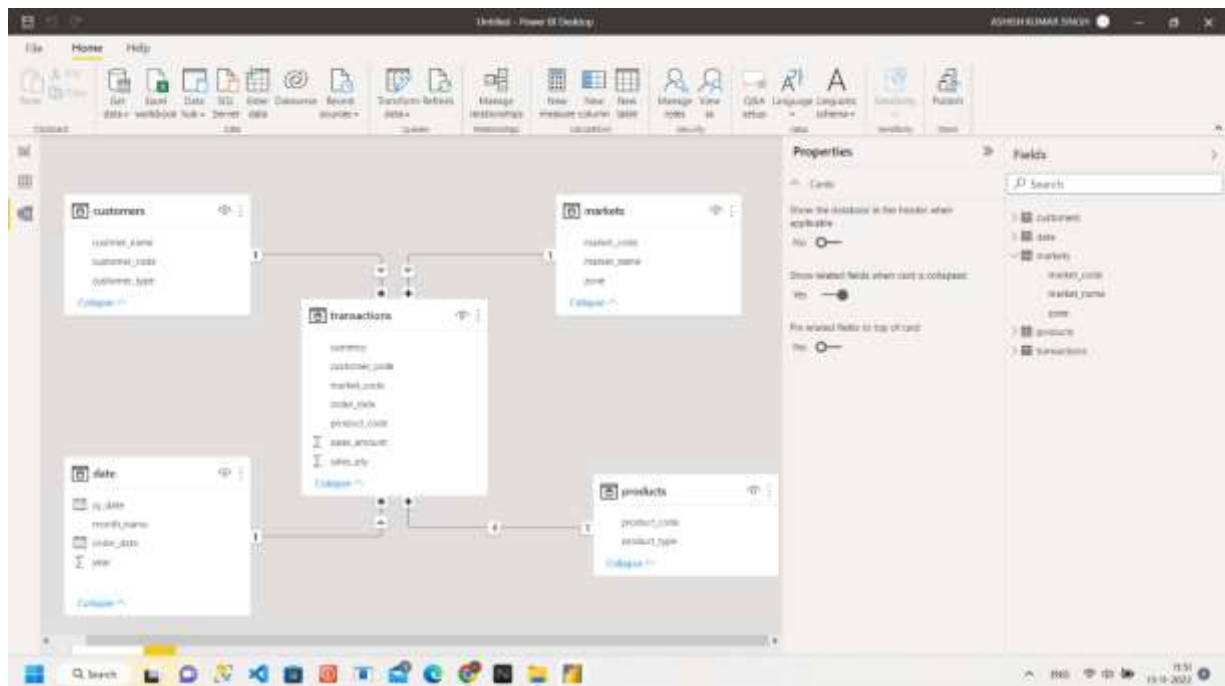
- Create Relationship with change the the column name order date in market csv file.



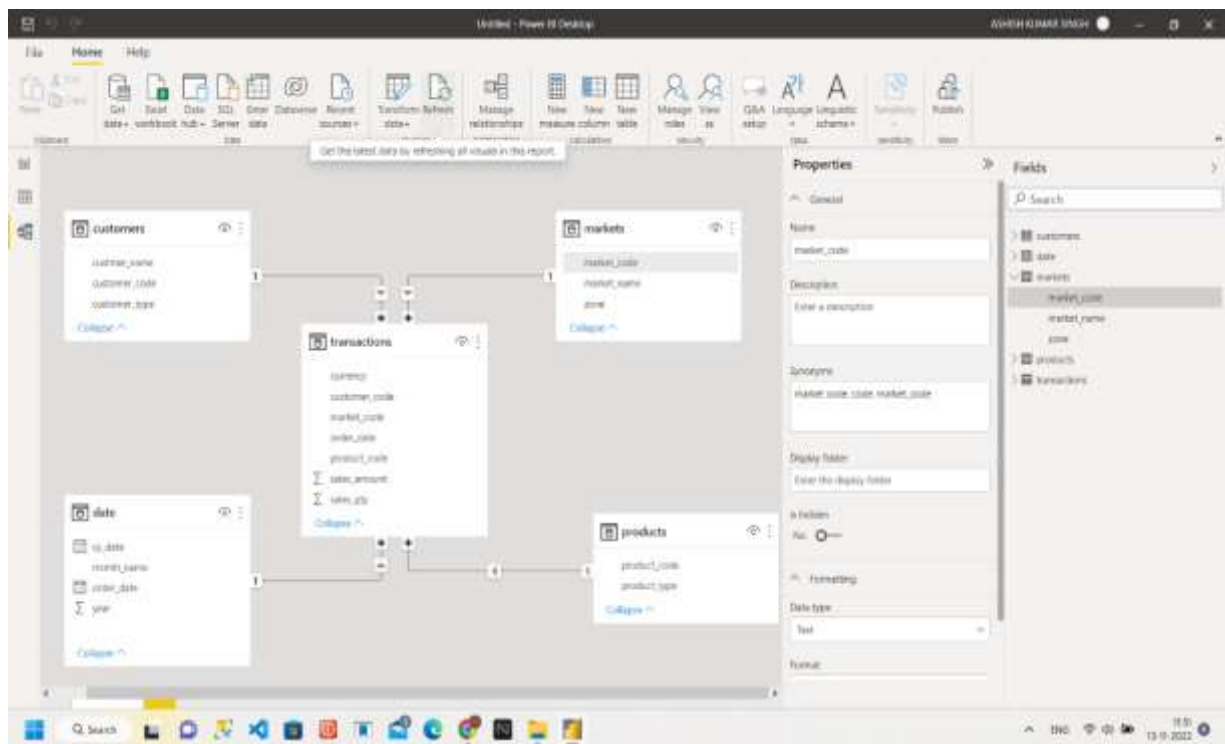
- Create Relation Between Market Table and Transaction Table.



## Step 14 : Watch The Relationship Table After Customizing All Tables.

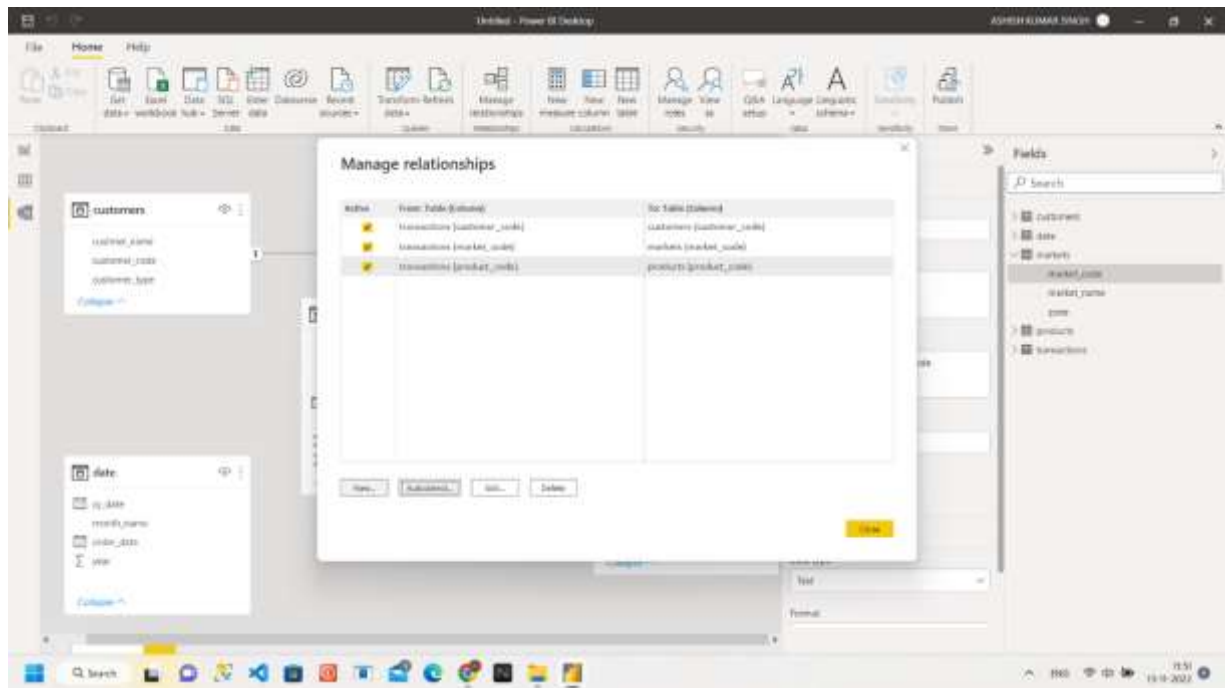


- After the we make or customize the relation between order date in Order date Table with Transaction Table.

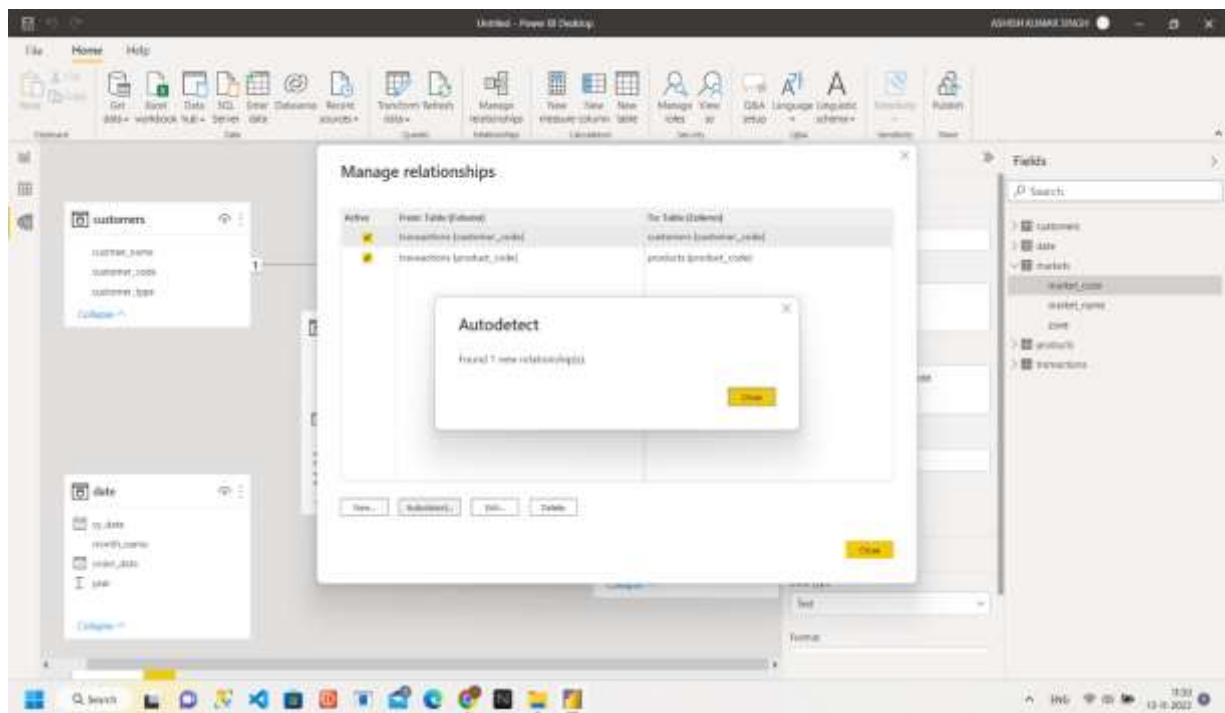


## Step 15 : Manage Relationship Of All The Tables.

- Autodetect relationship between transaction Table Product code and products product code.

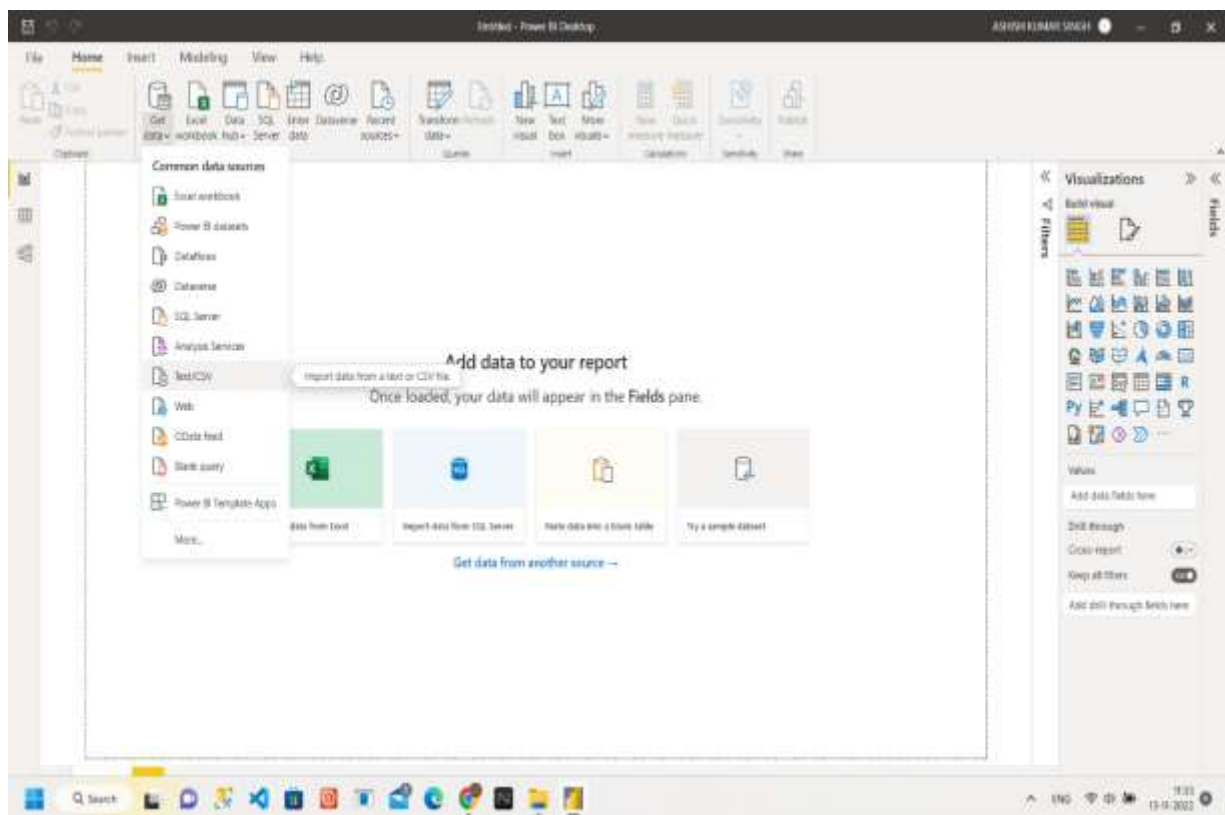


- Autodetect the new 1 relationship between Transaction table customer code and Customers table customer code.

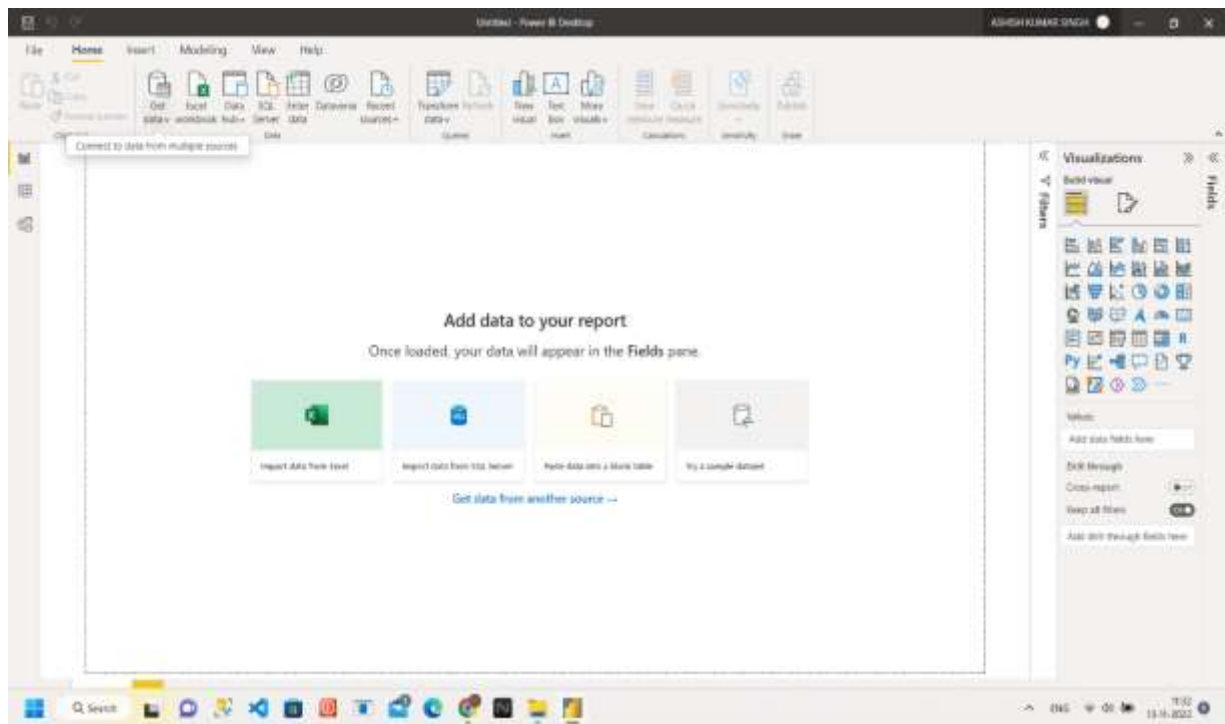


## ***PART V — Building a Dashboard or a Report***

Dashboards/reports are created according to the requirement. What actually the company wants to look for and what is more important for the company is taken into consideration and then after the dashboard is created. There can be n number of variations to create a dashboard. Generally, the dashboard should look understandable and an ease to access.



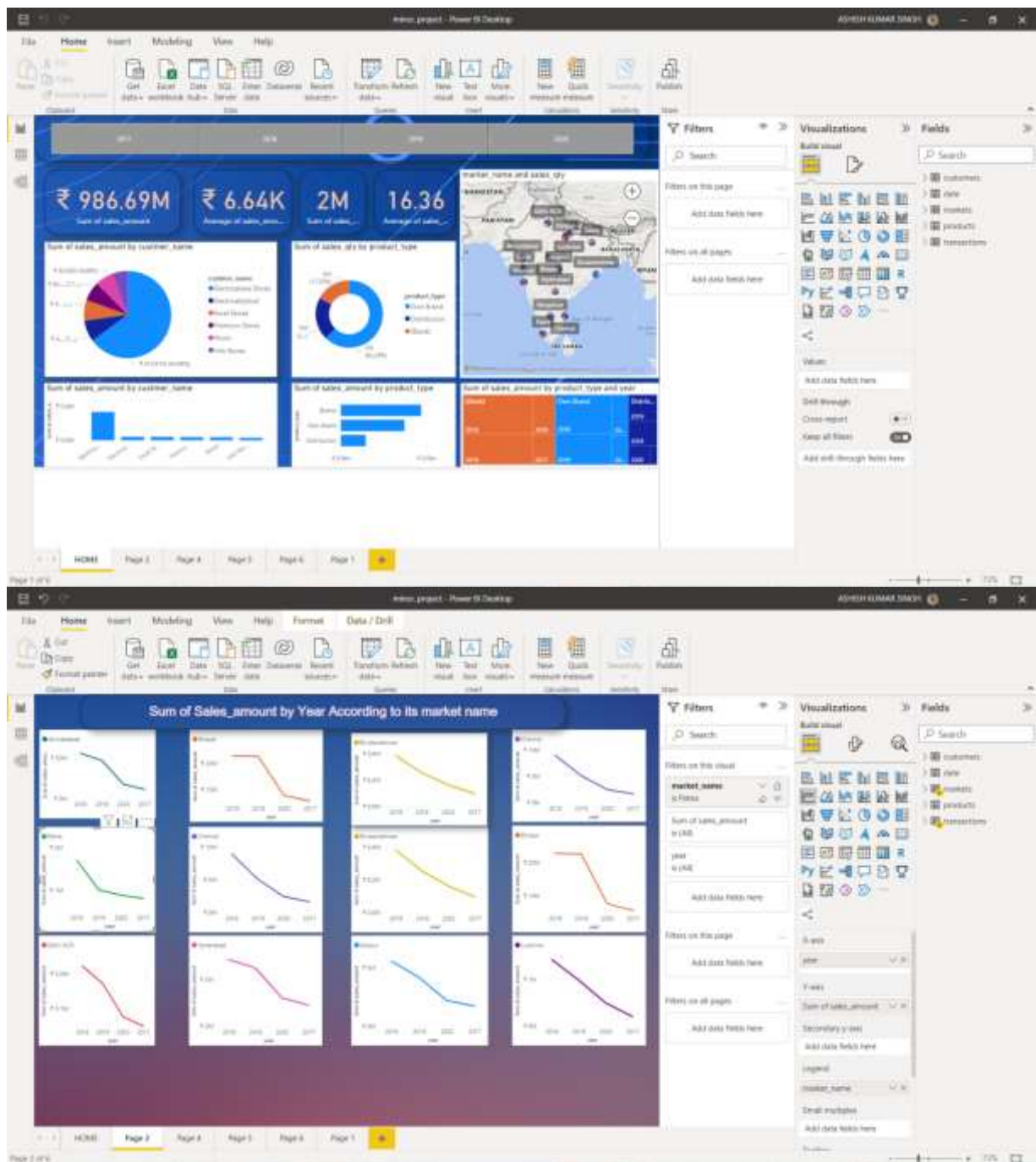
- After seeing this we see the visualizing Tool what tools we required.

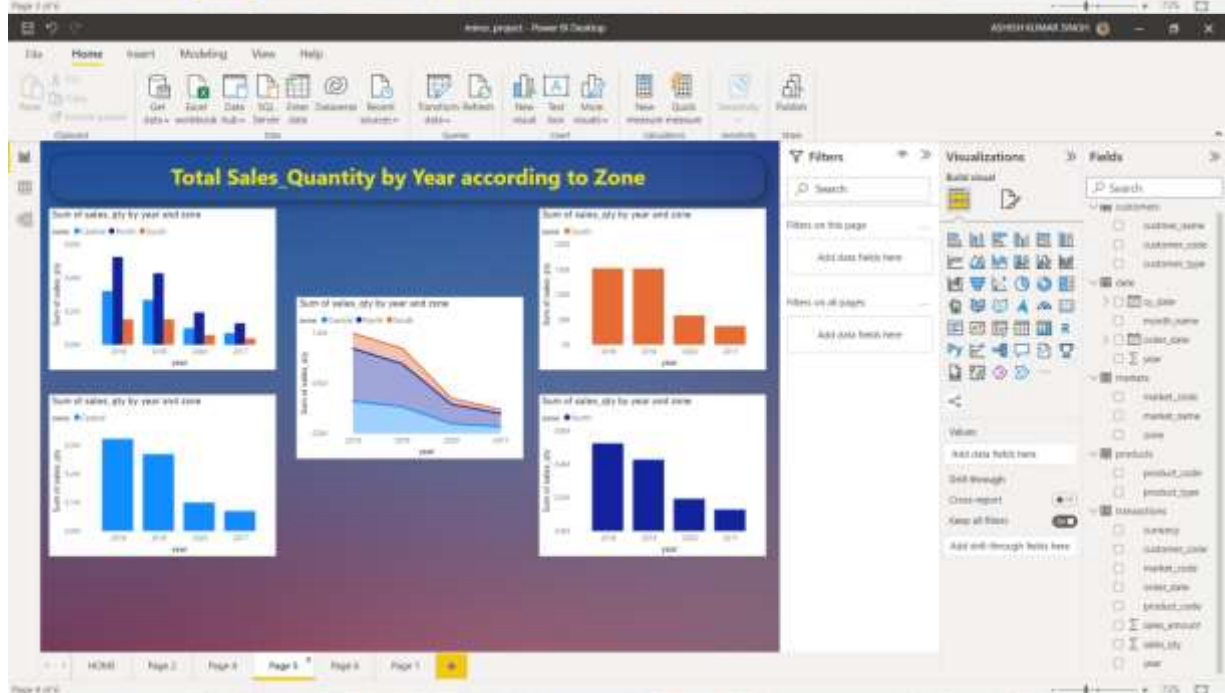
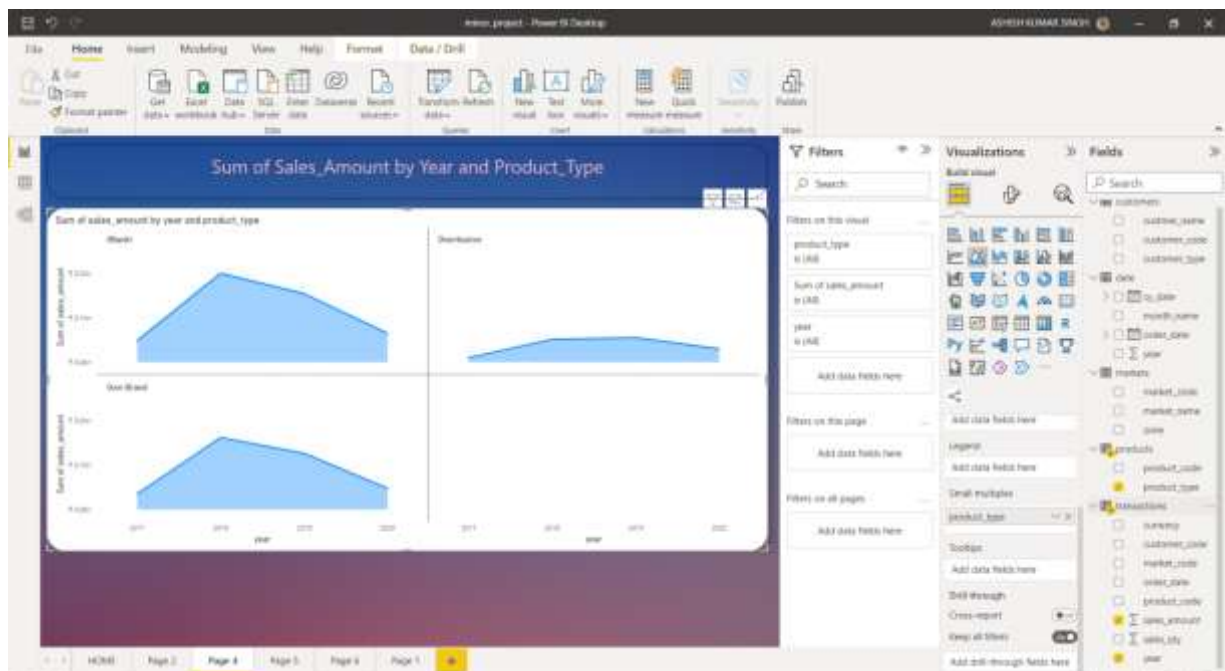


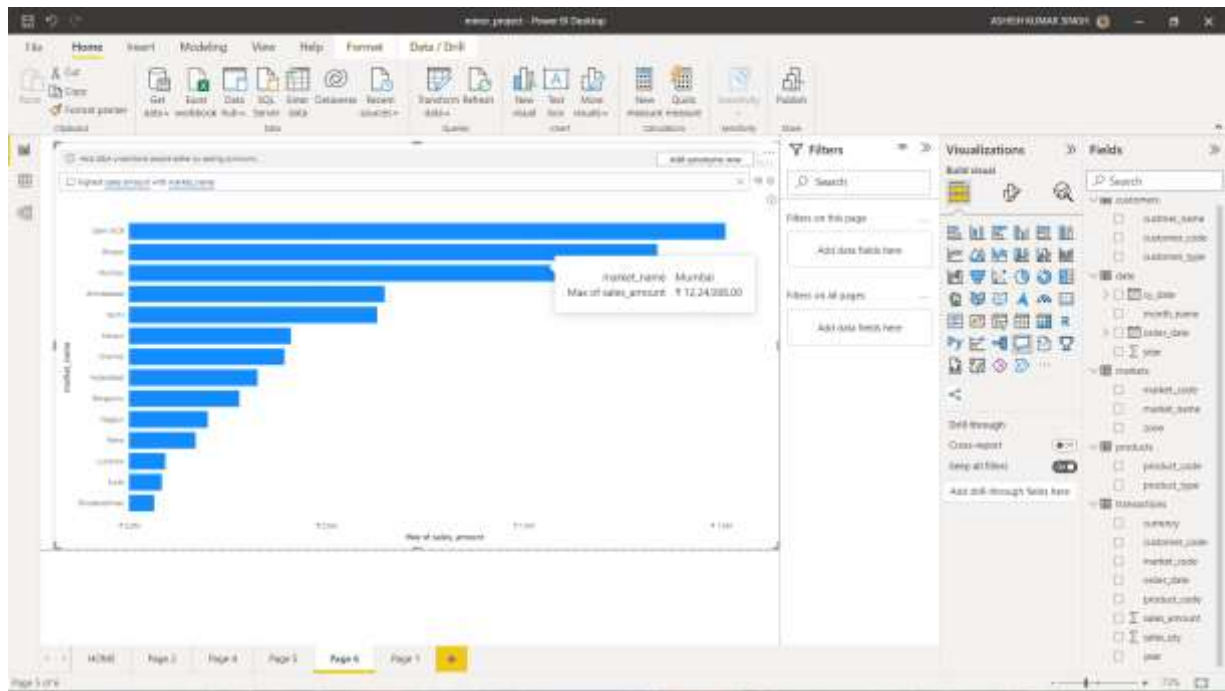


## PART VI – Publishing the Report

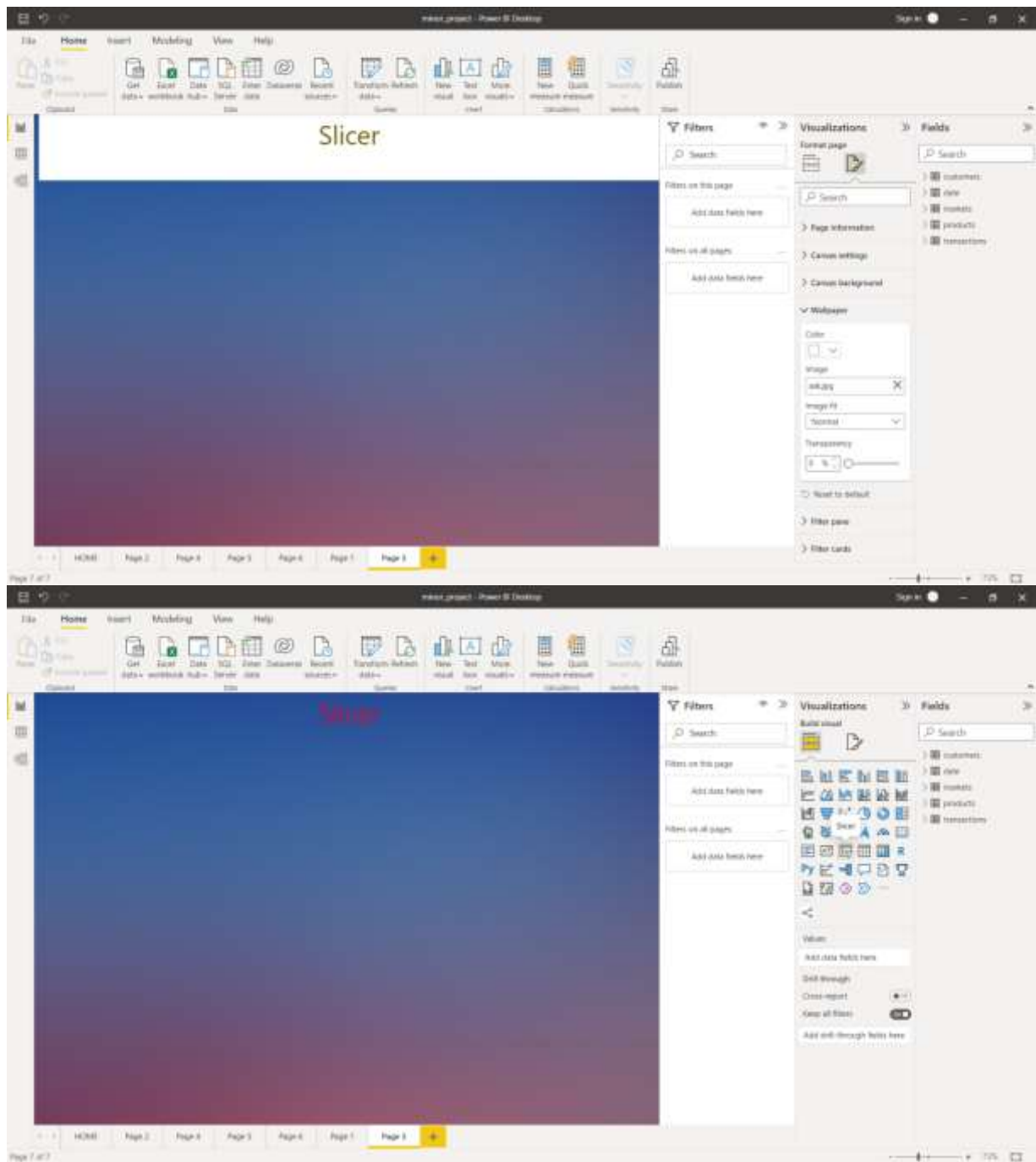
### Step 1: Publishing the report to the web version of PowerBI

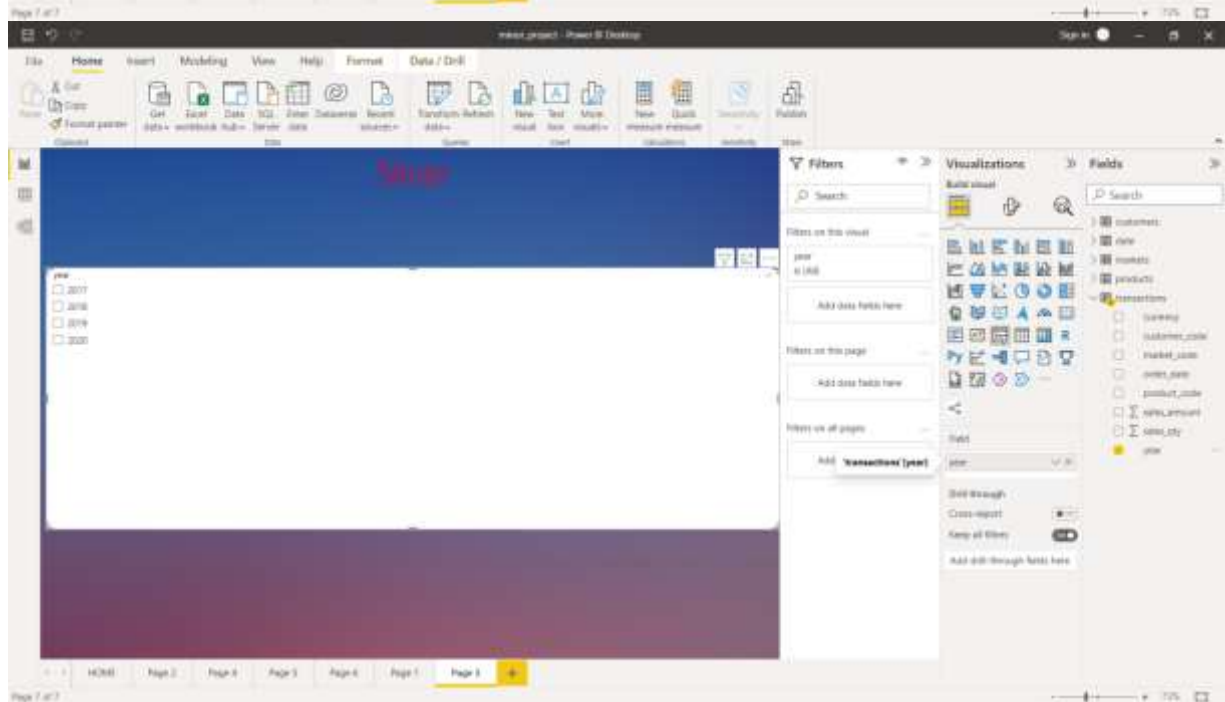
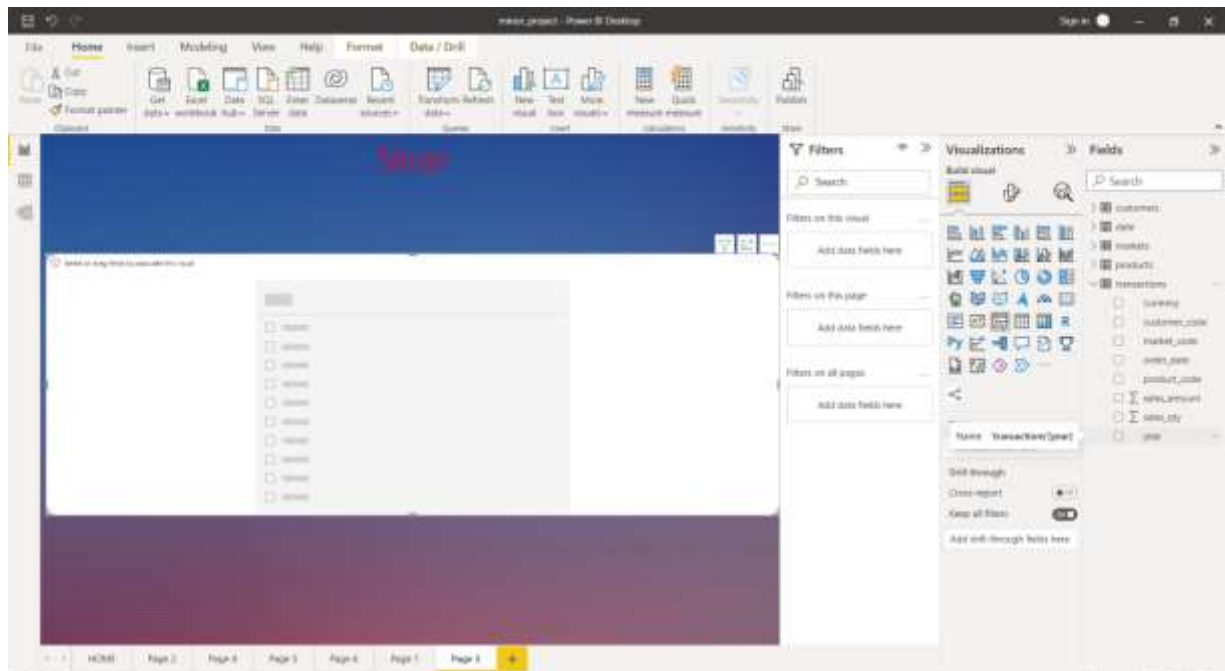


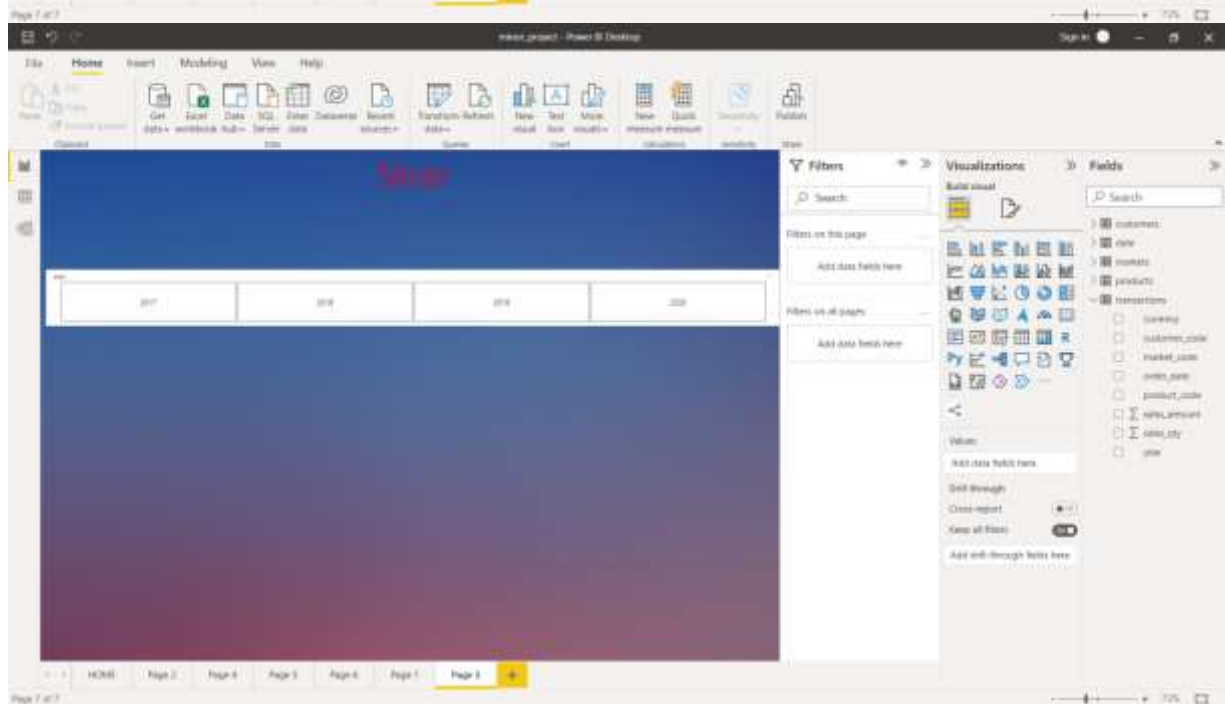
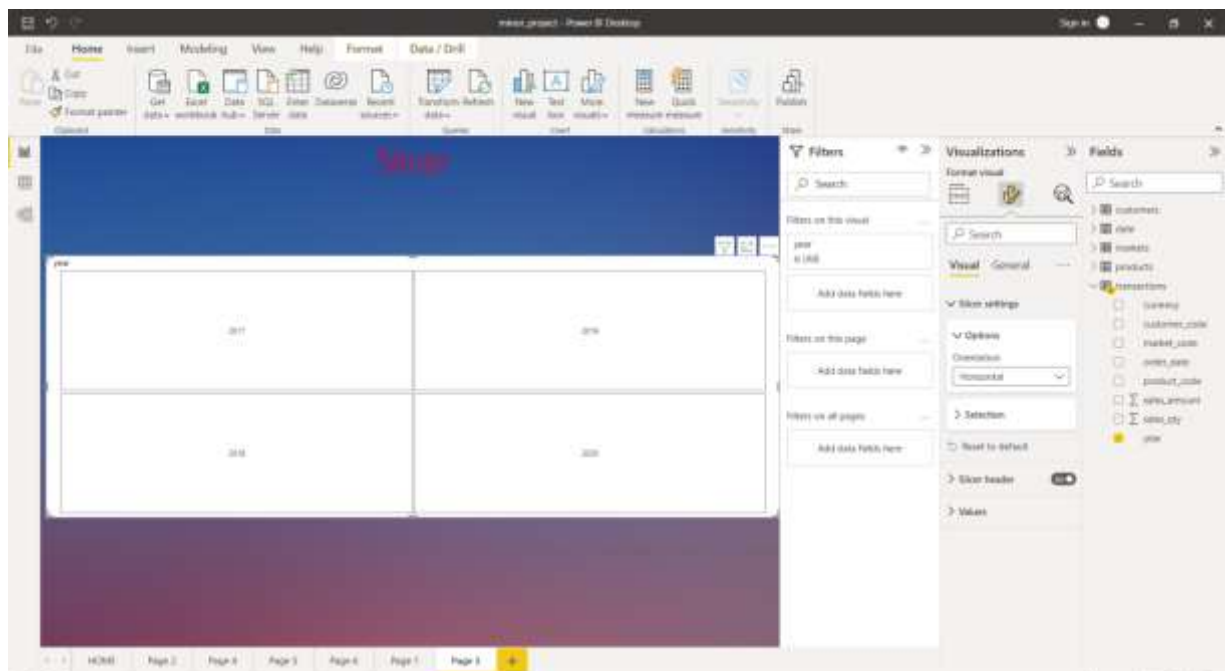




# SLICER

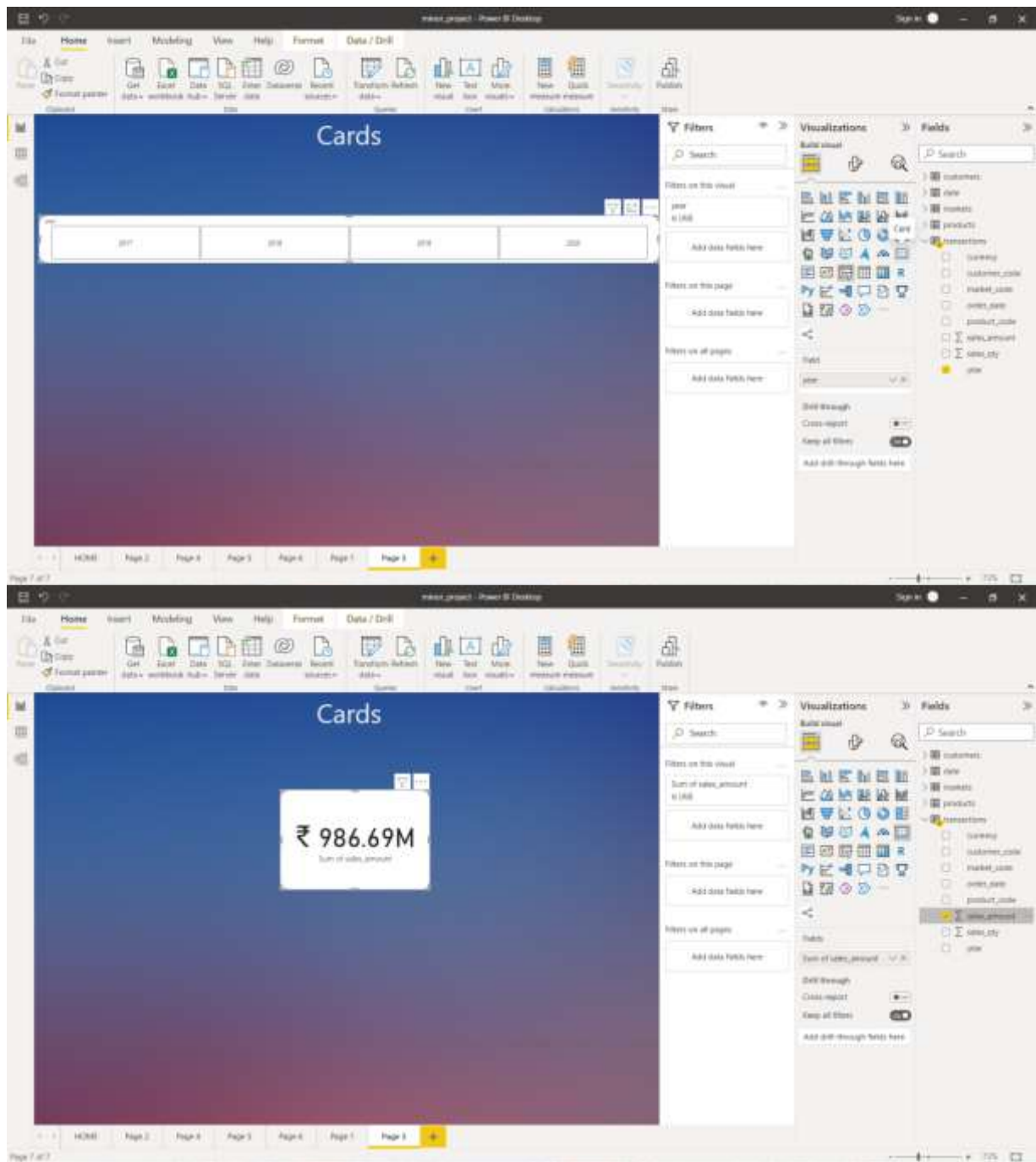




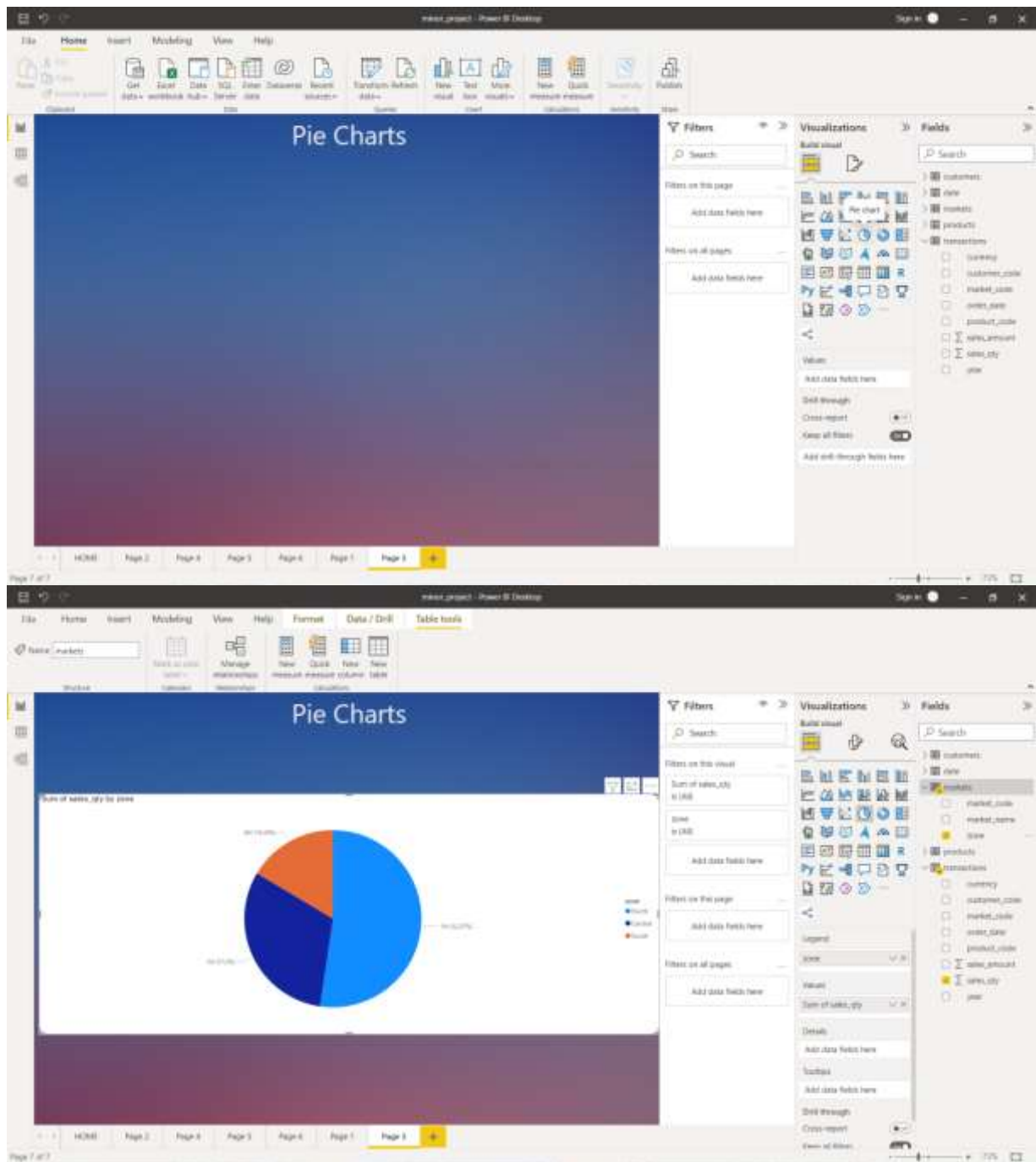




# CARD

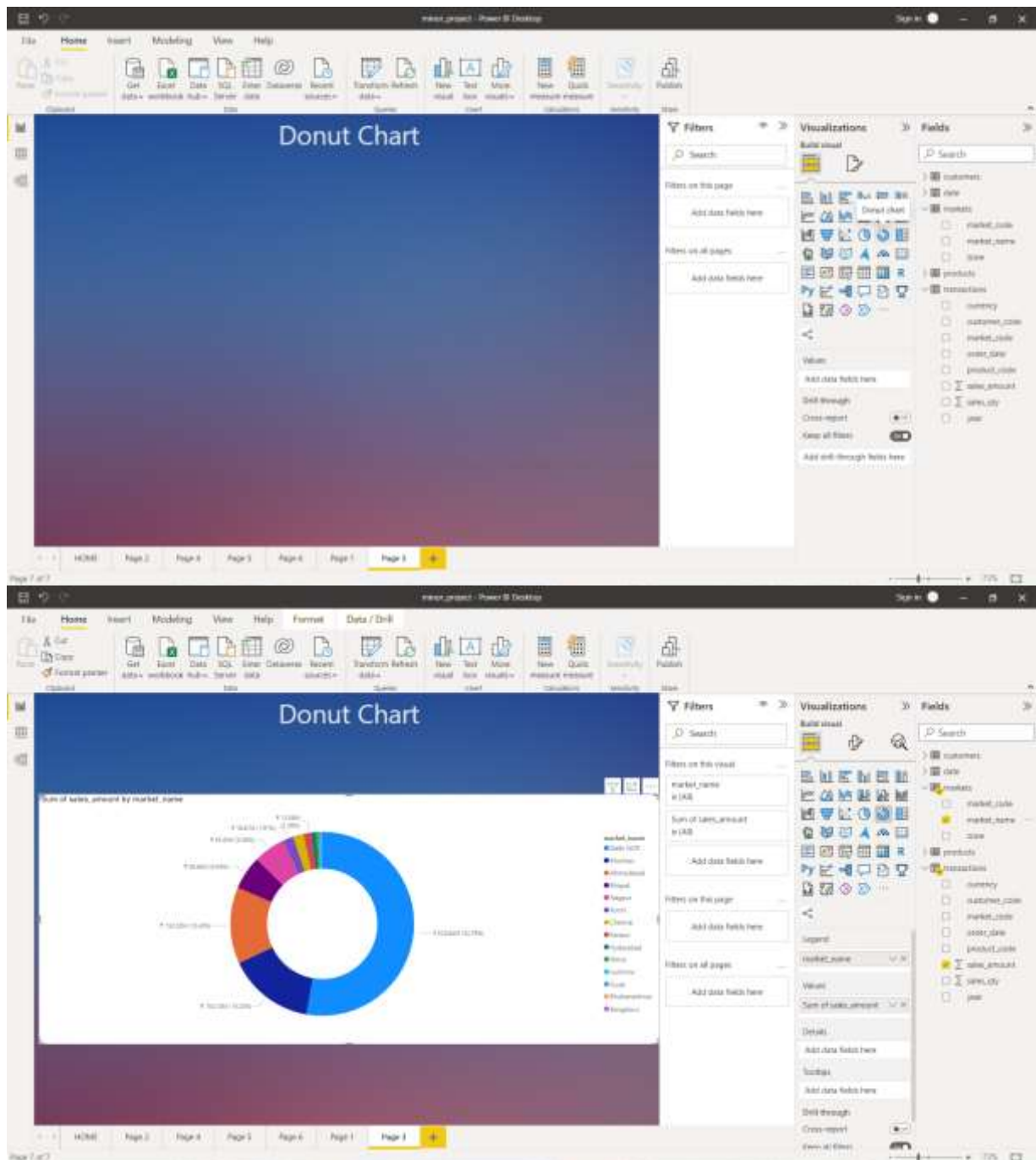


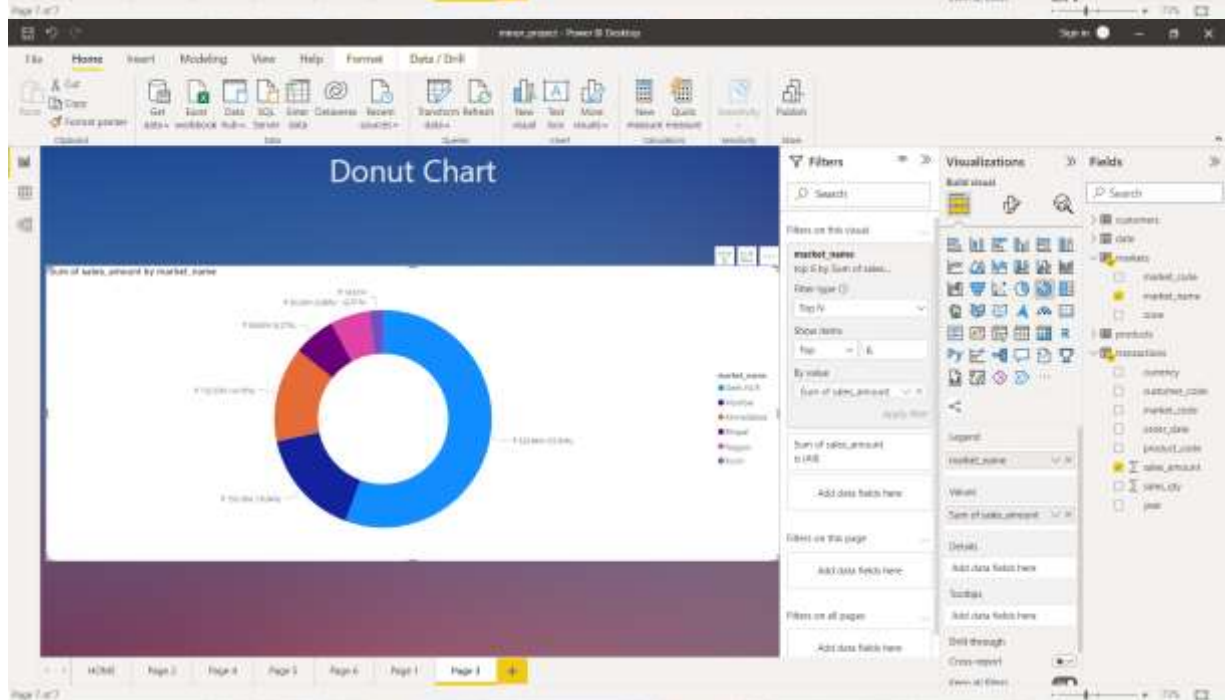
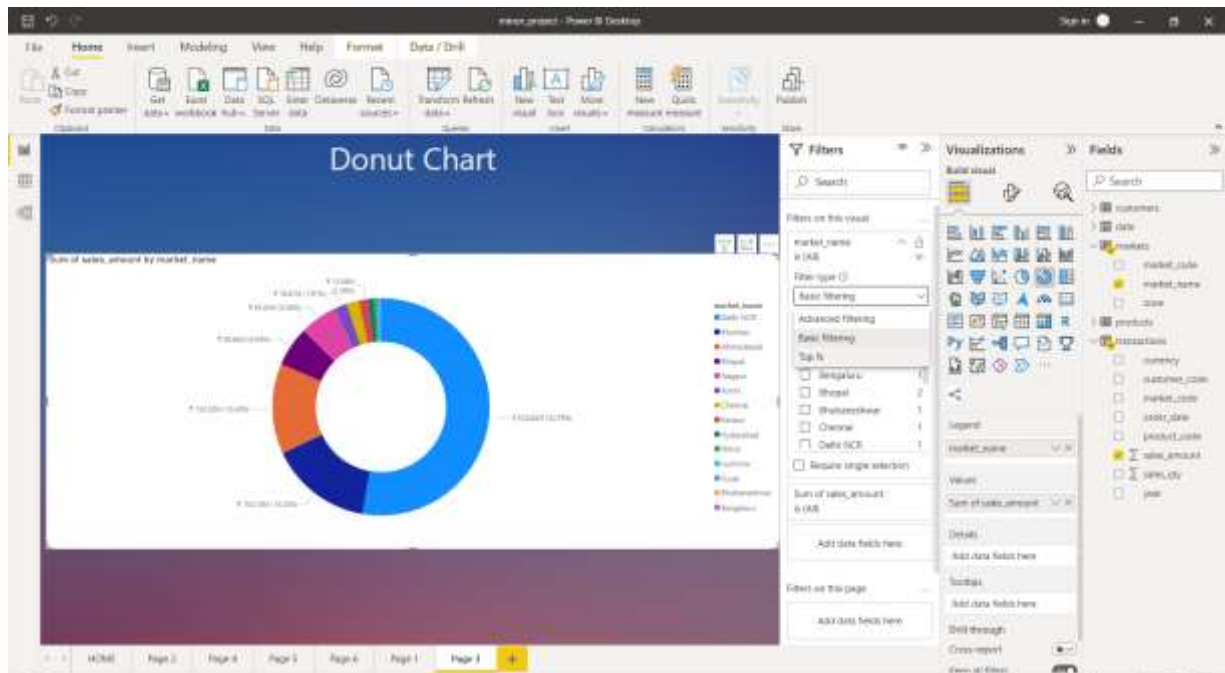
# Pie chart



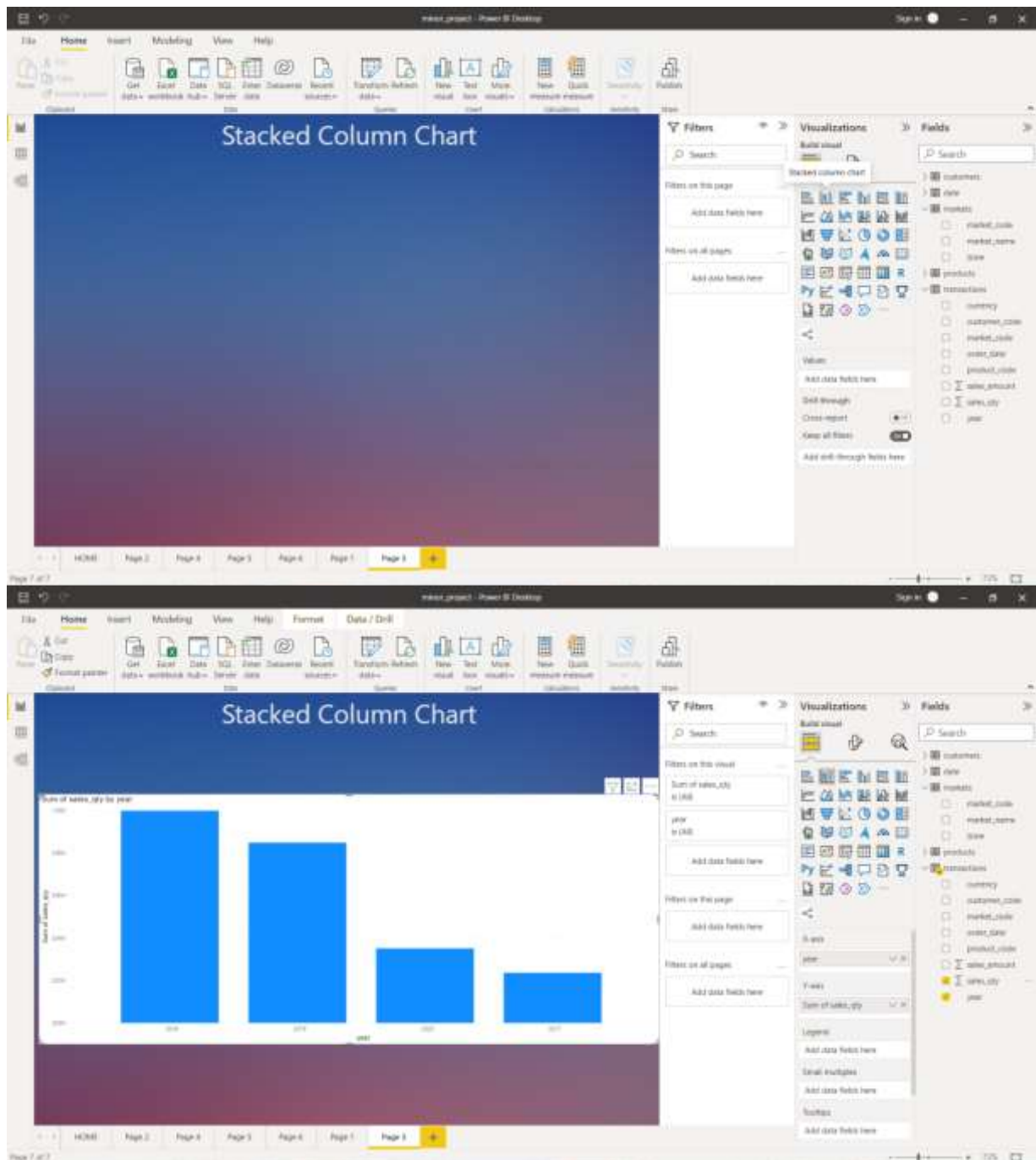


# Donut Chart

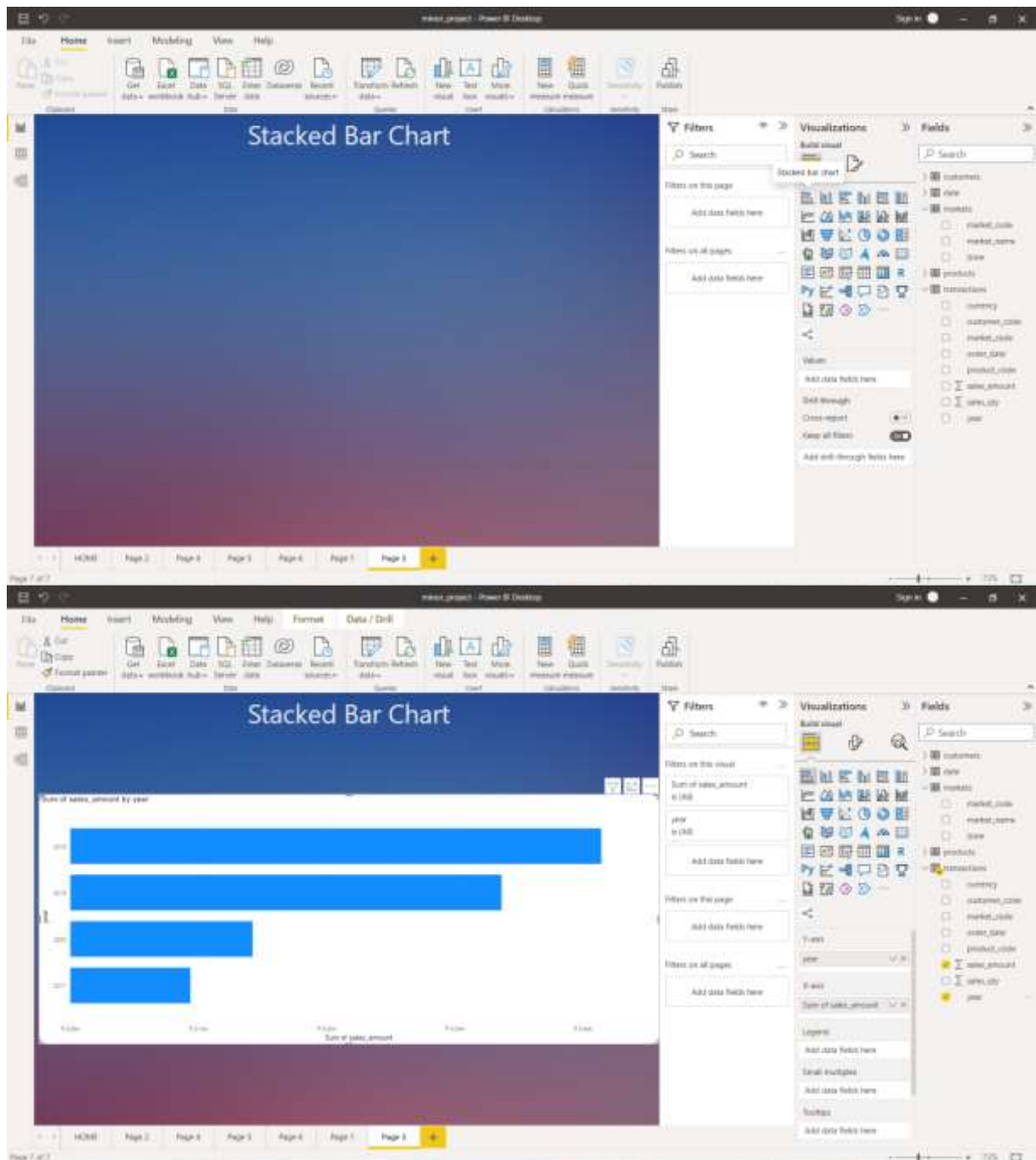




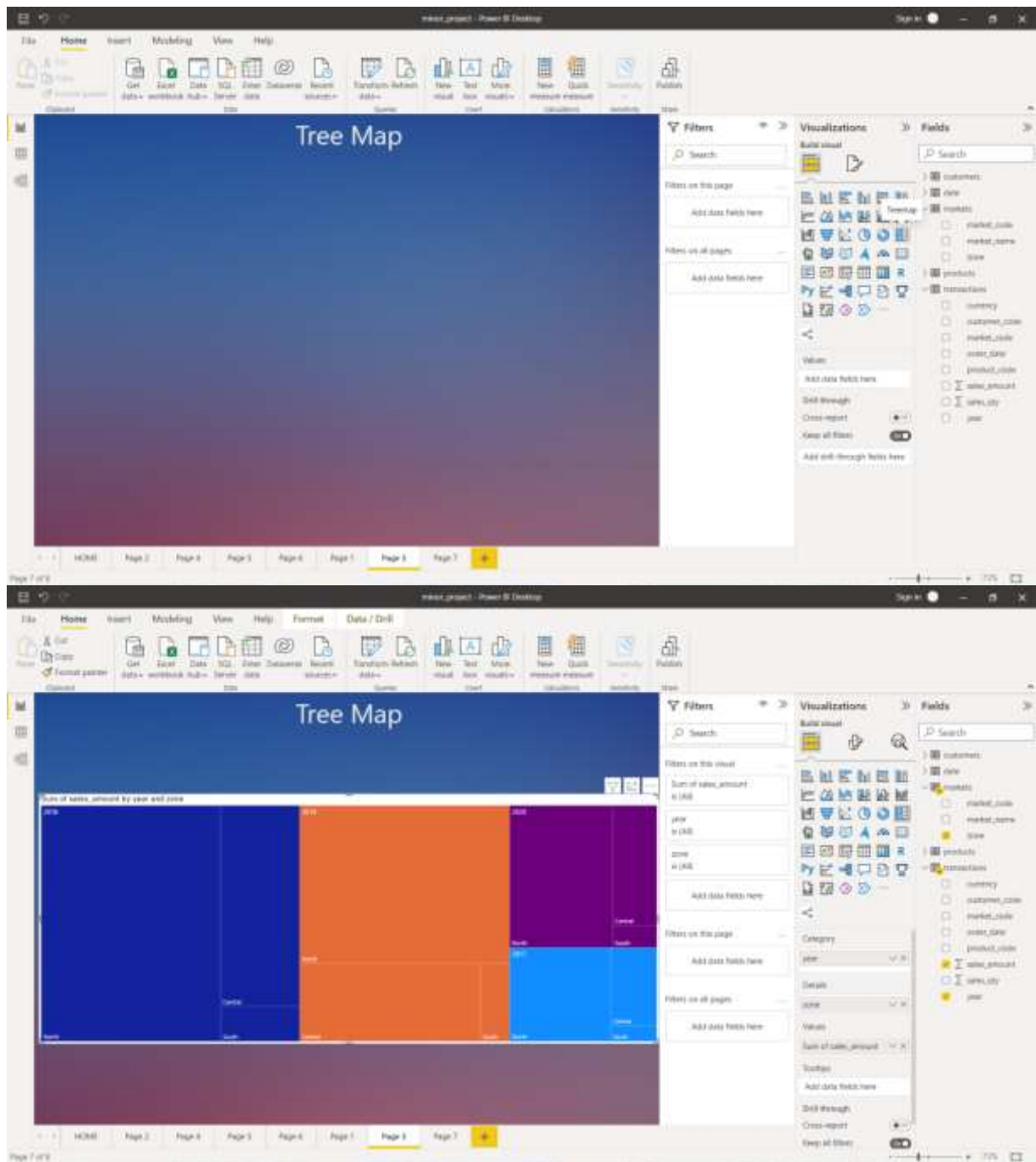
# Stacked Column Chart



# Stacked Bar Chart



# Tree Map

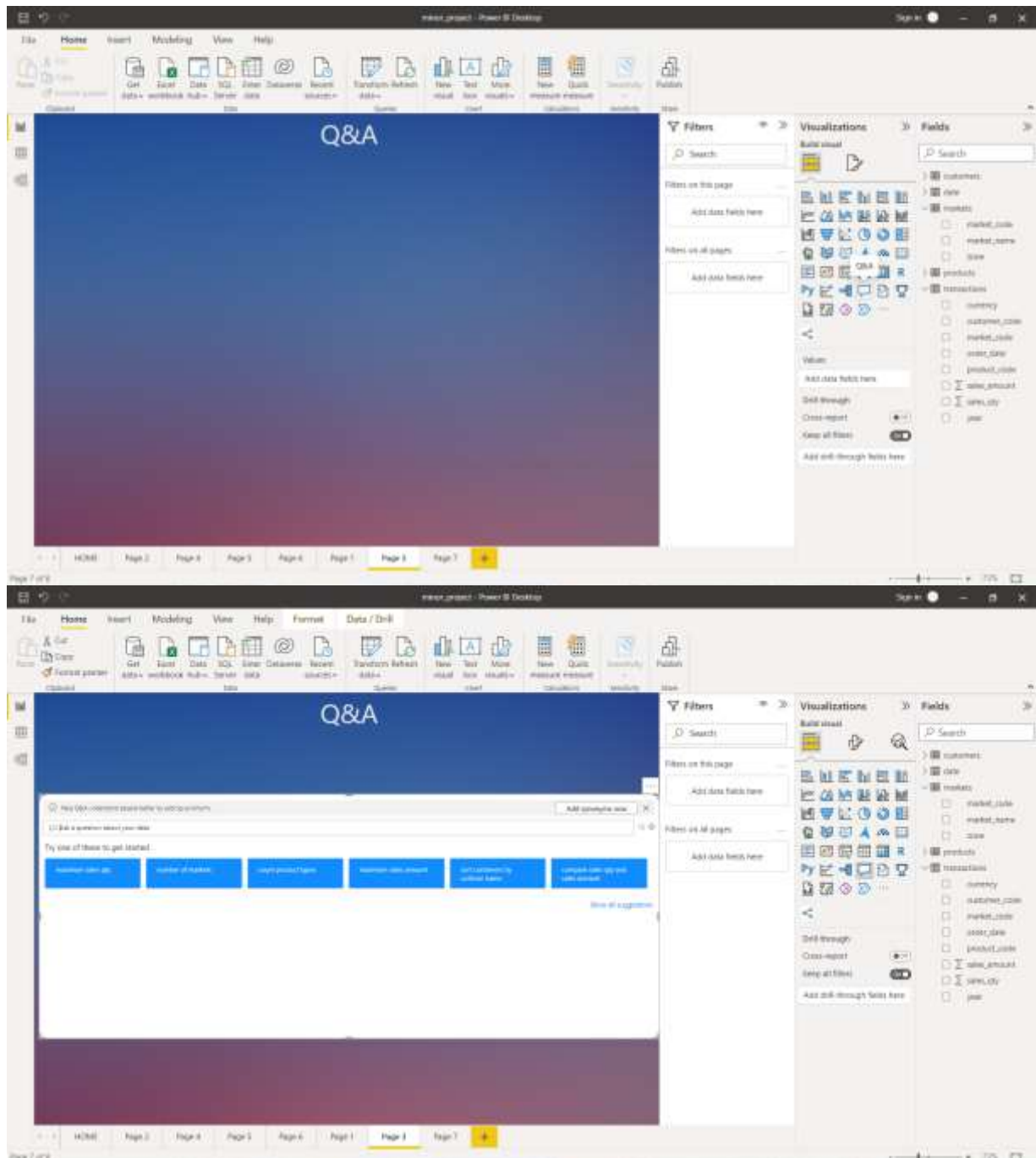




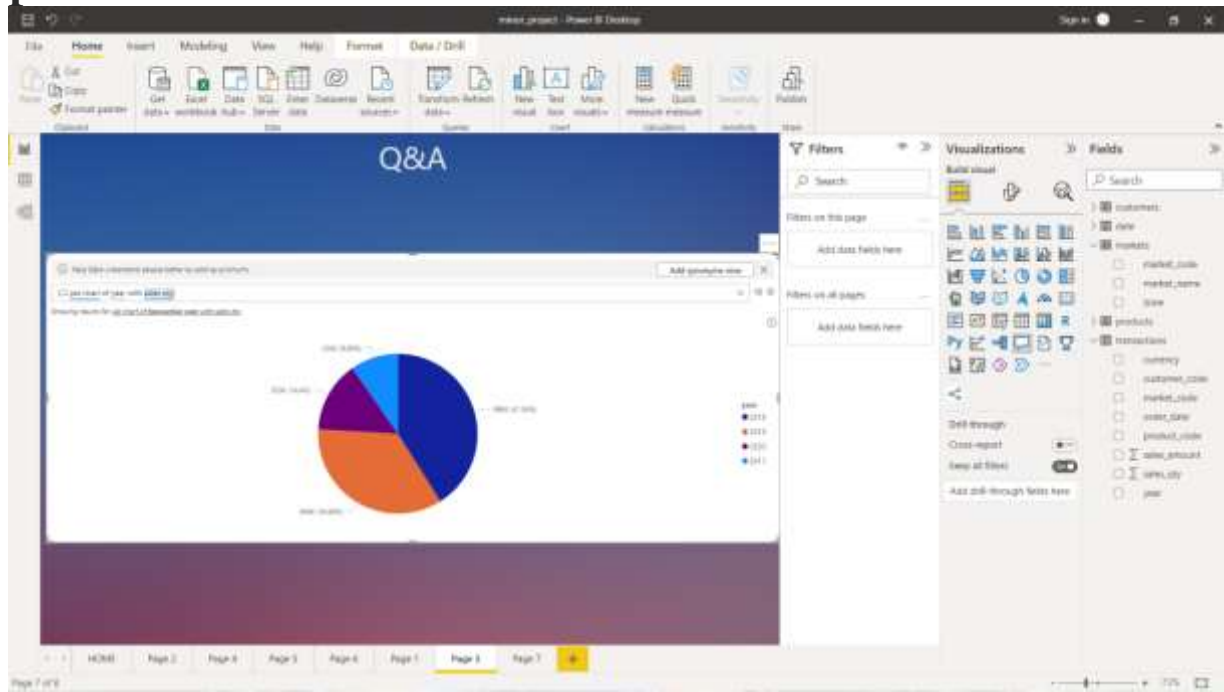
**So these are the various charts we used in making our report on Power Bi.**

**But there is one interesting visual known as Q & A**

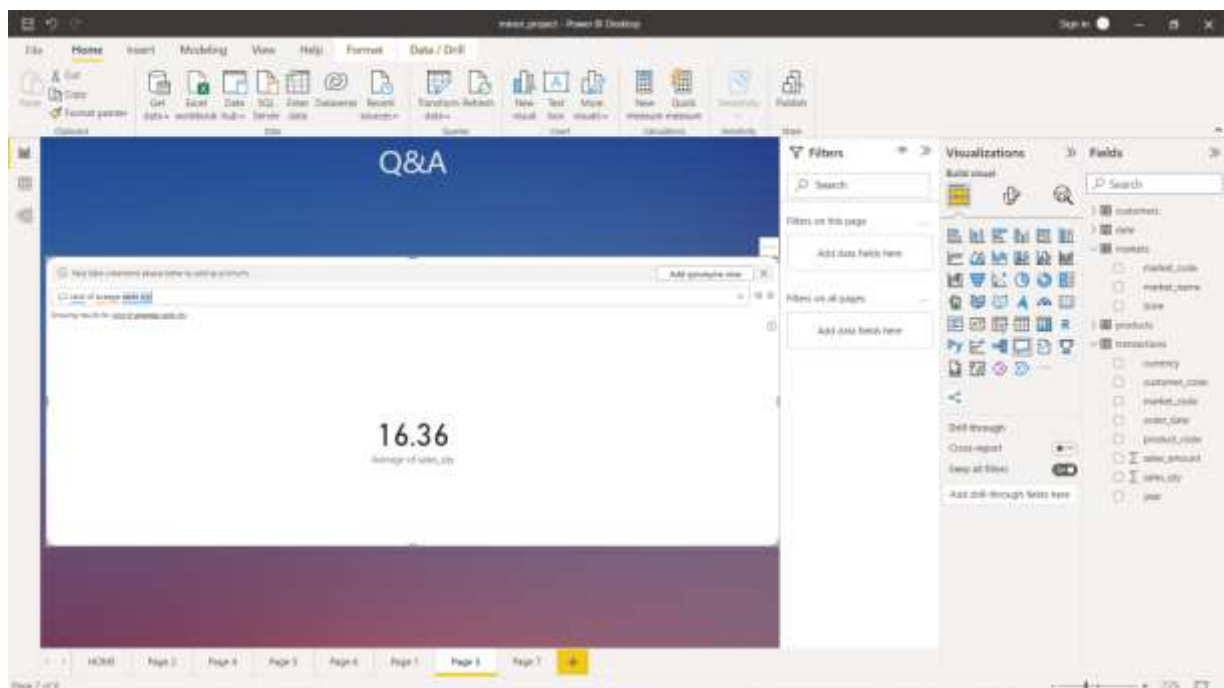
**Details:**



For example if we want to see the pie chart of year Vs sales\_qty we can directly ask the question in Q & A it will provide us the visual

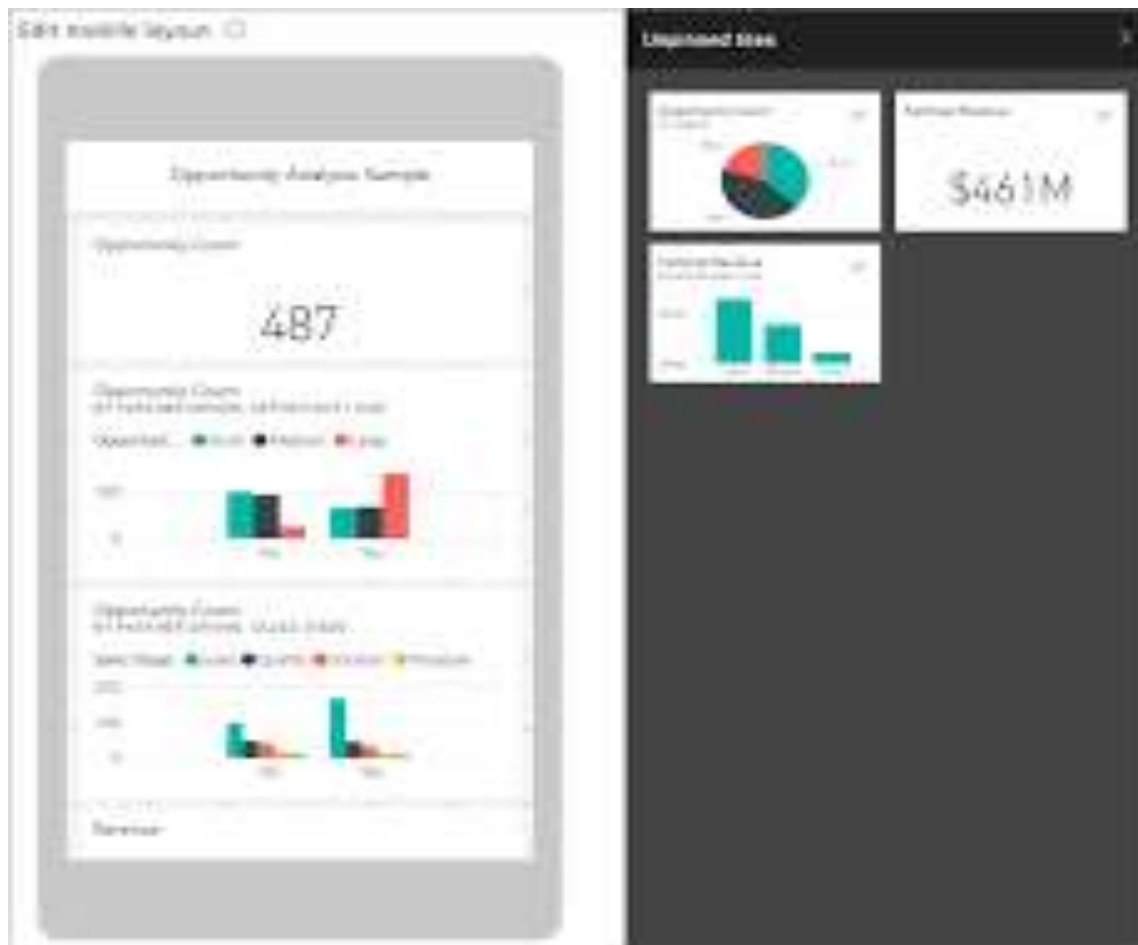


Another example suppose we want to see the average of sales\_qty we can use Q&A as shown below



## Step 2: Accessing the dashboard as a mobile application

Preparing the dashboard in the power BI and then publishing it on the website. By clicking on mobile layout, we can create the mobile app view.





- **ADVANTAGES AND DISADVANTAGES :-**

**Affordability :**

A major advantage of using Power BI is that it is inexpensive compared to other cloud service providers. The Power BI Desktop version is free of cost, you can download and start making the reports on your computer. However, if you wanna share your reports on the cloud you have to pay 9.99\$ per user per month.

**Excel Integration :**

In Power BI, you can also save data to Excel. No matter how great the data is presented using Graphs, maps and charts using data visualisation tools, people still tend to have the data in their excel sheet. For example, you can get the data of a manufacturing unit for the past six months within a few clicks from Power BI.

**Custom Visualisation :**

Power BI offers a wide range of custom visualisations where developers can take your requirements and convert them to KPI's, charts, graphs, maps etc.

**Data Accessibility and Interactive visualisation :**

Power BI provides great access to all the data source and the data sets that you create while designing the reports, in a centralized location. You can access the data anytime, anywhere from any device multiple times. Users can interact with the dashboards using filters, highlighting features, etc. by simple clicks.

**Newly Developed Features :**

One of the other advantages is that Microsoft provides the users with **monthly updates**.

- **DISADVANTAGES :-**

*Rigid formulas and Large data volumes :*

The language used in Power BI is mostly DAX. However, there are only so many actions that can be performed using DAX Formula. It may allow you to create a multitude of custom calculations but it is not flexible and the query becomes cumbersome. The outcome of these cannot always be good for the dashboard performance or speed.

Power BI has a limit on the size of data that it can ingest. If you wish to import data larger than 2GB you need to extend your free version to the paid version. And using Power BI can be frustrating at times as it can crash while processing large datasets with complex measures.

*Table Relationships :*

Another con is that Power BI can have difficulty in handling data with complex relationships between tables. You need to create data models carefully with unique fields so that Power BI can distinguish the relationship with the appropriate table.

*User Interface :*

Power BI has a cramped and bulky interface, as it is filled with many icon options that block the view of the reports and dashboards. Moreover the **Tooltip** in Power BI has a limited application.

*Compatibility with IOS :*


Power BI does not provide a compatible option in the **Desktop version for Mac Users**.

*Limitation in Excel Integration :*


Even though the excel integration is an advantage of Power BI, you can only download data upto **150,000 rows**.


# **OBJECTIVES OF SALES INSIGHTS DATA TESTING**


The software testing is usually performed for the following objectives:-

 **ALL DATA IN CSV.FILE FORMAT:-** At first we check computer and the software are mainly used for complex and critical applications and a bug or fault in software causes severe losses. So a great consideration is required for checking for quality of software.

 **VERIFICATION AND VALIDATION:-**

 ***Verification*** means to test that all the data we put in the power bi there is no error occur in the data are data are present there are no missing data and all the data we need that's are categorical data.

 ***Validation*** means to check whether that all the data we analyze that's valid or not.

 **RELIABILITY ESTIMATION:-** After the analyzing of all data which decision we get that's are useful for the company or not. The failure of the decision to make the impact in the analyzing of data..

## STRATEGY FOR ANALYZING DATA

Different levels of testing are used in the test process; each level of testing aims to test different aspects of the system.

- 🔧 The first level is .csv file or not . In this testing, all files are .csv file or convert to .csv file to execute the job. It focuses on verification efforts.
- 🔧 The second level is ***data is correct or not***. It is a systematic technique for constructing the program structure. In this testing, we make sure that all the data are correct and there is no missing value in this files.
- 🔧 Third level is ***dashboard designing***. Dashboard designing is the main part of the analyze the data. In Dashboard we see the all types of pie chart bar chart etc. In dashboard we can visualize the data and make decision for the company.

## **CONCLUSION**

This project developed, incorporated all the activities involved to analyze the sales data of the company. It provides all necessary information to the management as well as the customer with the use of this system; the user can simply sit in front of the system and monitor all the activities without any physical movement of the file. Management can service the customers request best in time. The system provides quickly and valuable information. These modules have been integrated for effective use of the management for future forecasting and for the current need.

## **SCOPE FOR FURTHER DEVELOPMENT**

The system can be designed for further enhancement .This could also be developed according to the growing needs of the company.

# **BIBLIOGRAPHY**

- **For Talend installation**

- <http://www.talend.com/index.html>

- **For Power BI Installation**

- <http://www.powerbi.microsoft.com>

- **Reference websites**

- [www.power BI.com](http://www.powerbi.com)
- [www.Talend.Tutorial.com](http://www.Talend.Tutorial.com)
- [www.w3schools.com](http://www.w3schools.com)
- <http://www.tutorialspoint.com/sales insight power BI /index.htm>

- **Reference Books**

- POWER BI.
  - ETL EXTENTION.
  - Learn POWER BI In Easy Way.
  - Complete reference Power BI.
-

## REFERENCE

1. Microsoft Power BI website (<https://powerbi.microsoft.com/>). This is the official website for Power BI, which provides information on the features and capabilities of the software.
2. "Power BI for Dummies" by Dan Clark (<https://www.amazon.com/Power-BI-Dummies-Dan-Clark/dp/1119549729>). This is a beginner's guide to Power BI that covers the basics of the software and how to use it for data visualization and analysis.
3. "Data Visualization with Power BI" by Reza Rad (<https://www.amazon.com/Data-Visualization-Power-BI-Rad/dp/1789808906>). This is a more advanced guide to using Power BI for data Visualization, covering topics such as creating custom visuals and using Power BI with big data.
4. "Power BI Cookbook" by Anthony B. Smith (<https://www.amazon.com/Power-BI-Cookbook-Anthony-Smith/dp/1788298756>). This book provides a collection of practical recipes for using Power BI to solve common data analysis challenges.
5. "Power BI Best Practices" by Miguel Myers (<https://www.amazon.com/Power-BI-Best-Practices-Miguel/dp/1484254634>), this book covers best practices for designing and implementing Power BI dashboards and reports, including tips for optimizing performance and scalability.

