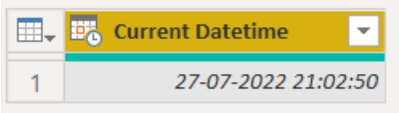
Power BI – Transformation, M coding and DAX

"Transformation, M Coding and DAX in PowerBI" is an academic assignment I worked on as part of the Business Intelligence course in my Big Data Analytics program. The intent of this project is to perform a given set of tasks using data provided by faculty to explore various PowerBI tools and functionalities especially M coding and DAX.

Table of Contents

Part 1: Power BI Transformation and M-Code

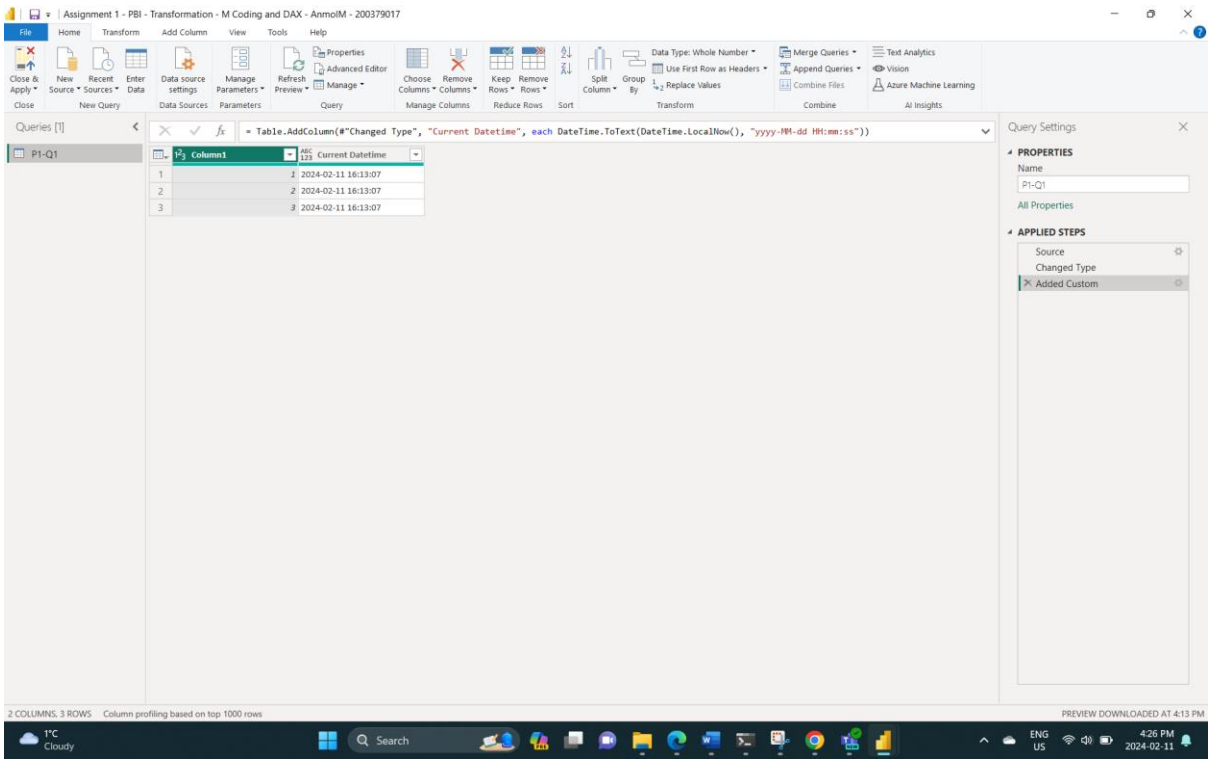
1. Capture the current datatime in Power Query using Power BI M-Code.



Answer:

We add a custom Column, label it as we need and input the following M-code to get the current datetime for each row - *DateTime.ToText(DateTime.LocalNow(), "yyyy-MM-dd HH:mm:ss")*

In the screenshot below, we can see this custom column showing us the date and time for each row as intended.



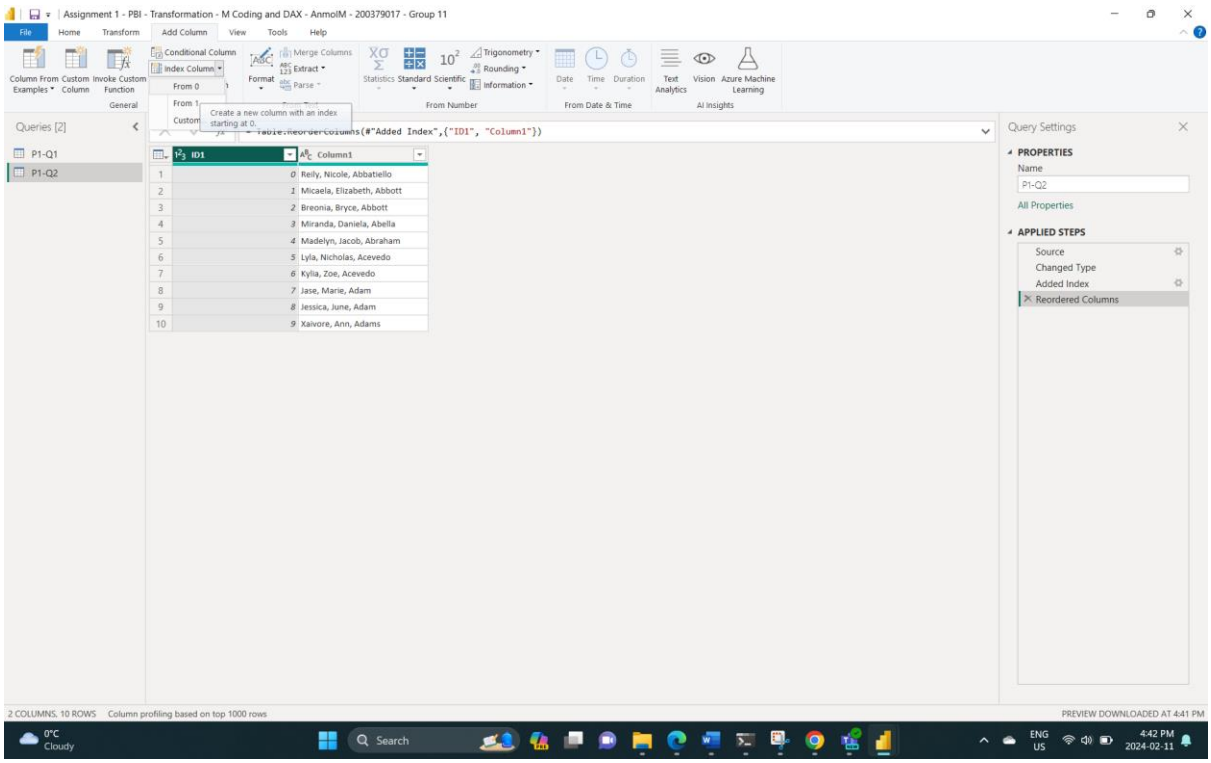
2. Create ID1, ID2, and ID3 columns against Full Name as shown in the table below and populate the required value in Power Query using M-Code.
    a. ID1: Value starts from 0 and increments by 1.
    b. ID2: Value starts from 1 and increments by 1.
    c. ID3: Value starts from 0 and increments by 5.

Note: Create a column using "Enter Data" option and populate the value mentioned in the table. Once the "Full Name" column is created add the other columns (ID1, ID2, and ID3) based on the above condition.

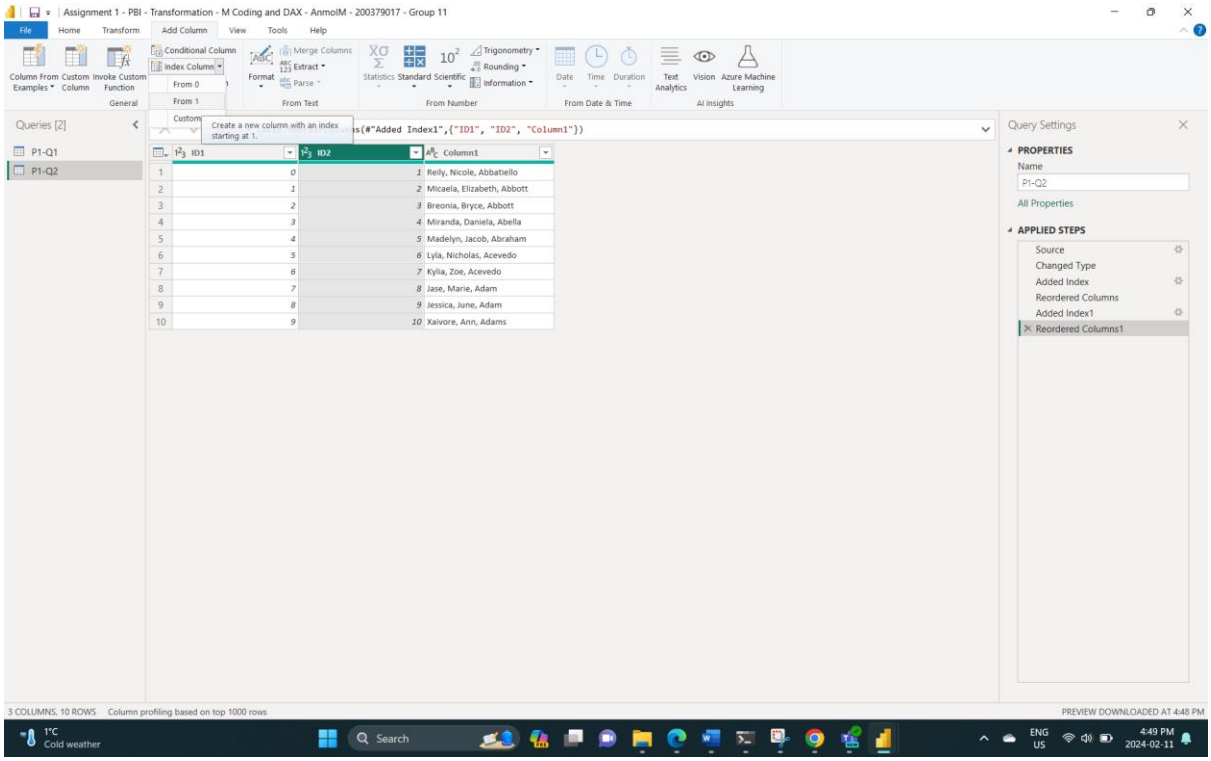
ID1	ID2	ID3	Full Name
0	1	0	Reily, Nicole, Abbatiello
1	2	5	Micaela, Elizabeth, Abbott
2	3	10	Breonia, Bryce, Abbott
3	4	15	Miranda, Daniela, Abella
4	5	20	Madelyn, Jacob, Abraham
5	6	25	Lyla, Nicholas, Acevedo
6	7	30	Kylia, Zoe, Acevedo
7	8	35	Jase, Marie, Adam
8	9	40	Jessica, June, Adam
9	10	45	Xaivore, Ann, Adams

Answer:

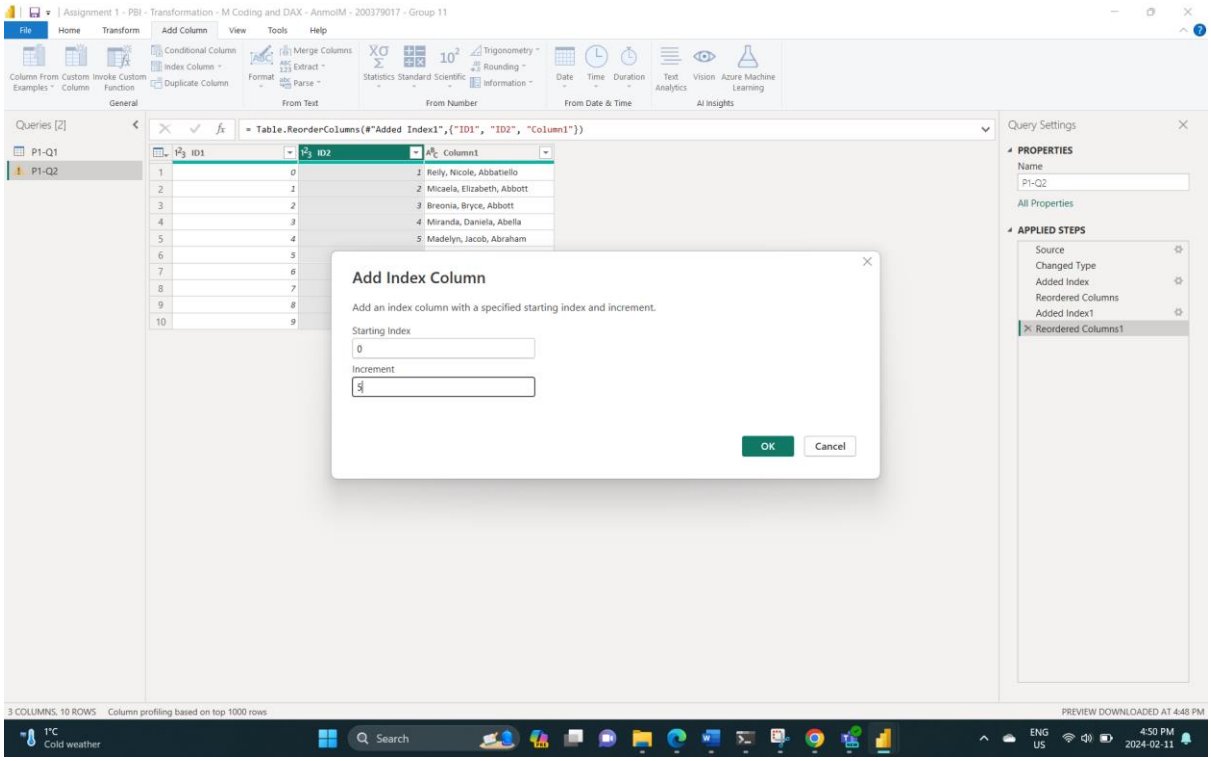
a) ID1: Value starts from 0 and increments by 1 – Go to add a column, click on dropdown by the Index Column and select 'from 0' as shown in the picture below. This should add an index starting from 0 with an increment of 1. Rename the column name as needed.



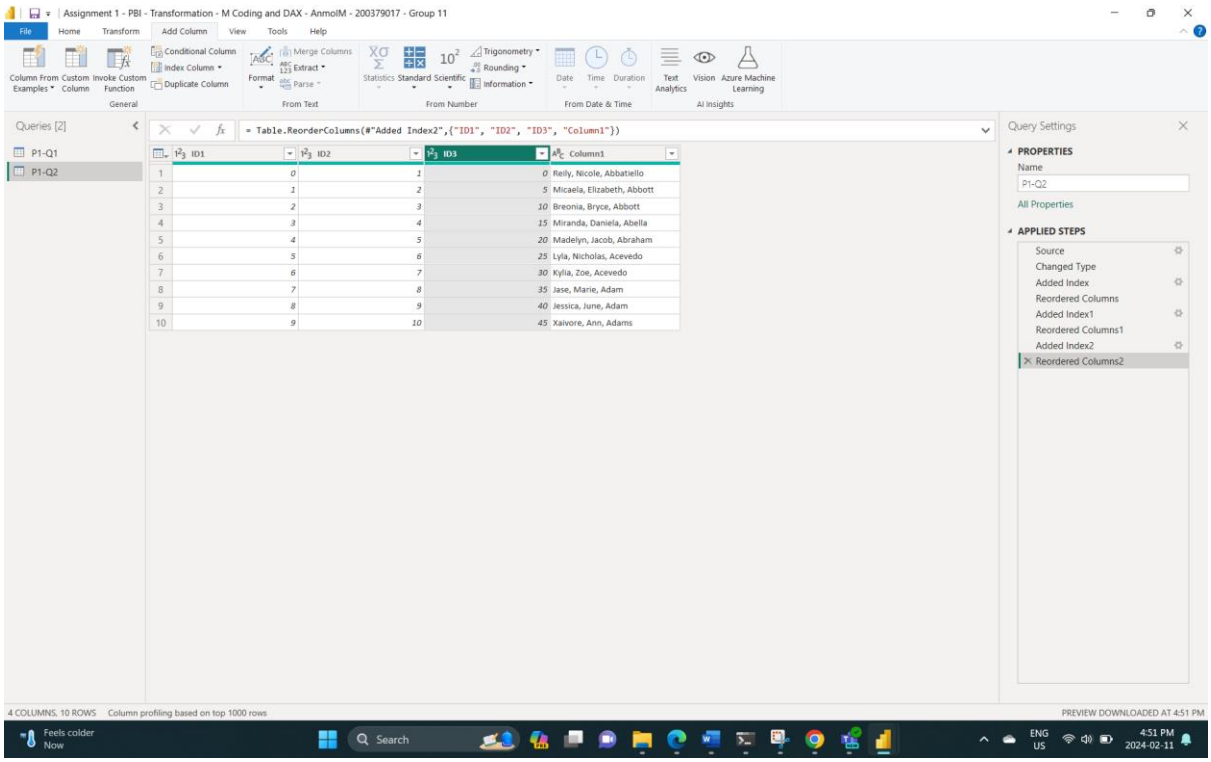
b) ID2: Value starts from 1 and increments by 1: Go to add a column, click on dropdown by the Index Column and select ‘from 1’ as shown in the picture below. This should add an index starting from 1 with an increment of 1. Rename the column name as needed.



c) ID3: Value starts from 0 and increments by 5: Go to add a column, click on dropdown by the Index Column and select ‘custom’ and enter the info as shown in the picture below.



This should add an index starting from 0 with an increment of 5 as shown in the picture below. Rename the column name as needed.



	ID1	ID2	ID3	Column1
1	0	1	0	Reily, Nicole, Abbatiello
2	1	2	5	Micaela, Elizabeth, Abbott
3	2	3	10	Breonia, Bryce, Abbott
4	3	4	15	Miranda, Daniela, Abella
5	4	5	20	Madelyn, Jacob, Abraham
6	5	6	25	Lyla, Nicholas, Acevedo
7	6	7	30	Kylia, Zoe, Acevedo
8	7	8	35	Jase, Marie, Adam
9	8	9	40	Jessica, June, Adam
10	9	10	45	Xaivore, Ann, Adams

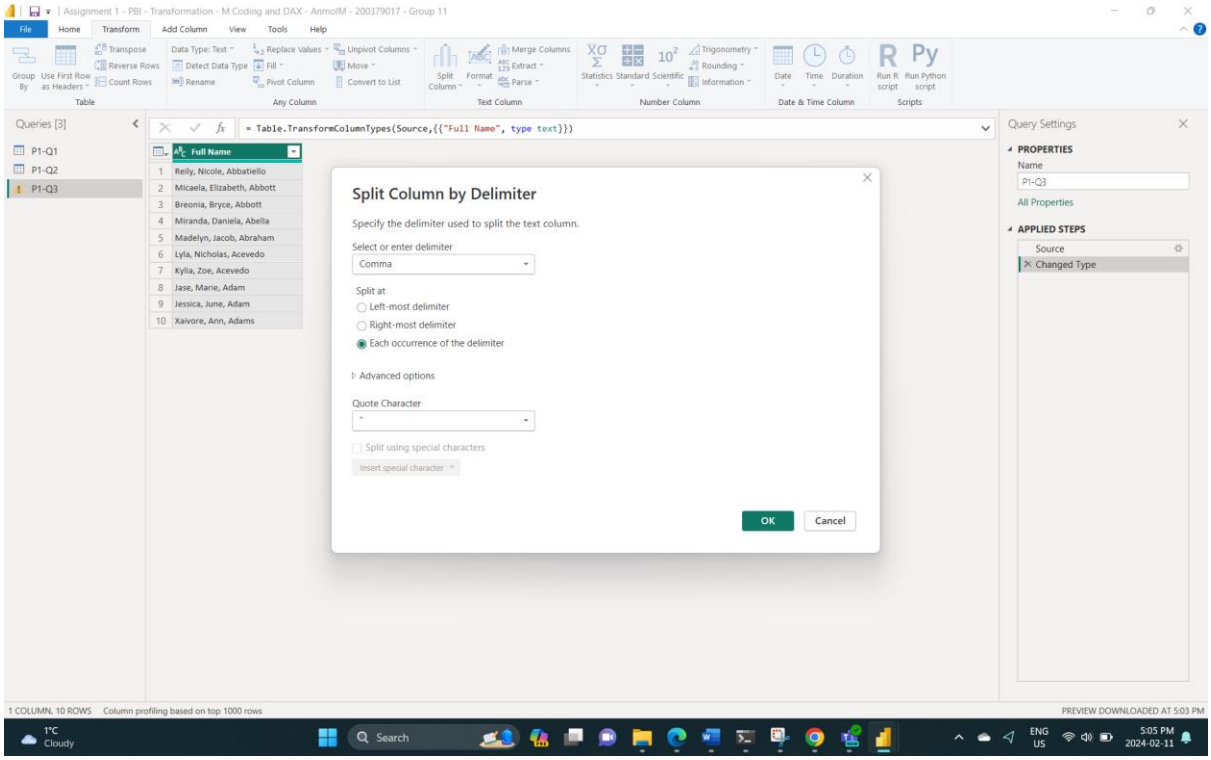
3. Extract First Name, Middle Name, and Last Name from the "Full Name" column as mentioned below using Power Query. [Use Enter data to create the below table with a "Full Name" column].
   Note: Full Name is a combination of "First Name, Middle Name, Last Name".

Full Name
Reily, Nicole, Abbatiello
Micaela, Elizabeth, Abbott
Breonia, Bryce, Abbott
Miranda, Daniela, Abella
Madelyn, Jacob, Abraham
Lyla, Nicholas, Acevedo
Kylia, Zoe, Acevedo
Jase, Marie, Adam
Jessica, June, Adam
Xaivore, Ann, Adams

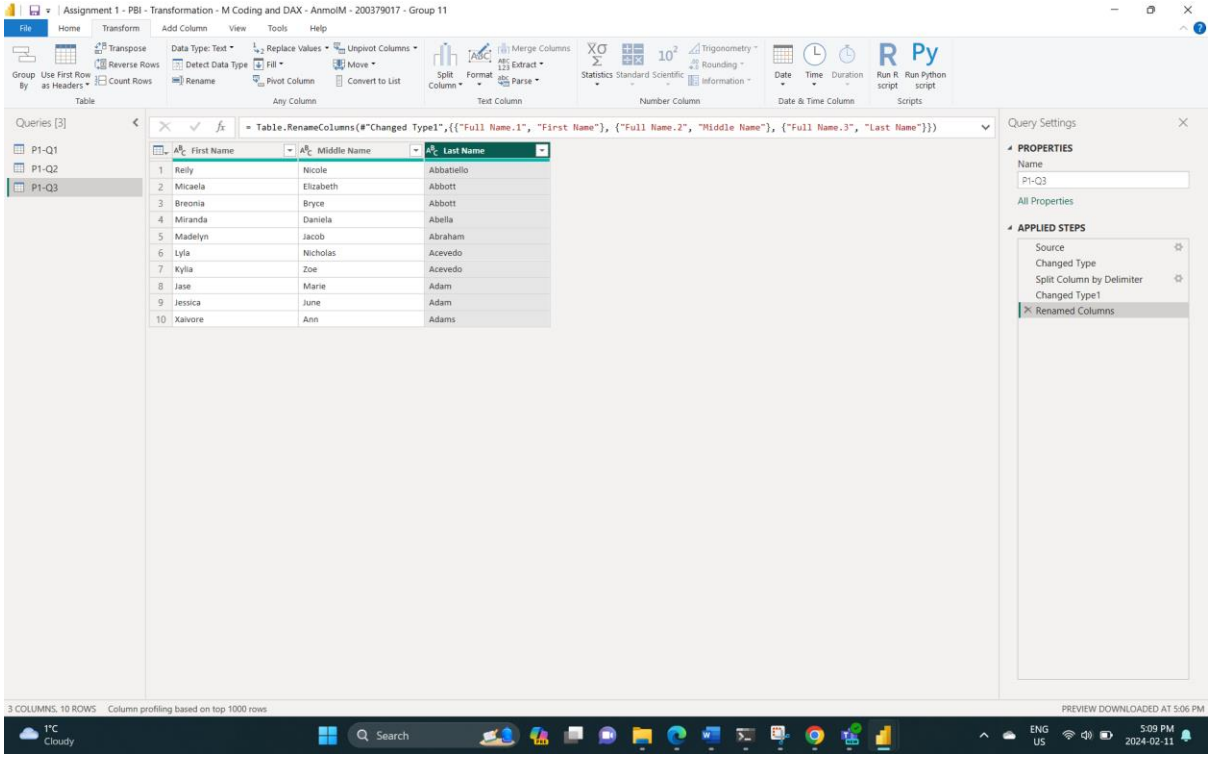
Answer:

After making the table, head to transform and click on split columns by delimiter to transform our Full name column into three columns showing First, Middle and Last Name separately. Select the

delimiter we want to use to split the column and choose where we want to make the split. In our case, we choose at each occurrence to get each of the tree parts of the Full Name.



After applying the desired settings, we press ok and we get the separated columns as shown below. Rename columns as needed.



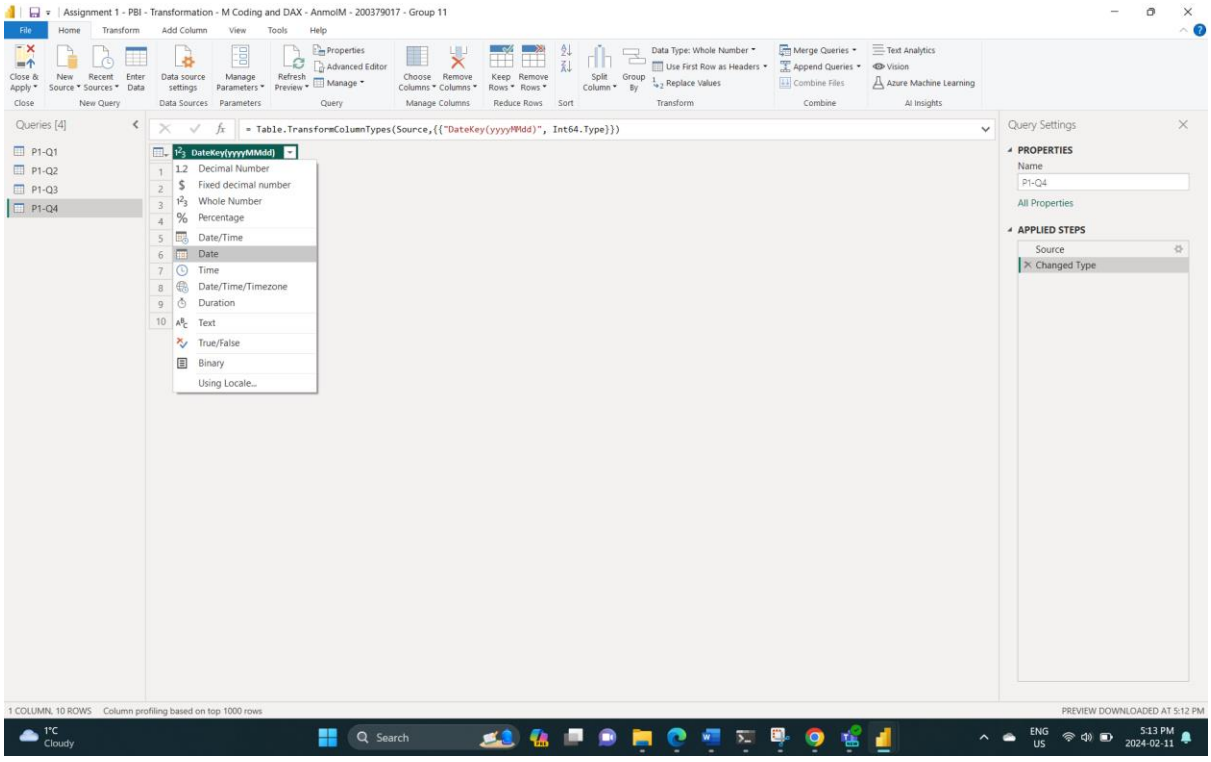
4. Extract Date from DateKey column present in the table below using Power Query.

DateKey(yyyyMMdd)
20220101
20220102
20220103
20220104
20220105
20220106
20220107
20220608
20220109
20220110

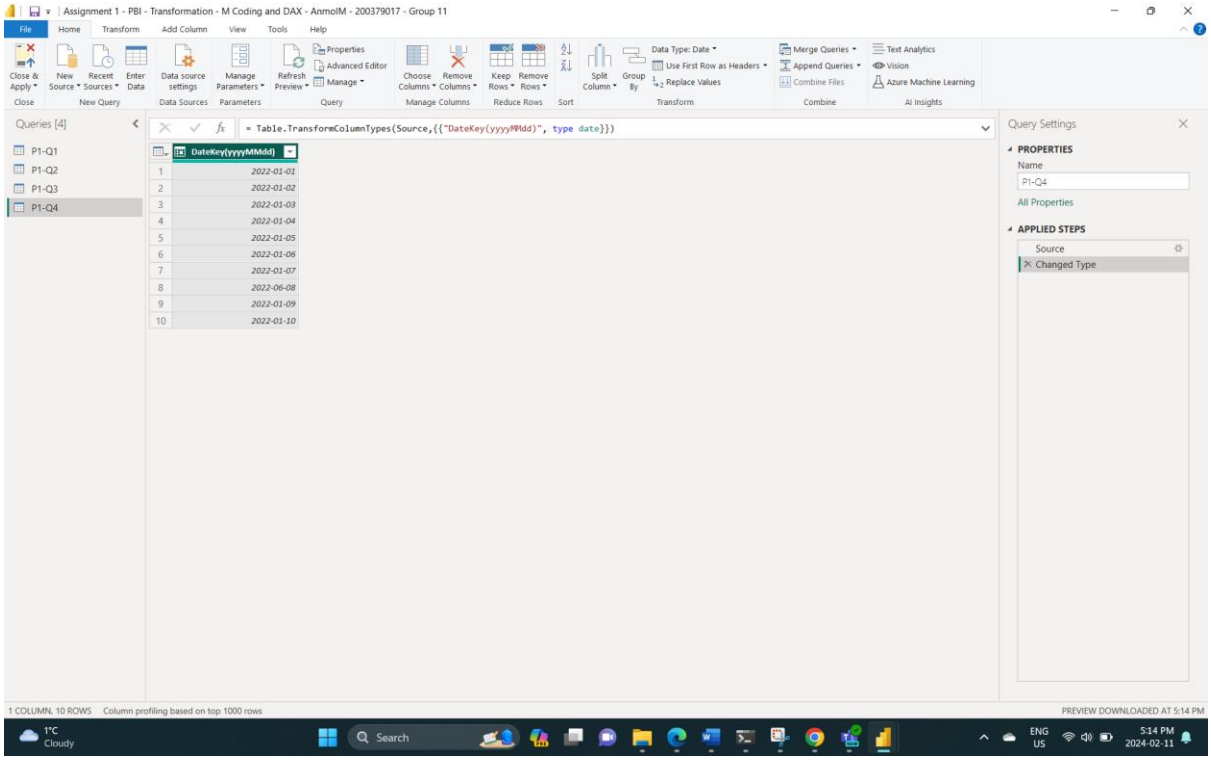
Note: Use “Enter Data” option to create the above table in Power BI.

Answer:

Enter data to make the DateKey column in a new table. Click on the data type and convert it to Date from numerical.



This extracts our numerical DateKey Data and converts it into Date format as shown below:



5. Create two parameters to input the Server and Database name and import a Product table using Power Query.

   Use the following configuration for the parameters-

   - Parameter1: Servername
     Local server as Localhost & 127.0.0.1
   - Parameter2: Database
     Database as “AdventureWorks2014” & “AdventureWorks2012”

   Once the parameter is created, import the mentioned table using MS SQL Server connector.

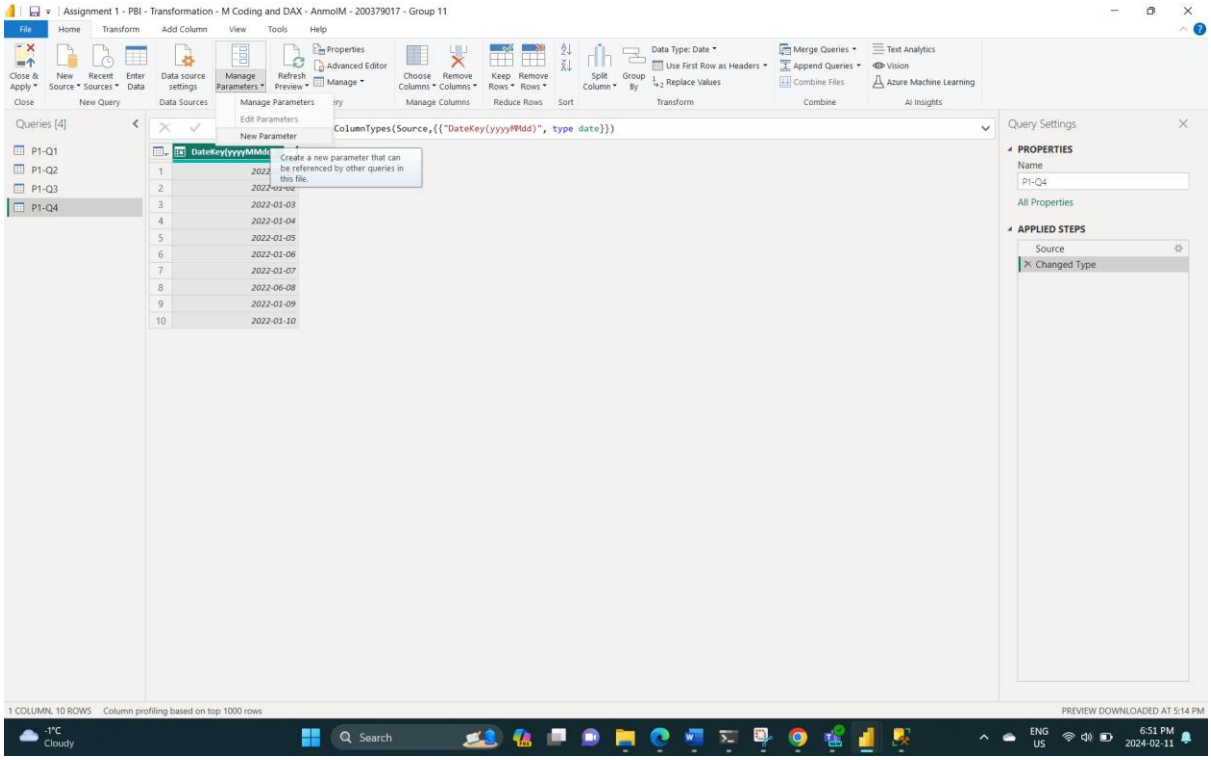
   Note: Before performing the above exercise make sure you have restored the mentioned database.

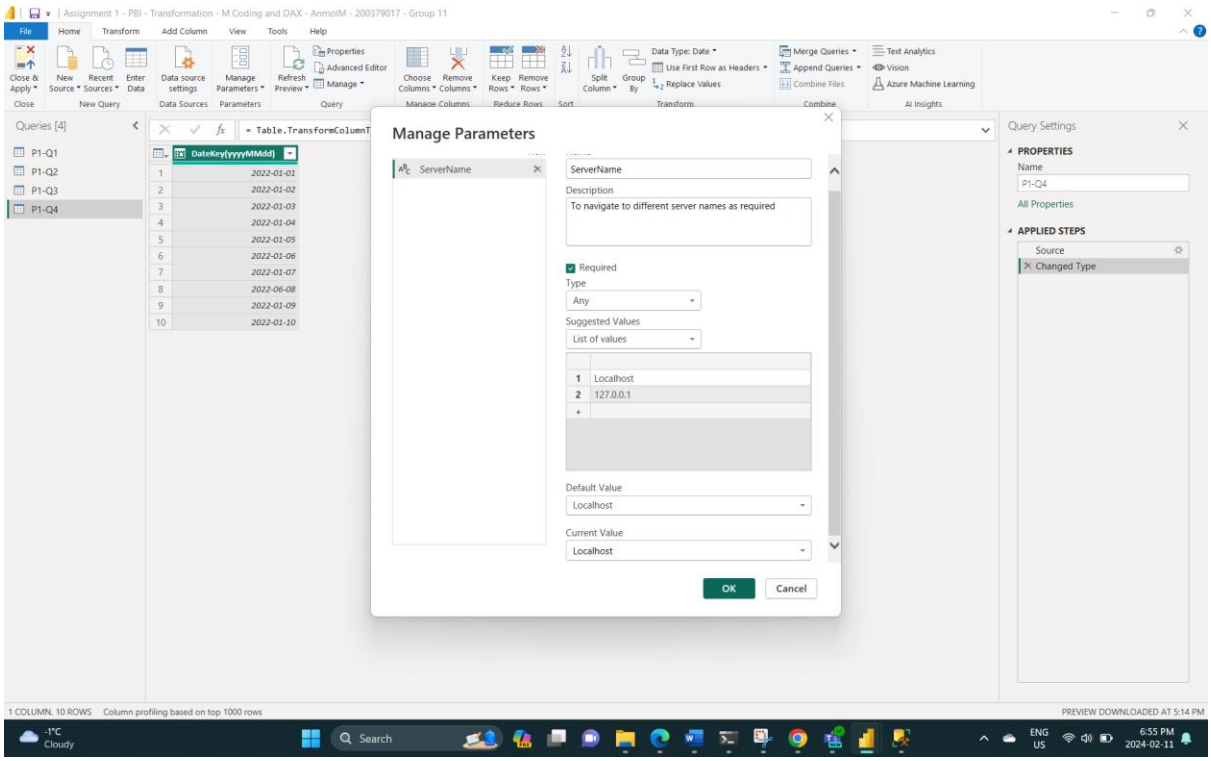
   Refer to the below URL to download the database backup file, after downloading the .bak file restore to the local SQL Server database.

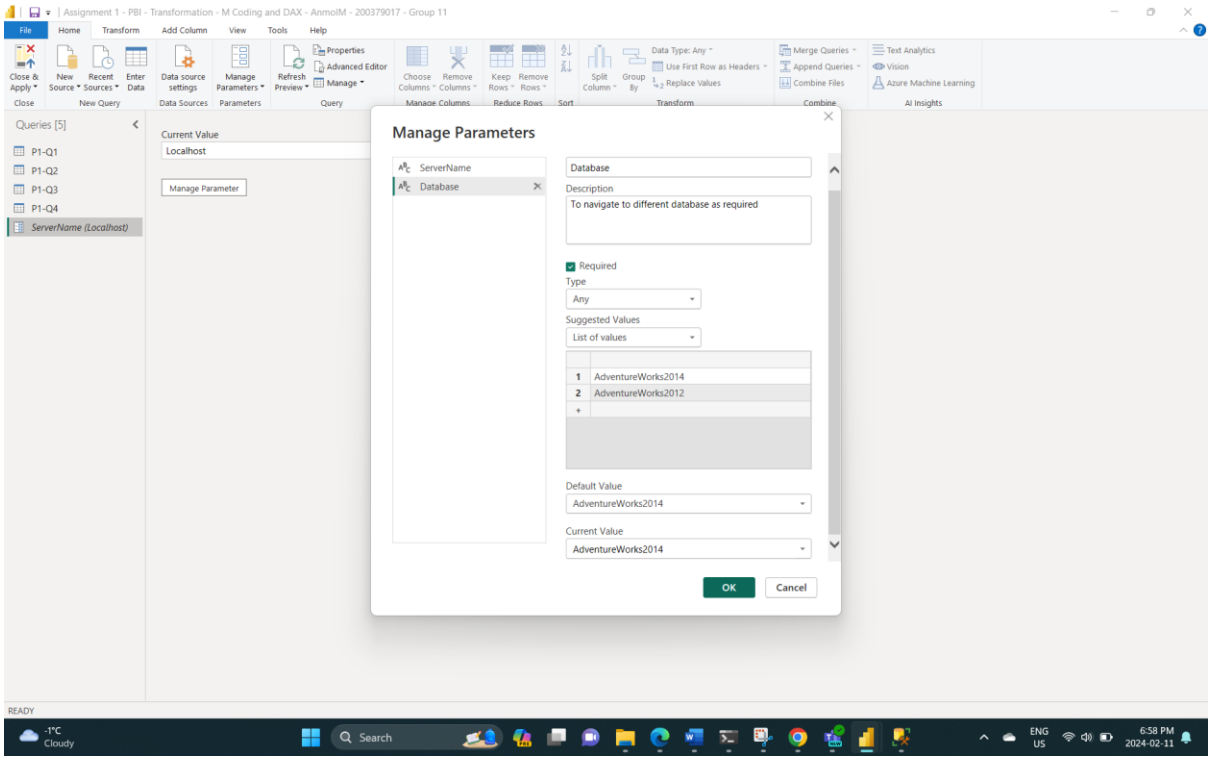
   ✓ AdventureWorks2014: https://github.com/Microsoft/sql-server-samples/releases/download/adventureworks/AdventureWorks2014.bak
   ✓ AdventureWorks2012: https://github.com/Microsoft/sql-server-samples/releases/download/adventureworks/AdventureWorks2012.bak

Answer:

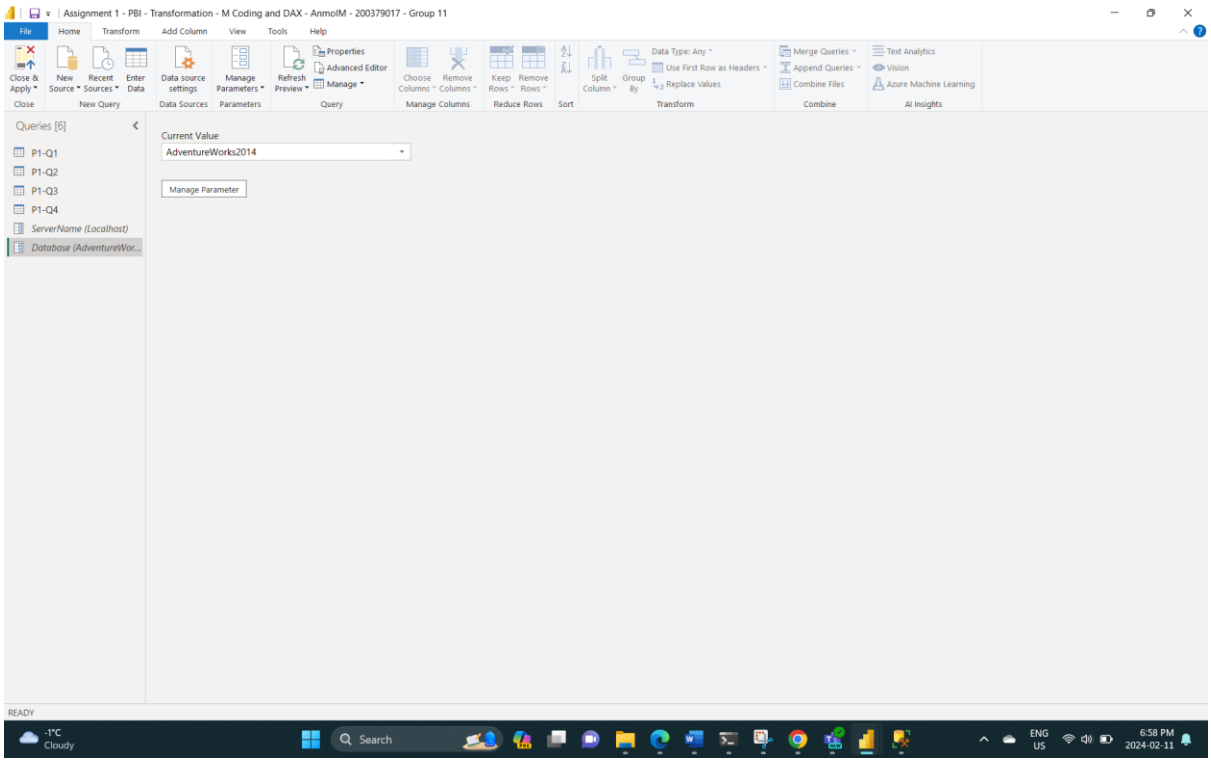
Dropdown Manage Parameters and select New parameter, fill out the configurations as needed as shown below for Servername and do the same steps for Database parameter and click ok. This creates our parameters for further use as required.



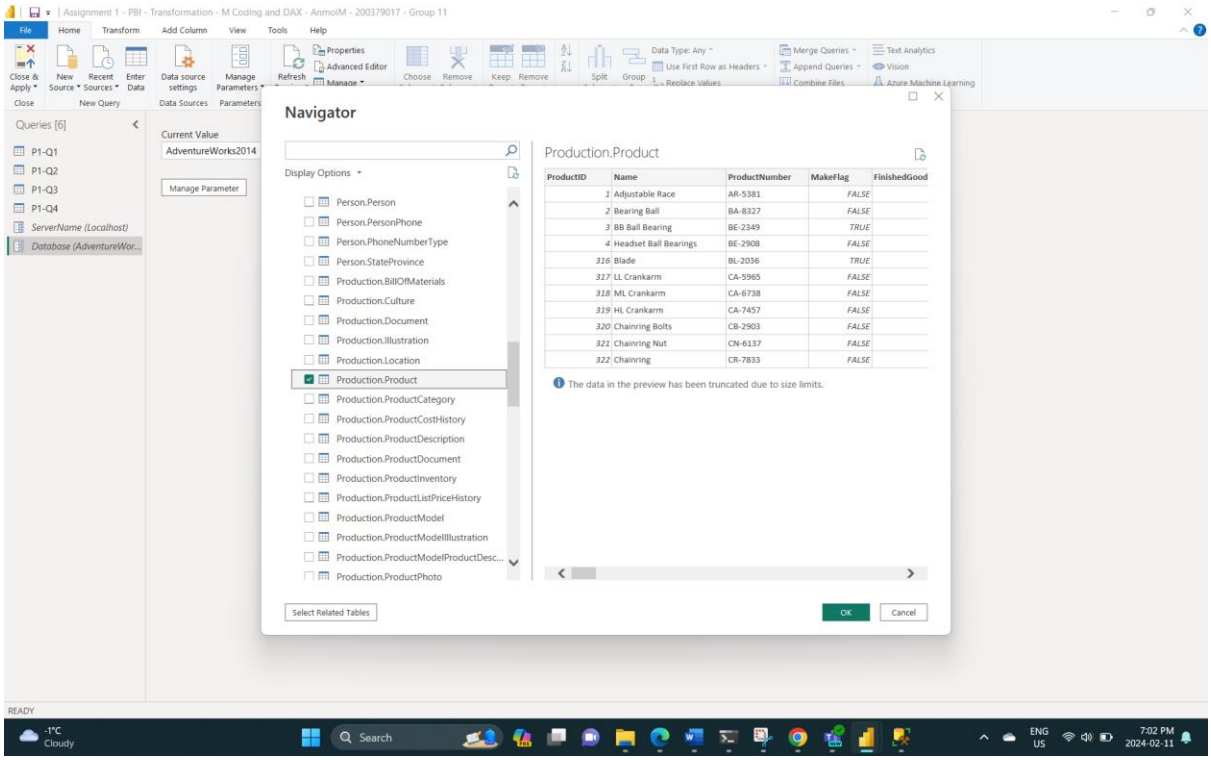


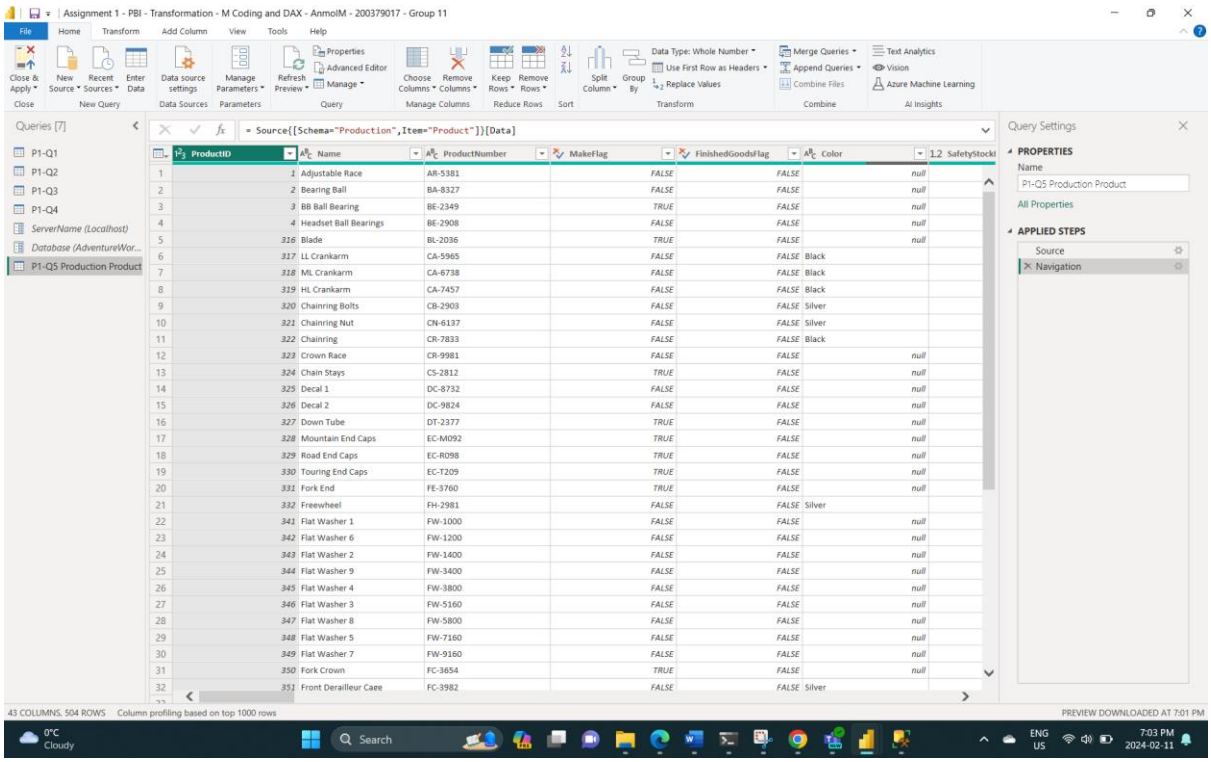


As we can see in the screenshot below, our two parameters have been created and can be accessed from under the queries tab:

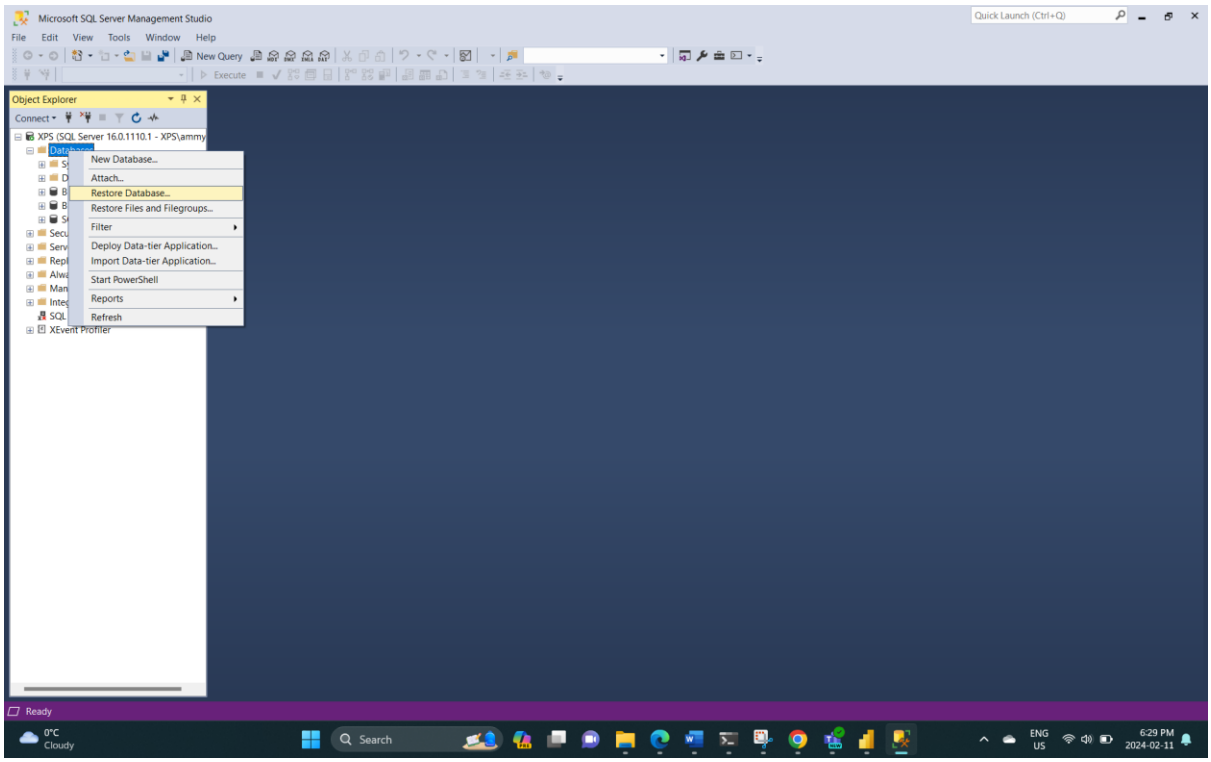


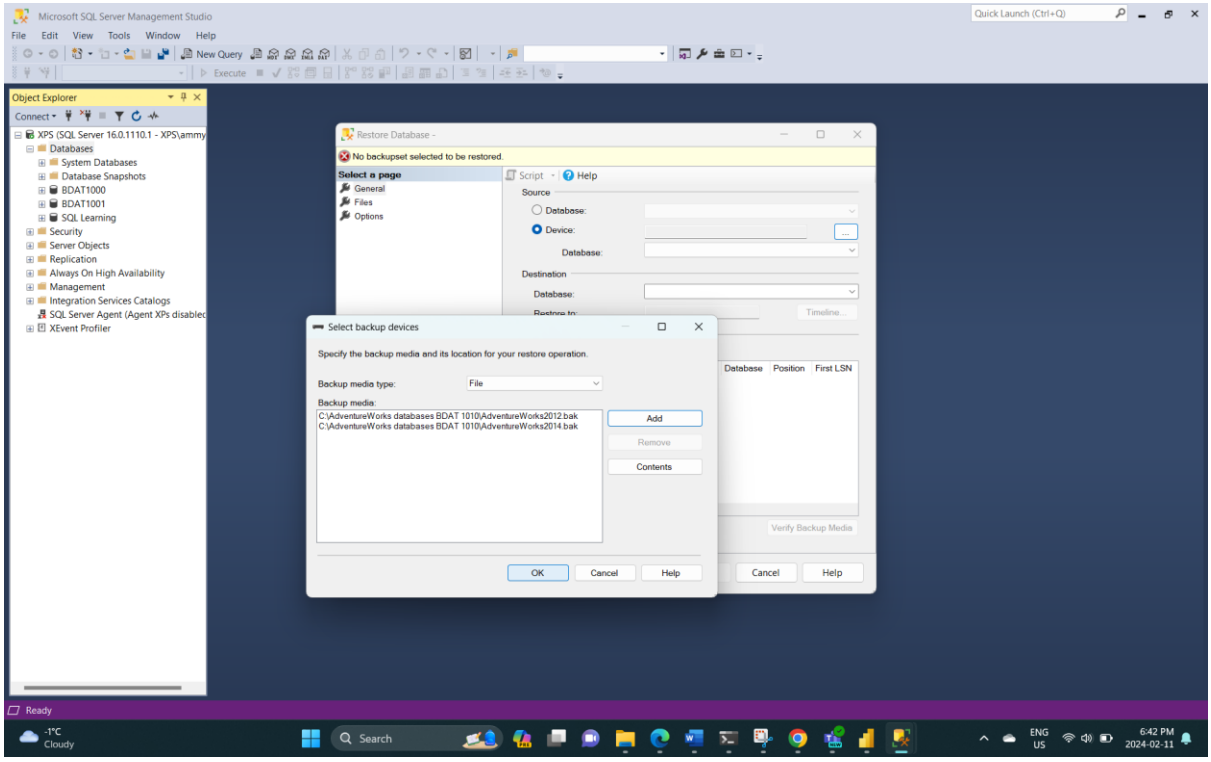
Now we import a product table using our created parameters. Click on new source and select SQL server, select our newly created parameters under server name and database to import. After a connection has been made, import the table from the selected database per criteria as shown below.

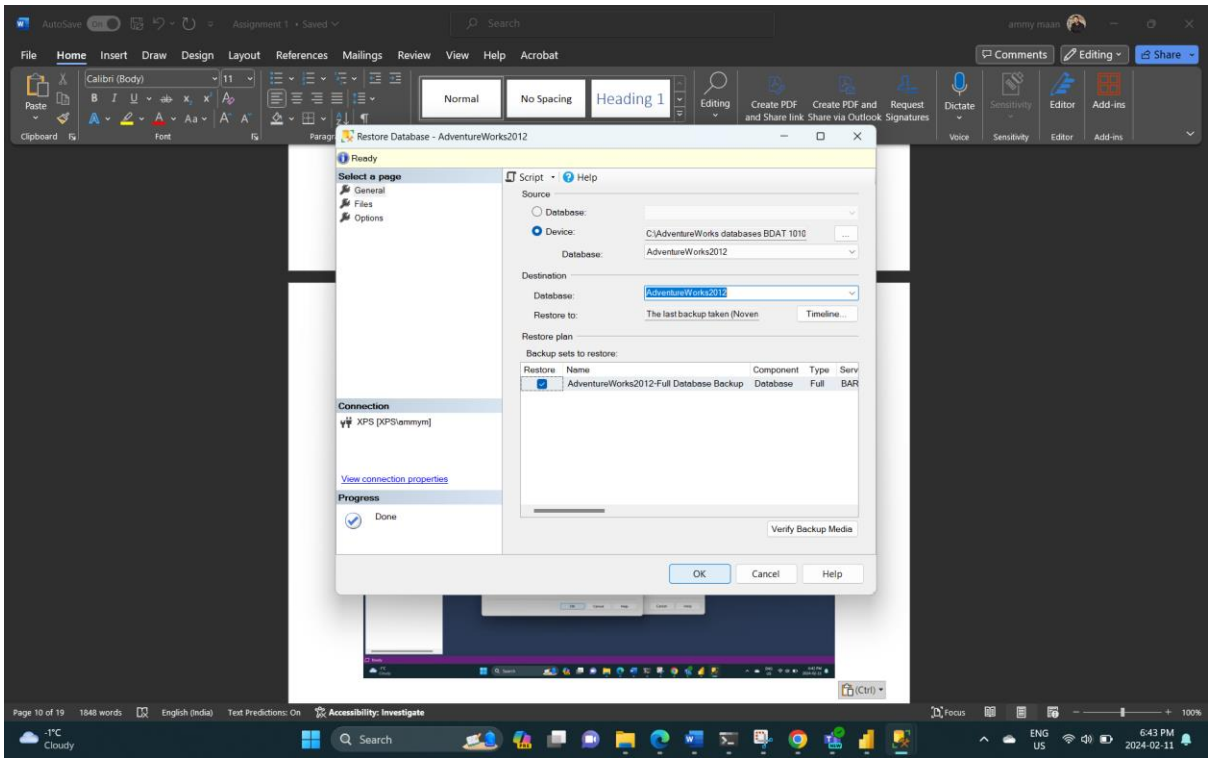




#Note to self - Restoring a database: Connect to SQL Server, right click on databases and click on restore database and follow instructions as shown below:


Microsoft SQL Server Management Studio
File Edit View Tools Window Help
New Query
Execute
Quick Launch (Ctrl+Q)
Object Explorer
Connect
XPS (SQL Server 16.0.1110.1 - XPS\ammy
Databases
New Database...
Attach...
Restore Database...
Restore Files and Filegroups...
Filter
Deploy Data-tier Application...
Import Data-tier Application...
Start PowerShell
Reports
Refresh
XEvent Profiler
Ready
0°C
Cloudy
Search
ENG
US
6:29 PM
2024-02-11


Microsoft SQL Server Management Studio
File Edit View Tools Window Help
New Query
Execute
Quick Launch (Ctrl+Q)
Object Explorer
Connect
XPS (SQL Server 16.0.1110.1 - XPS\ammy
Databases
System Databases
Database Snapshots
BDAT1000
BDAT1001
SQL Learning
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
Restore Database -
No backupset selected to be restored.
Select a page
General
Files
Options
Script
Help
Source
Database:
Device:
Database:
Destination
Database:
Restore to:
Timeline...
Select backup devices
Specify the backup media and its location for your restore operation.
Backup media type:
File
Backup media:
C:\AdventureWorks databases BDAT 1010\AdventureWorks2012.bak
C:\AdventureWorks databases BDAT 1010\AdventureWorks2014.bak
Add
Remove
Contents
OK
Cancel
Help
Database
Position
First LSN
Verify Backup Media
Cancel
Help
Ready
-1°C
Cloudy
Search
ENG
US
6:42 PM
2024-02-11



6. Convert the below table to matrix structure (Expected output) using Power Query.

Input Dataset:

Year	Month	Sales
2021	Jan	520
2021	Feb	360
2021	Mar	210
2021	Apr	320
2020	May	160
2020	Jun	963
2020	Jul	201
2020	Jan	302
2020	Feb	500
2020	Mar	450

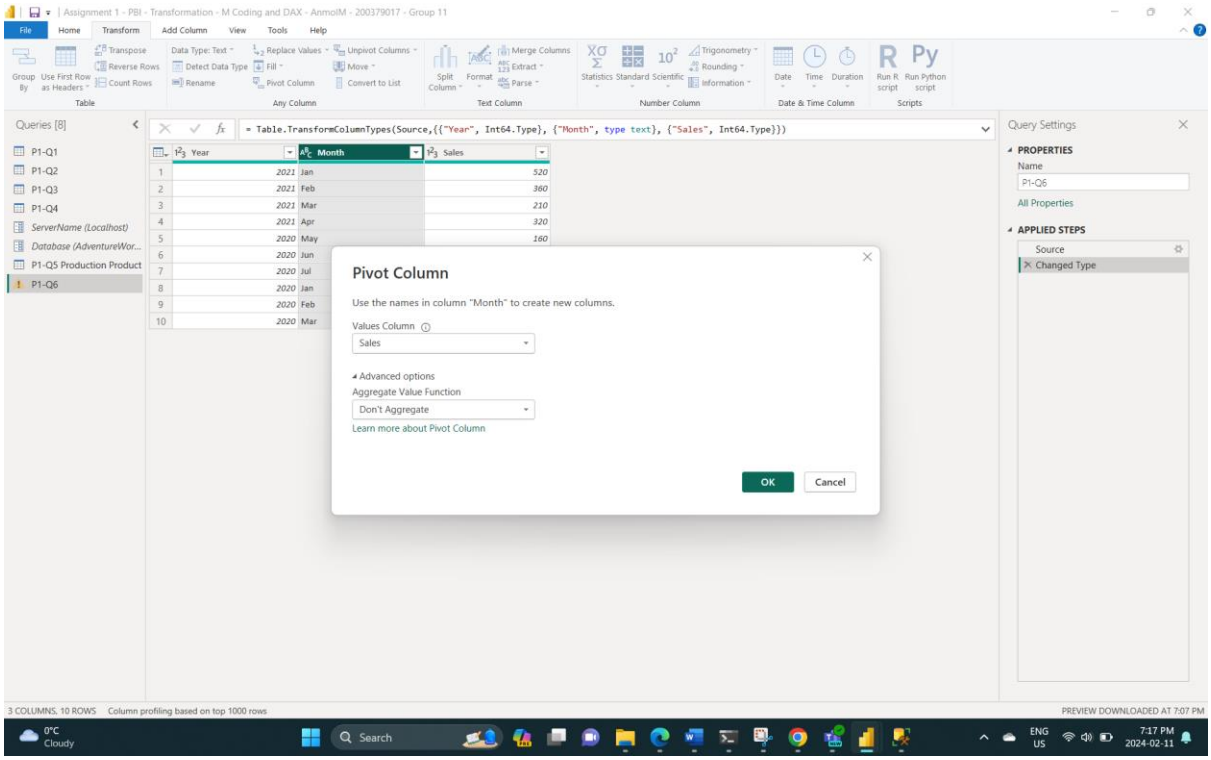
Expected Output:

Year	Jan	Feb	Mar	Apr	May	Jun	Jul
2020	302	500	450	null	160	963	201
2021	520	360	210	320	null	null	null

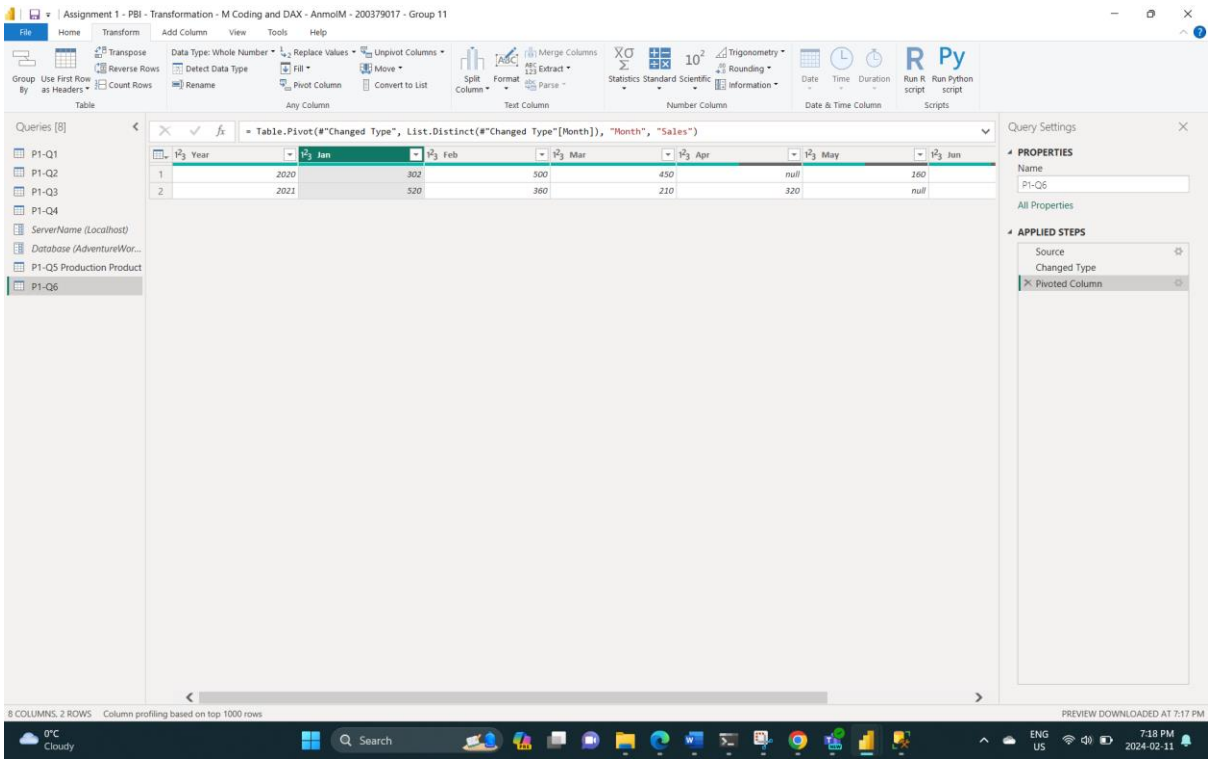
Note: Use “Enter Data” option to create the above table in Power BI.

Answer:

Create the table as follows, then select months column and click on pivot column based on our desired outcome. In this case we select Sales as values and under advanced setting, select don’t aggregate to keep the values free from additional aggregations.



After performing this step, we get our result.



7. Combine all the records from “StudentFromLocationA” table with “StudentFromLocationB” table using Power Query. You can use Append Query to combine these two tables.

Note: Use “Enter Data” option to create the below tables in Power BI.

StudentFromLocationA:

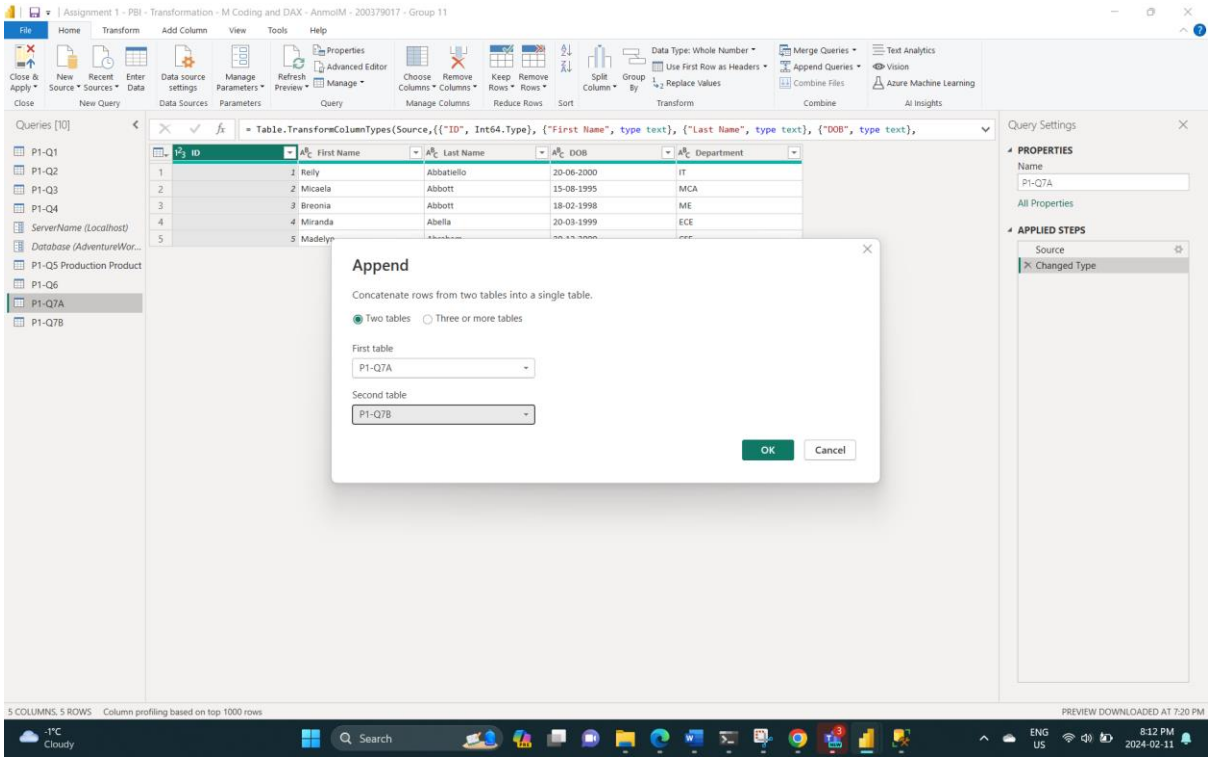
ID	First Name	Last Name	DOB	Department
1	Reily	Abbatiello	20-06-2000	IT
2	Micaela	Abbott	15-08-1995	MCA
3	Breonia	Abbott	18-02-1998	ME
4	Miranda	Abella	20-03-1999	ECE
5	Madelyn	Abraham	30-12-2000	CSE

StudentFromLocationB:

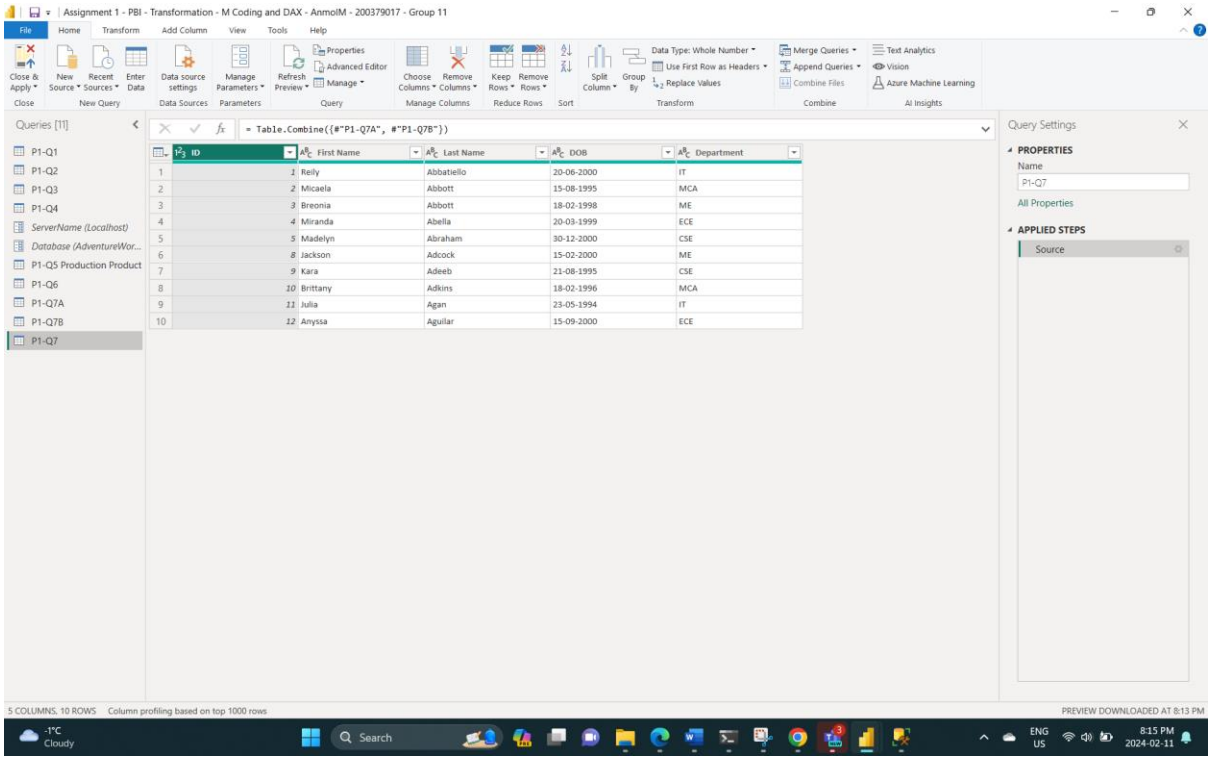
ID	First Name	Last Name	DOB	Department
8	Jackson	Adcock	15-02-2000	ME
9	Kara	Adeeb	21-08-1995	CSE
10	Brittany	Adkins	18-02-1996	MCA
11	Julia	Agan	23-05-1994	IT
12	Anyssa	Aguilar	15-09-2000	ECE

Answer:

Click on append Queries and select the two table we want to append.



This will join our two tables in order of selection as shown below:



8. Merge all the records from the "Student" table with "Department" using Power Query. You can use Merge Query to combine these two tables. Here DepartmentID is a key column in both the tables.

   Note: Use "Enter Data" option to create the below tables in Power BI.

Student

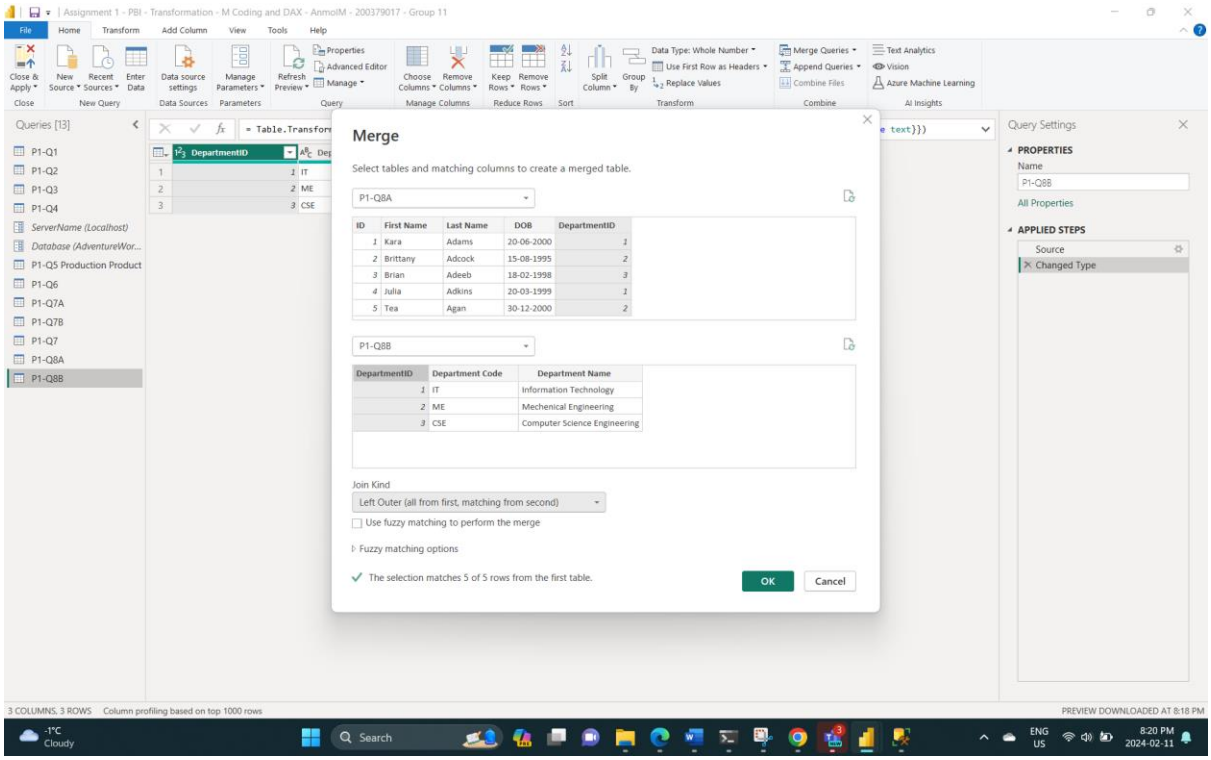
ID	First Name	Last Name	DOB	DepartmentID
1	Kara	Adams	20-06-2000	1
2	Brittany	Adcock	15-08-1995	2
3	Brian	Adeeb	18-02-1998	3
4	Julia	Adkins	20-03-1999	1
5	Tea	Agan	30-12-2000	2

Department

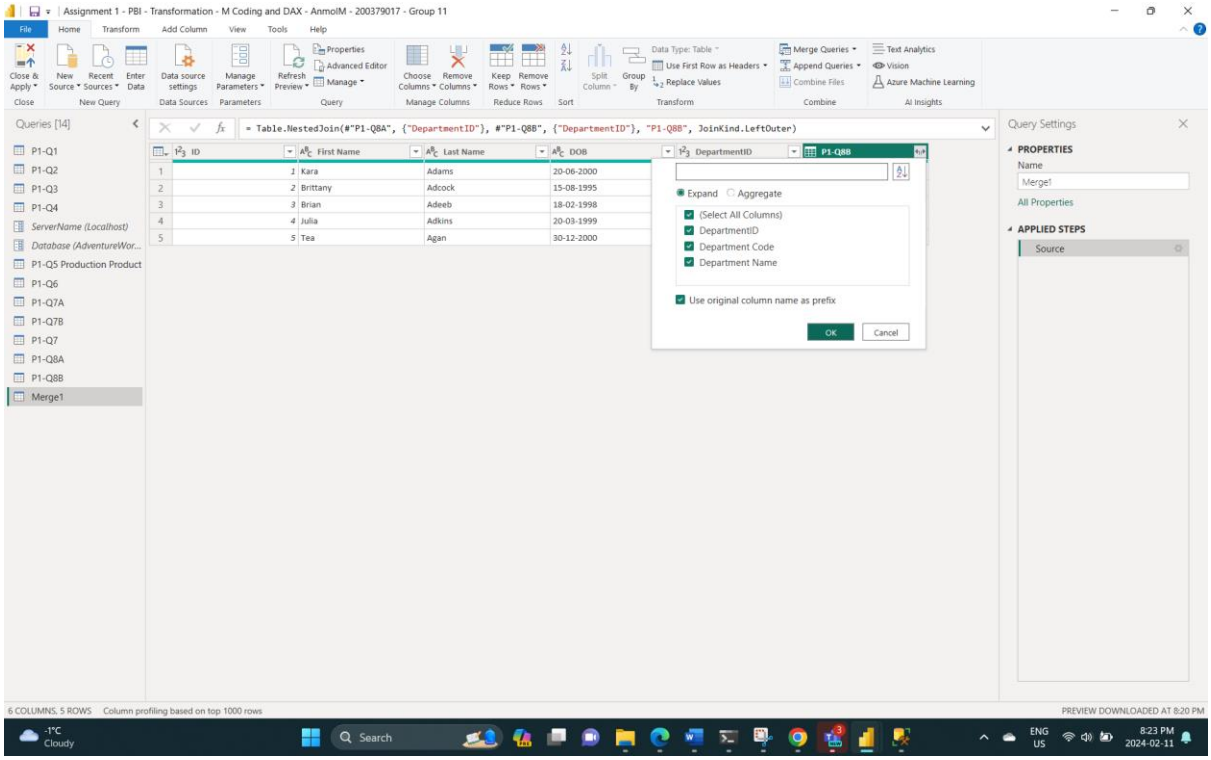
DepartmentID	Department Code	Department Name
1	IT	Information Technology
2	ME	Mechenical Engineering
3	CSE	Computer Science Engineering

Answer:

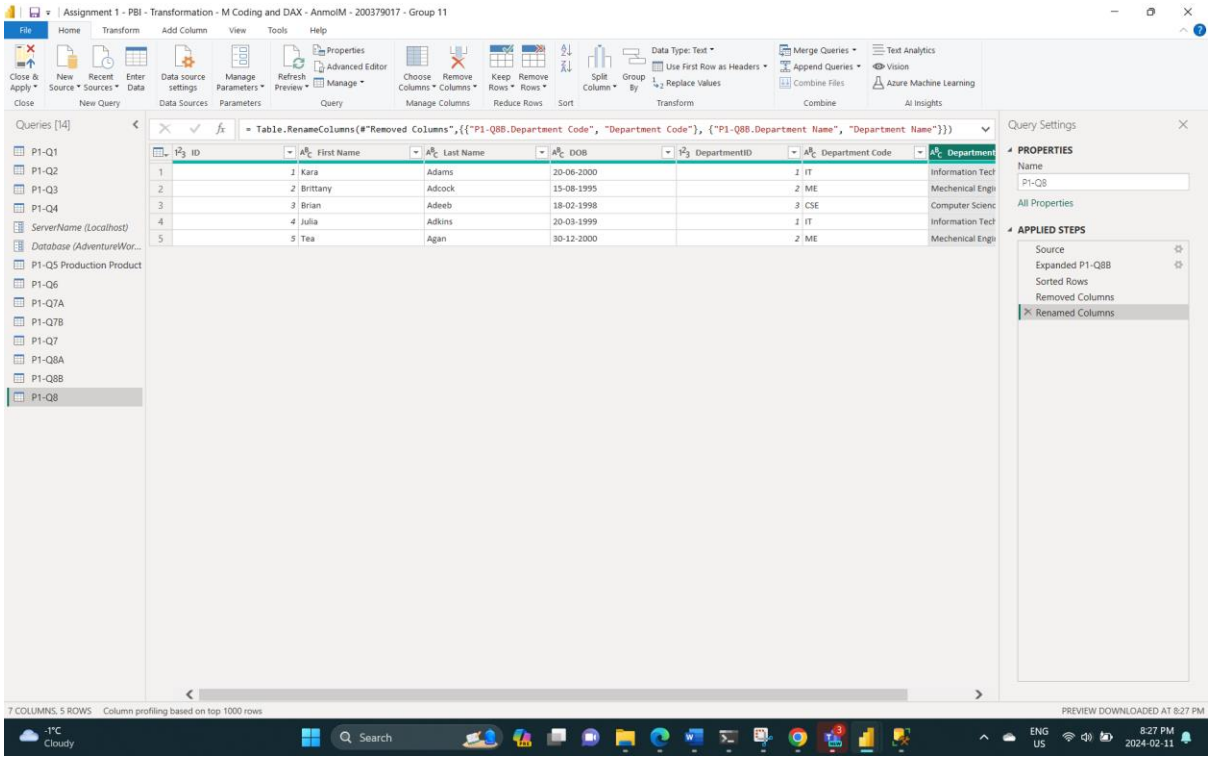
Select Merge Queries and select the tables we want to merge and the common column we have in both in order to join. We select Left join, as this function keeps all the info from our main table and joins the department info from our second table to the available rows.



After this step, the merged table is created as follows. Expand the columns as needed.



After removing unneeded columns or duplicate columns, the table should look as follows:



9. Download “2020-monthly-visitor-statistics.xlsx” file from the link given below. After downloading the file, load the “Days by Island” sheet in Power BI Desktop and apply Power Query transformation to extract a part of the data (highlighted in the dataset snapshot).

Note: Download “2020-monthly-visitor-statistics.xlsx” file from the below URL: https://www.hawaiitourismauthority.org/media/7901/2020-monthly-visitor-statistics.xlsx

Input Dataset: -

2020 Visitor Days by Island and Month (Arrivals by Air)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	TOTAL
O'ahu	36,47,908	31,06,258	17,19,457	92,974	1,91,462	3,11,721	3,56,088	4,55,805	3,18,122	5,22,327	8,63,638	12,43,871	1,28,29,630
Maui	21,97,687	19,62,660	10,57,111	13,195	31,900	54,141	62,432	66,381	87,807	3,13,046	7,12,026	9,94,709	75,53,095
Moloka'i	35,919	28,029	14,135	212	1,922	2,796	3,688	1,434	2,592	7,359	8,247	11,615	1,17,947
Lāna'i	25,108	23,405	9,158	34	178	1,073	1,365	1,033	711	6,084	6,622	8,457	83,228
Kaua'i	9,20,439	8,37,629	4,42,047	5,960	17,333	30,915	39,219	46,169	44,253	1,59,765	3,22,018	74,607	29,40,354
Hawai'i Island	14,45,673	11,63,151	6,66,084	19,249	40,412	72,483	1,04,803	1,30,566	1,60,688	2,13,822	3,73,547	6,02,066	49,92,542

Notes: monthly data may not add up to total due to rounding

2020 Visitor Arrivals by Island and Month (Arrivals by Air)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	TOTAL
O'ahu	5,06,708	4,67,959	2,36,640	3,601	7,365	13,029	16,448	16,917	11,885	36,009	76,662	1,13,090	15,06,316
Maui	2,43,086	2,34,823	1,25,353	624	1,131	1,988	2,568	2,453	2,479	23,178	63,748	91,171	7,92,602
Moloka'i	6,858	5,089	2,384	31	76	114	225	109	75	375	640	1,050	17,025
Lāna'i	6,066	6,146	2,604	21	26	64	121	81	39	595	904	1,257	17,924
Kaua'i	1,13,796	1,10,478	56,725	306	603	1,068	1,349	1,342	1,096	11,249	28,487	3,762	3,30,263
Hawai'i Island	1,65,297	1,48,204	77,933	701	1,300	2,605	3,608	3,683	3,642	10,641	28,056	48,147	4,93,817

2020 Visitor Average Length of Stay by Island and Month (Arrivals by Air)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	TOTAL
O'ahu	7.20	6.64	7.27	25.82	26.00	23.92	21.65	26.94	26.77	14.51	11.27	11.00	8.52
Maui	9.04	8.36	8.43	21.13	28.22	27.23	24.31	27.06	35.42	13.51	11.17	10.91	9.53
Moloka'i	5.24	5.51	5.93	6.94	25.37	24.49	16.42	13.19	34.45	19.62	12.89	11.06	6.93

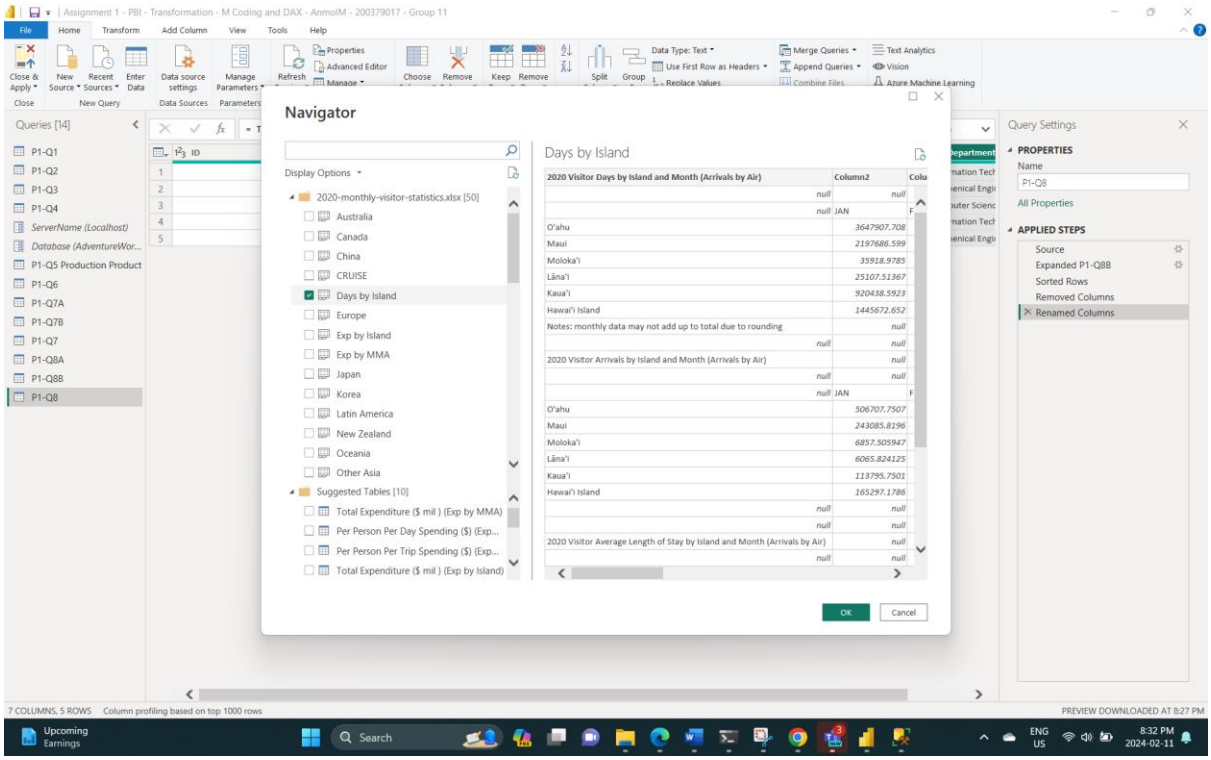
... | China | Korea | Taiwan | Europe | Latin America | Exp by MMA | Exp by Island | Days by Island | CRUISE

Expected Output: -

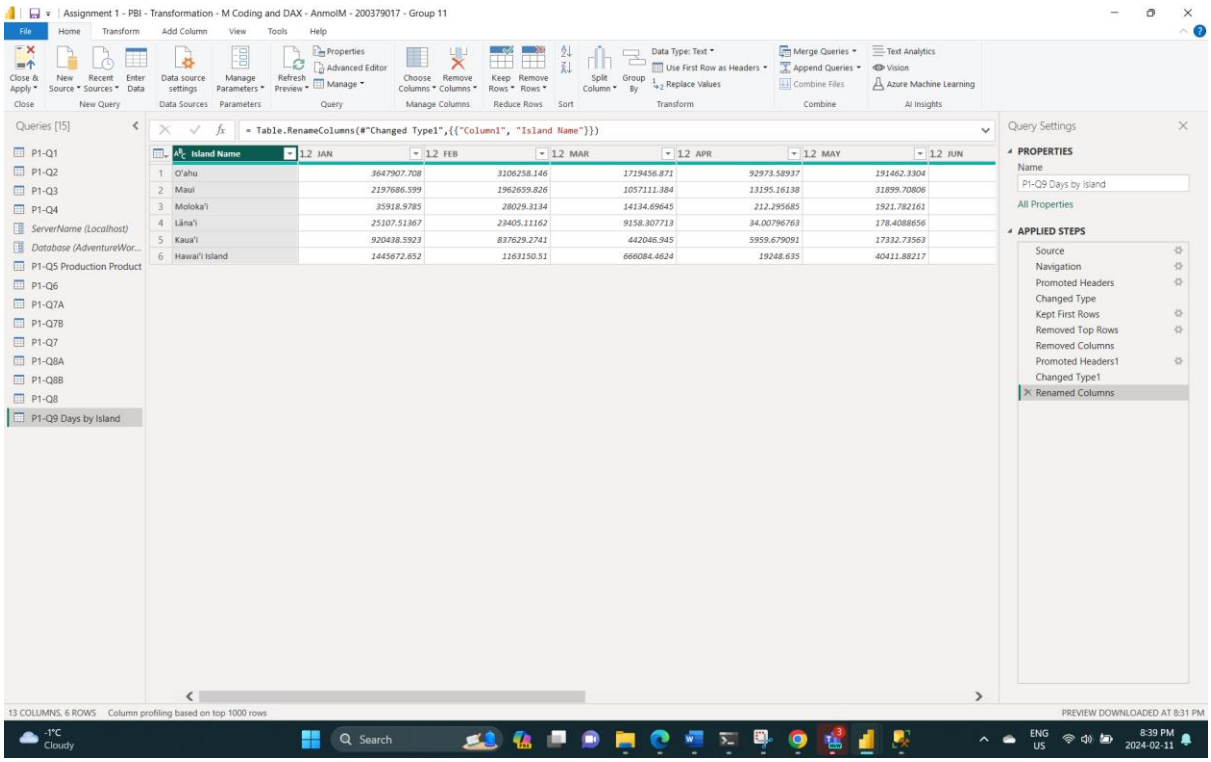
	Island Name	Month	Number of Visitors
1	O'ahu	JAN	3647908
2	O'ahu	FEB	3106258
3	O'ahu	MAR	1719457
4	O'ahu	APR	92974
5	O'ahu	MAY	191462
6	O'ahu	JUN	311721
7	O'ahu	JUL	356088
8	O'ahu	AUG	455805
9	O'ahu	SEP	318122
10	O'ahu	OCT	522327
11	O'ahu	NOV	863638
12	O'ahu	DEC	1243871
13	Maui	JAN	2197687
14	Maui	FEB	1962660
15	Maui	MAR	1057111
16	Maui	APR	13195
17	Maui	MAY	31900
18	Maui	JUN	54141
19	Maui	JUL	62432
20	Maui	AUG	66381
21	Maui	SEP	87807
22	Maui	OCT	313046
23	Maui	NOV	712026
24	Maui	DEC	994709
25	Moloka'i	JAN	35919
26	Moloka'i	FEB	28029
27	Moloka'i	MAR	14135
28	Moloka'i	APR	212

Answer:

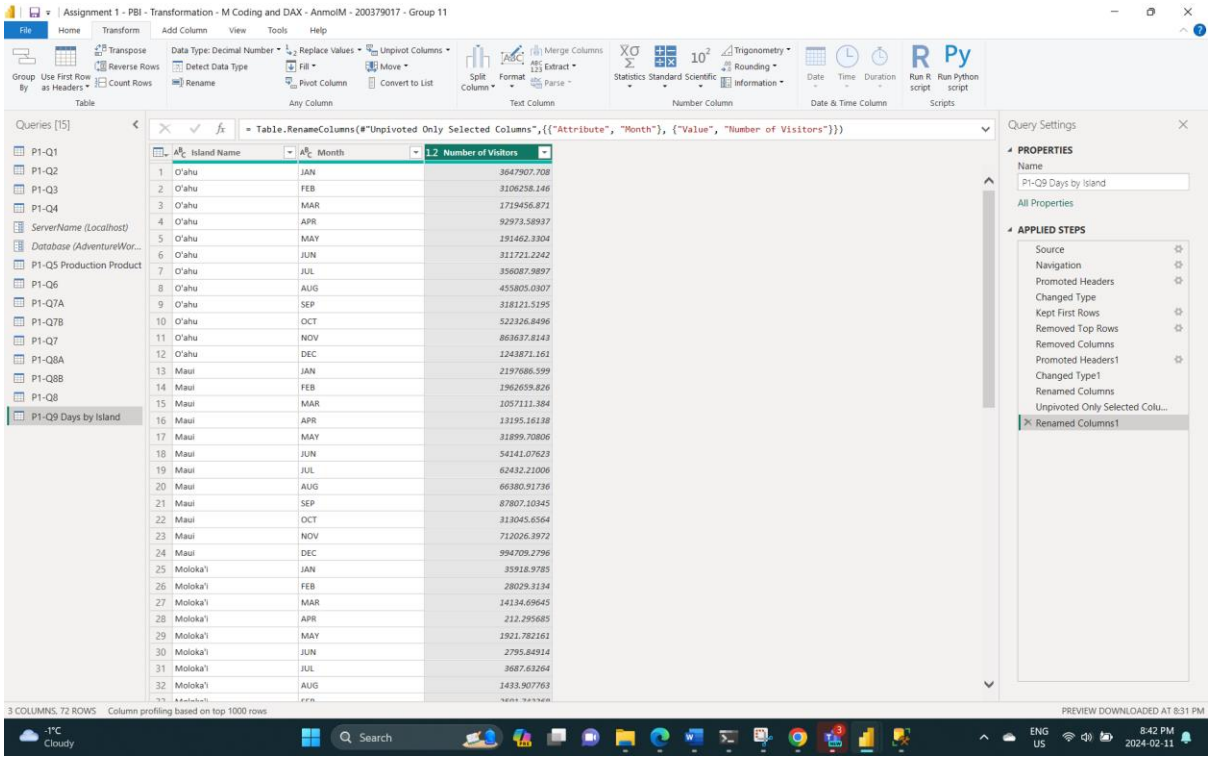
Import the required sheet and transform the table to keep first 8 rows only, then remove the first empty row and the last 'total' column. Then we make the first row with month names our headers and edit first columns name to 'Island Name'



After this, our table should look like below:



Then we select all the month columns and unpivot columns. This brings us to our required output. Rename columns as needed.



10. Transform the below dataset using Power Query transformation.

Input Dataset: -

Col1	Col2	Col3	Col4	Col5	Col6	Col7
	2020			2021		
	Technology	Office Supplies	Furniture	Technology	Office Supplies	Furniture
Jan	433.2	255.4	2.6	4.7	93.9	122.8
Feb	435.0	229.1	2.1	5.7	83.4	97.4
Mar	409.3	230.7	1.9	6.3	74.2	89.2
Apr	377.6	209.2	2.0	4.4	87.9	86.4
May	403.4	226.0	1.2	5.8	86.3	83.4
Jun	471.8	260.0	2.2	3.8	106.5	121.9
Jul	540.5	272.1	2.4	3.4	106.5	116.5
Aug	485.5	243.6	2.4	3.4	100.3	120.3
Sep	432.8	183.4	1.7	3.3	83.7	95.6
Oct	442.7	234.1	2.3	4.6	88.1	108.9
Nov	419.6	197.8	2.1	4.8	80.9	116.5
Dec	532.8	260.0	2.5	6.4	96.2	133.5

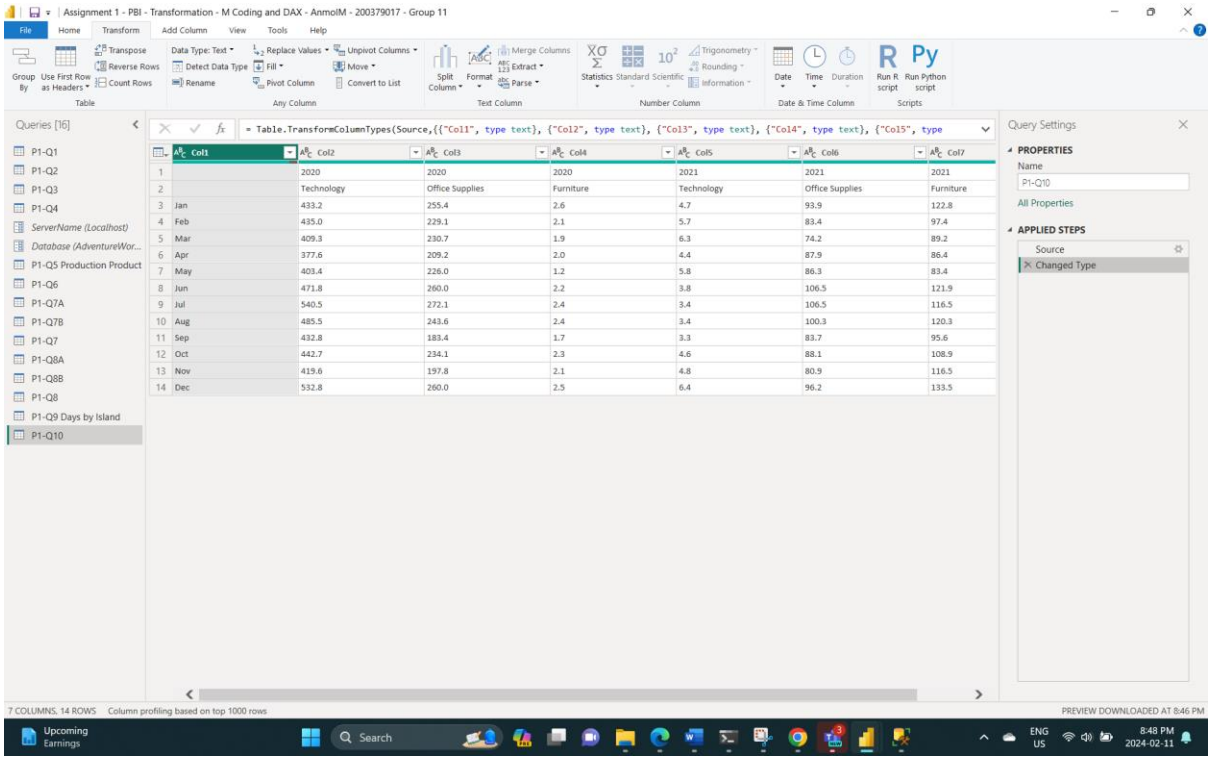
Note: Use “Enter Data” option to create the above table in Power BI.

Expected Output: -

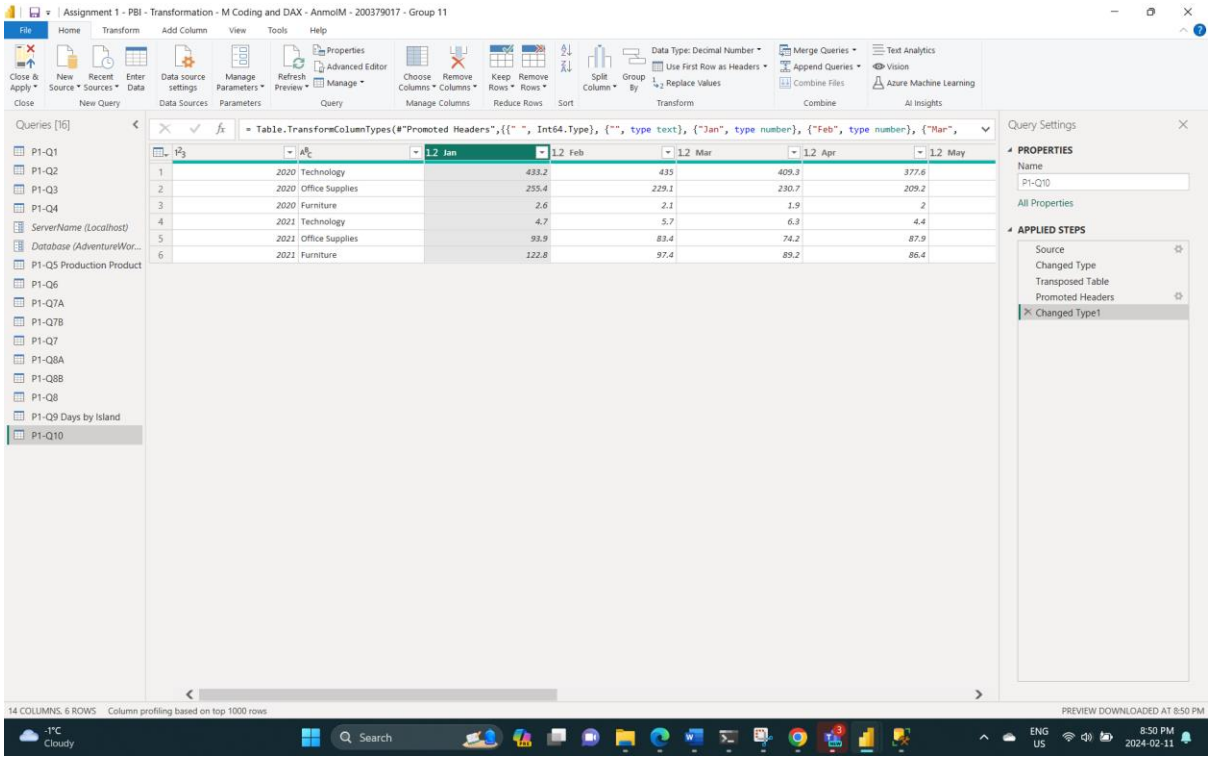
	Year	Product	Month	Sales
1	2020	Technology	Jan	433.2
2	2020	Technology	Feb	435
3	2020	Technology	Mar	409.3
4	2020	Technology	Apr	377.6
5	2020	Technology	May	403.4
6	2020	Technology	Jun	471.8
7	2020	Technology	Jul	540.5
8	2020	Technology	Aug	485.5
9	2020	Technology	Sep	432.8
10	2020	Technology	Oct	442.7
11	2020	Technology	Nov	419.6
12	2020	Technology	Dec	532.8
13	2020	Office Supplies	Jan	255.4
14	2020	Office Supplies	Feb	229.1
15	2020	Office Supplies	Mar	230.7
16	2020	Office Supplies	Apr	209.2
17	2020	Office Supplies	May	226
18	2020	Office Supplies	Jun	260
19	2020	Office Supplies	Jul	272.1
20	2020	Office Supplies	Aug	243.6
21	2020	Office Supplies	Sep	183.4
22	2020	Office Supplies	Oct	234.1
23	2020	Office Supplies	Nov	197.8
24	2020	Office Supplies	Dec	260
25	2020	Furniture	Jan	2.6

Answer:

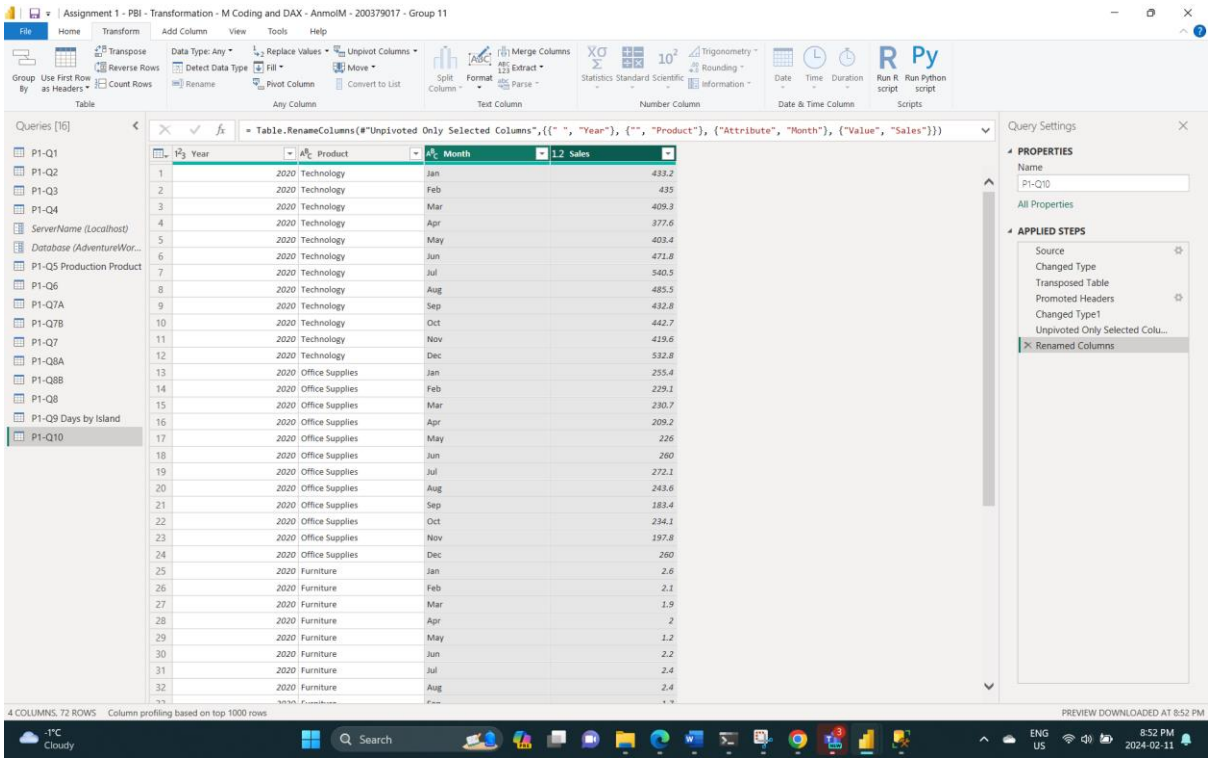
We input our data to make the table below:



Then we transpose the columns and make the first row as headers:

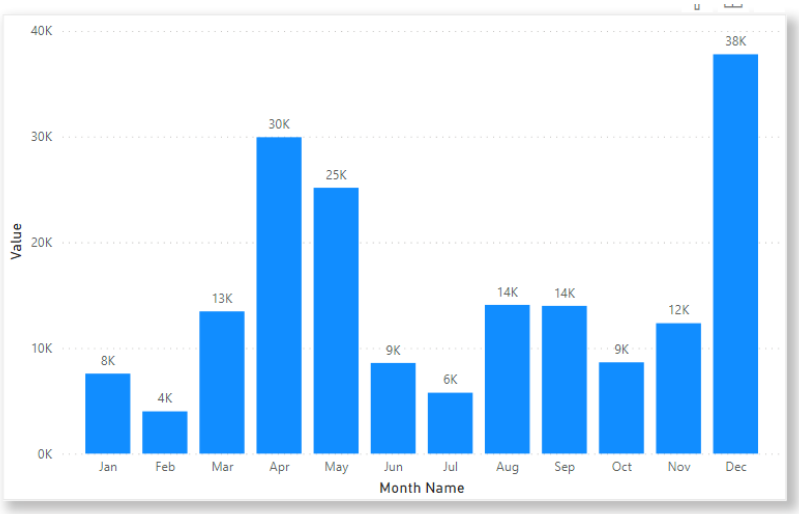


Now we select the month columns and unpivot to get the desired output. Rename the columns as needed.



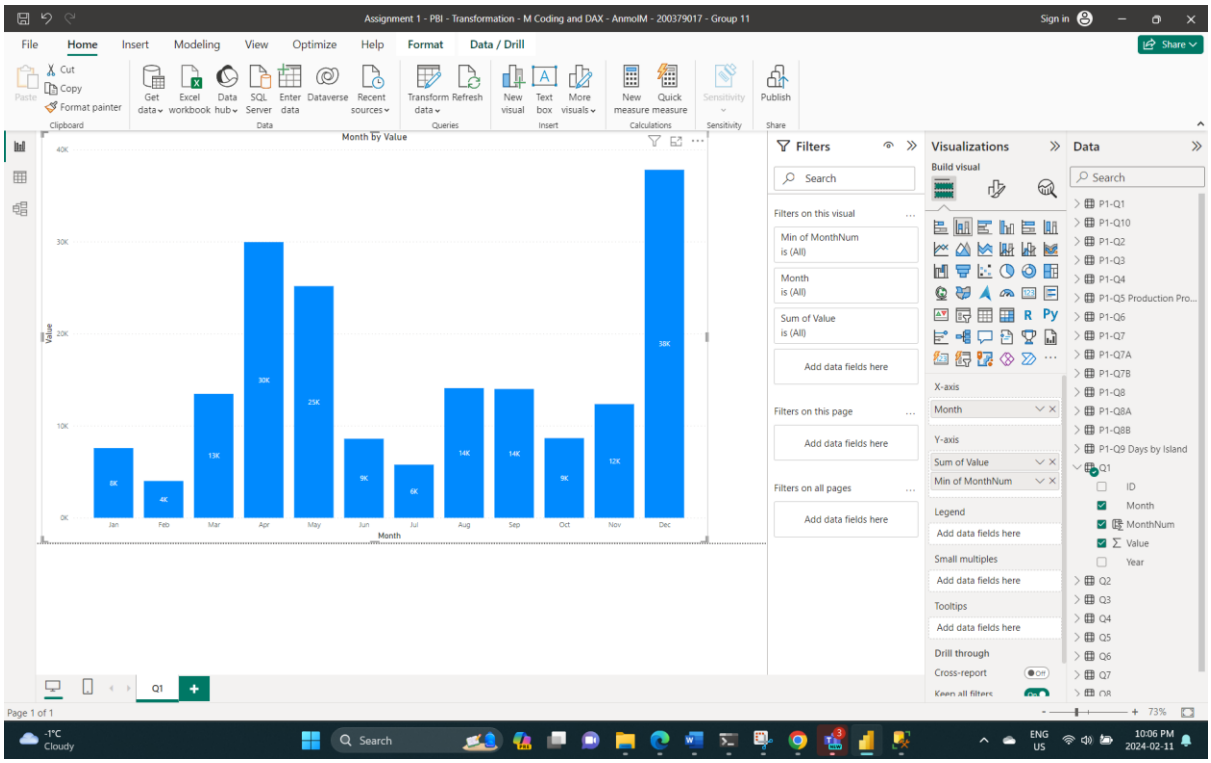
Part 2: DAX

1. Sort the month name as shown in the snapshot using Power BI DAX. [Use Q1 sheet from DAX_Data.xlsx file].

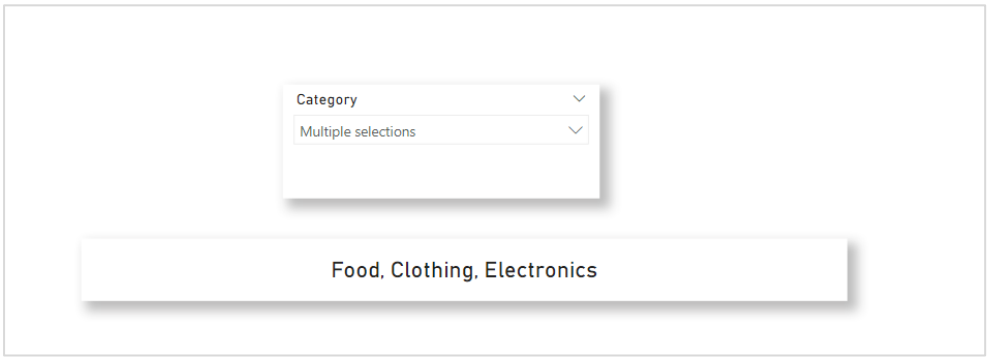


Answer:

Create a new column MonthNum using DAX that gives the number for each Month name. and use it to sort the month names in the chart.

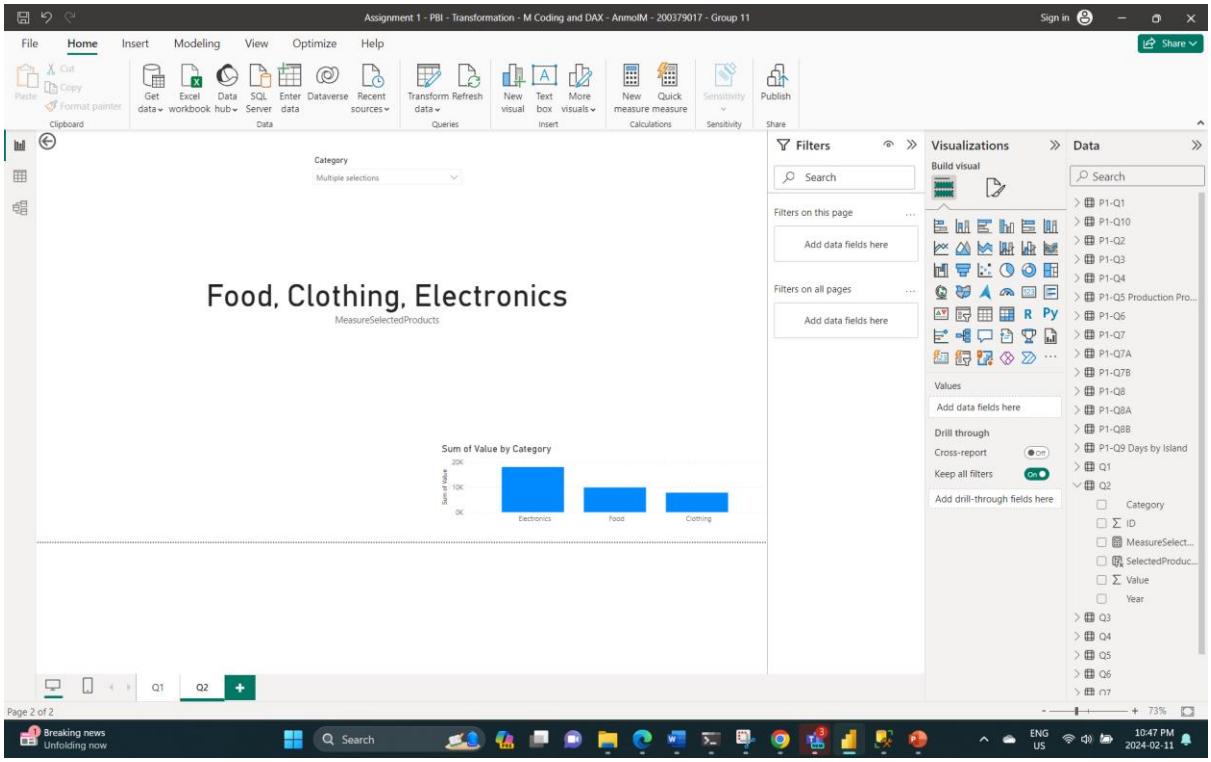


2. Capture the Values selected from the slicer, if nothing is selected then show "All". Use card visuals to display the value. [Use Q2 sheet from DAX_Data.xlsx].



Answer:

We use ISFILTERED, IF, NOT and CONCATENATEX functions to concatenate and present all items that are selected by the filter.

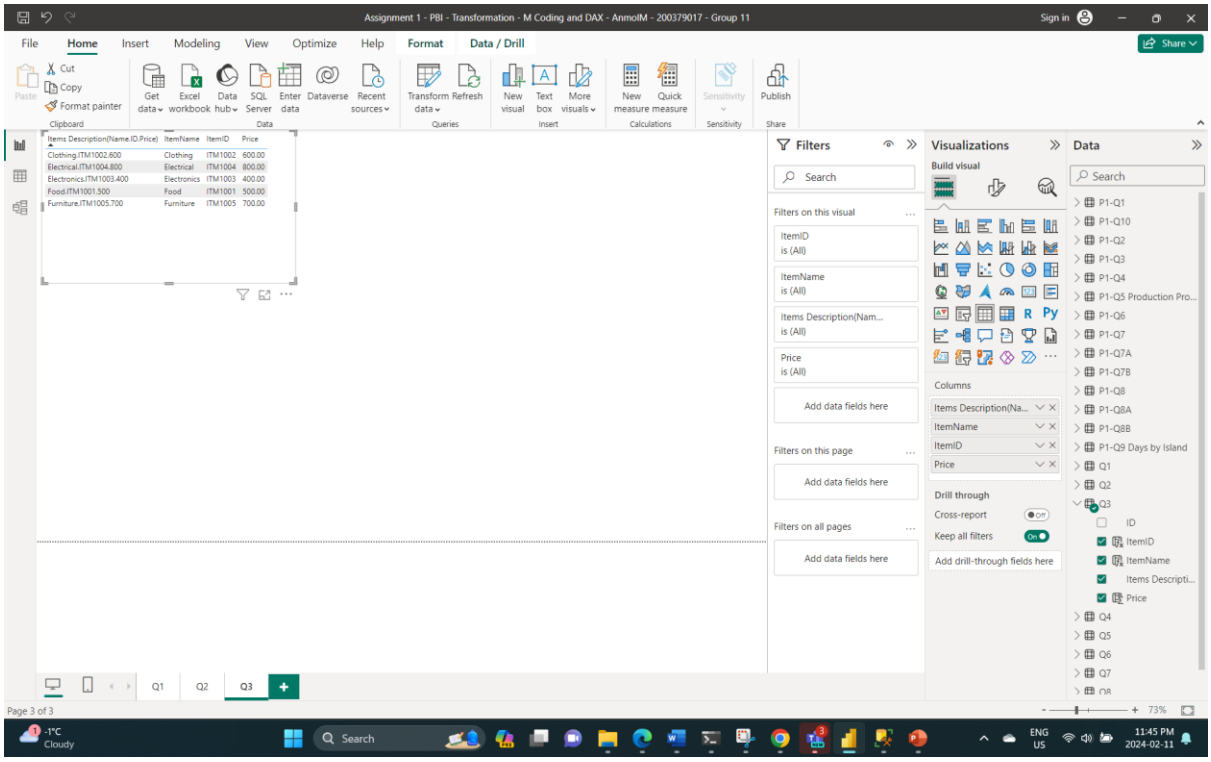


3. Use the DAX function to extract the Item name, Item ID, and Price from the "Item Description" column (which contains a combination of Item Name, Item ID, and price). [Use Q3 sheet from DAX_Data.xlsx].

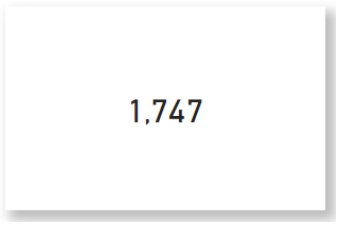
Items Description(Name.ID.Price)	Item Name	Item Id	Price
Clothing.ITM1002.600	Clothing	ITM1002	600
Electrical.ITM1004.800	Electrical	ITM1004	800
Electronics.ITM1003.400	Electronics	ITM1003	400
Food.ITM1001.500	Food	ITM1001	500
Furniture.ITM1005.700	Furniture	ITM1005	700

Answer:

We use LEFT, MID, RIGHT and FIND functions to extract the text based on the position of the delimeter.

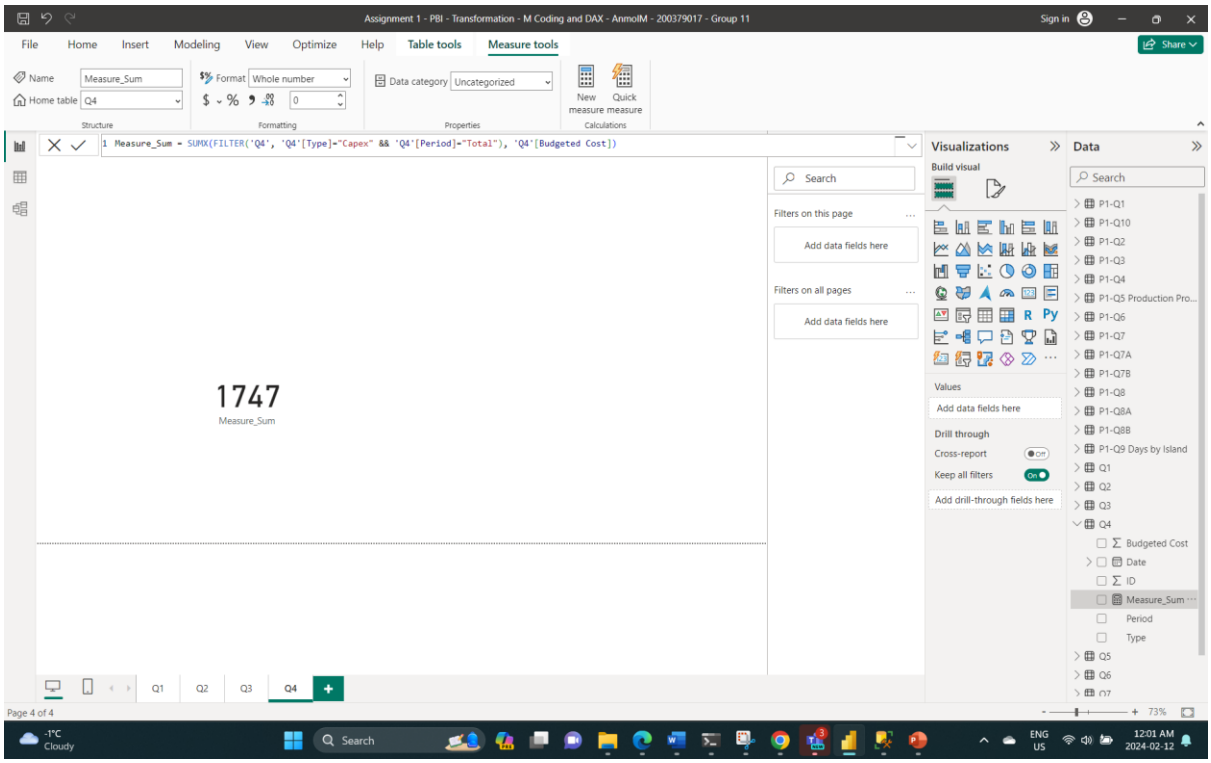


4. Write a DAX function to calculate sum of Budget cost where [Type] = Capex and [Period] = Total. [Use Q4 sheet from DAX_Data.xlsx].



Answer:

We use the SUMX function along with FILTER to filter the vales from the table to only when Budget Cost is ‘Capex’ and Period is ‘Total’.



5. Calculate the MAX of the number after multiplying with a constant value and put this into a column say as MAX_Value, use the formula for the calculation as shown below-

MAX_Value = MAXX([M] * 1750, [W]* 1, [V] * 330)

Note: S, M, W, and V are the column name. [Use Q5 sheet from DAX_Data.xlsx].

Input Data:

S	M	W	V
1	15	500	60
2	10	60000	2
3	12.5	284000	300
4	5	4200	360
5	10	40250	240

Expected Result

M	W	V	Max_Value
26250	500	19800	26250
17500	60000	660	60000
21875	284000	99000	284000
8750	4200	118800	118800
17500	40250	79200	79200

Answer:

6. The table is having item name column, add two columns based on the following value, 1st column contains a value based on the distinct item and 2nd column contains a value based on the item by skipping the row if there is a tie as shown in the screenshot below. [Use Q6 sheet from DAX_Data.xlsx].

Input Data:

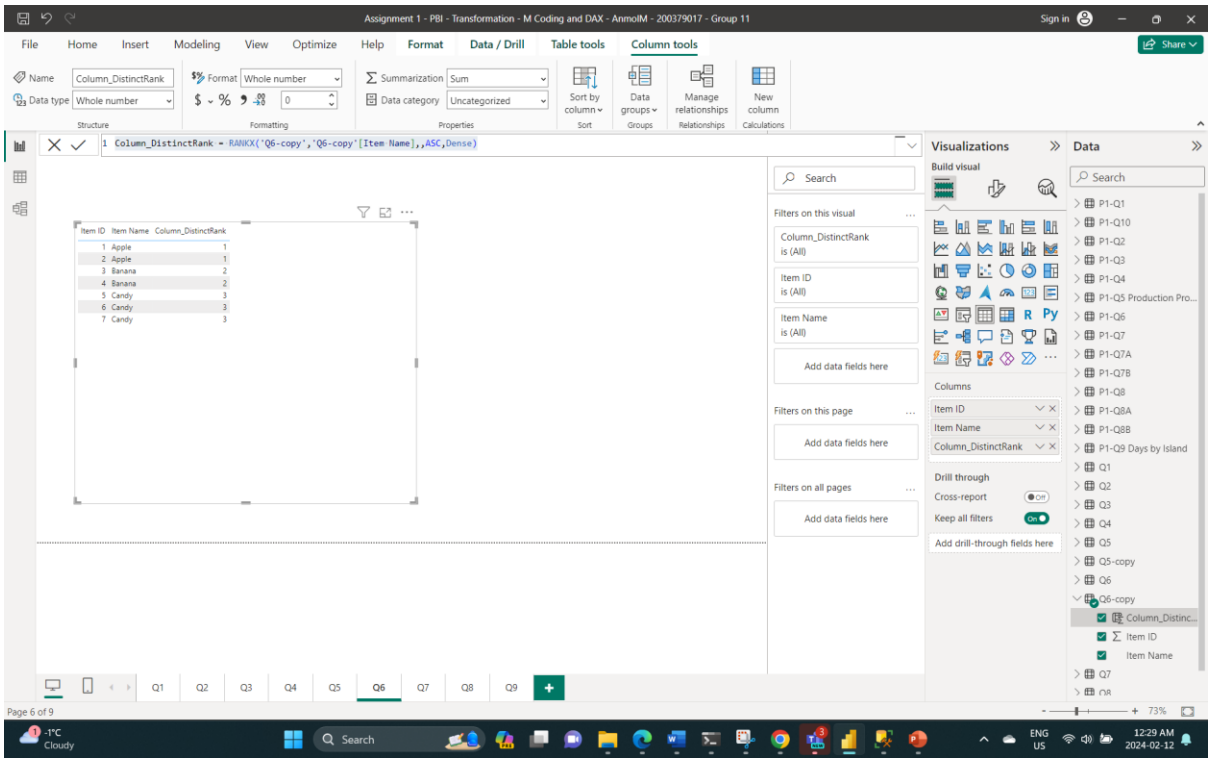
Item ID	Item Name
1	Apple
2	Apple
3	Banana
4	Banana
5	Candy
6	Candy
7	Candy

Expected Result

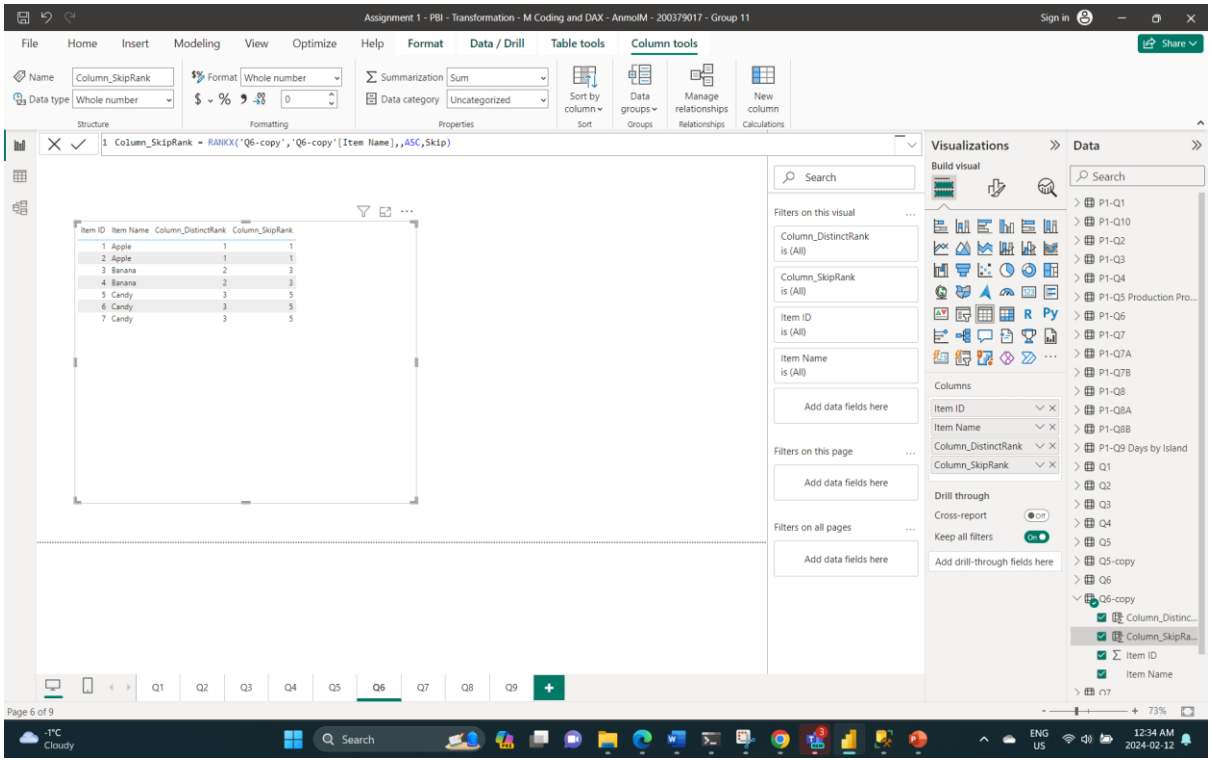
Item ID	Item Name	ID1	ID2
1	Apple	1	1
2	Apple	1	1
3	Banana	2	3
4	Banana	2	3
5	Candy	3	5
6	Candy	3	5
7	Candy	3	5

Answer:

We add 1st column using RANKX function using Dense Ties. This column contains value for every distinct item.



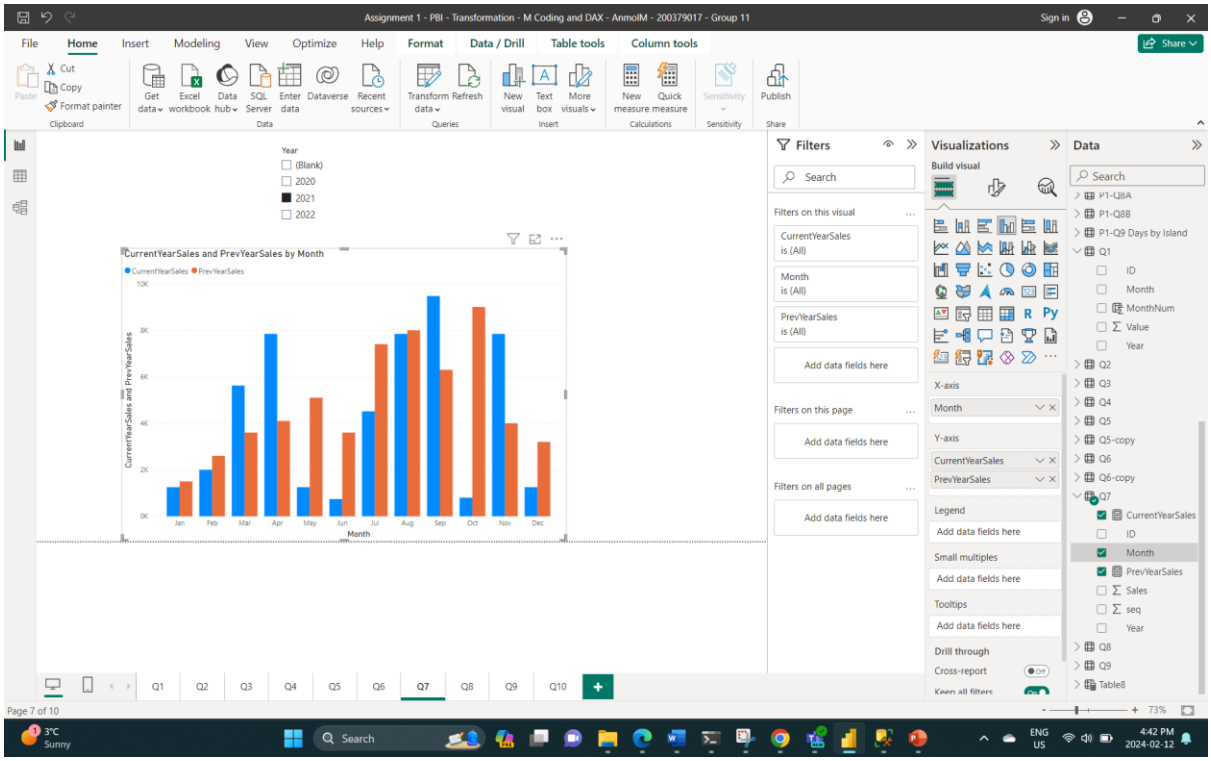
Now we add the second column using RANKX function but we change Ties to Skip this time, to get our desired results that is skipping a column in case of a tie.



7. Create a bar graph as shown below with previous year and present year sales month on month basis. [Use Q7 sheet from DAX_Data.xlsx].

Answer:

We create two new measures that calculate sum of the selected year and then the selected year – 1 to give us current year sales and previous year sales.

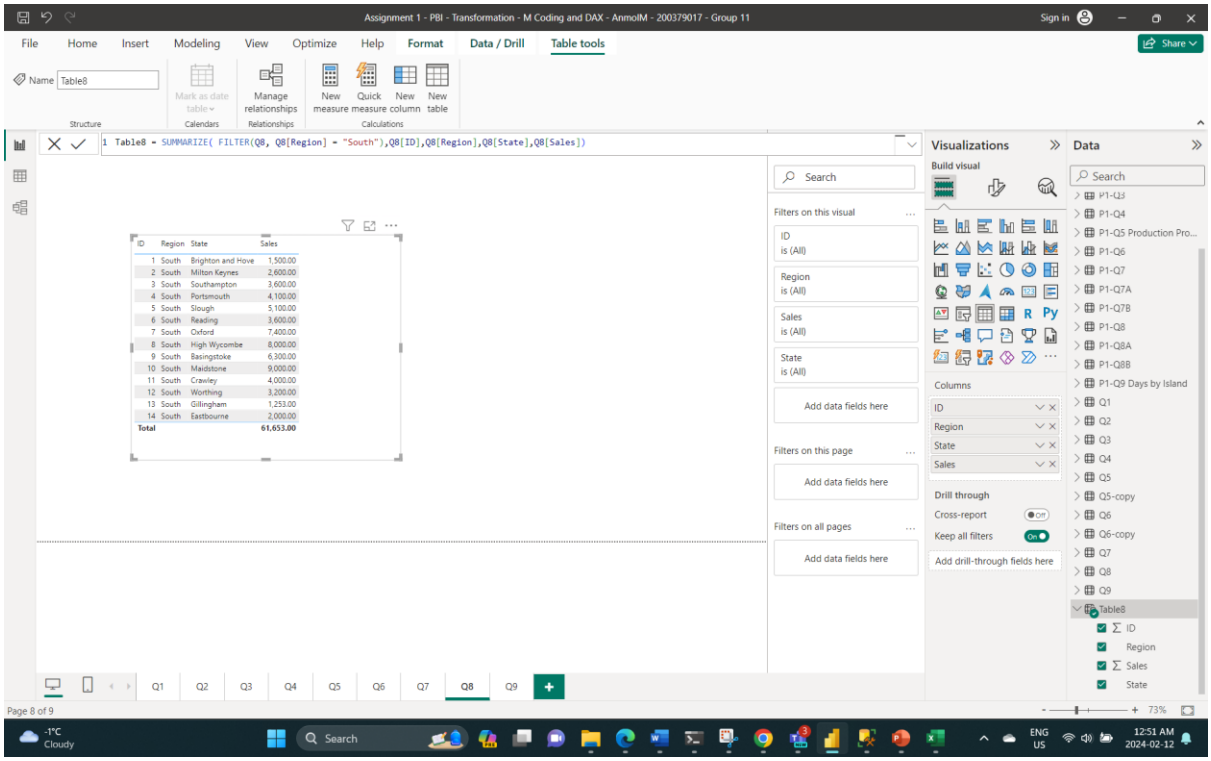


8. Write a DAX function to create a filter (Region= “South”) set of rows from an existing table to a new table as shown in the snapshot below. [Use Q9 sheet from DAX_Data.xlsx].

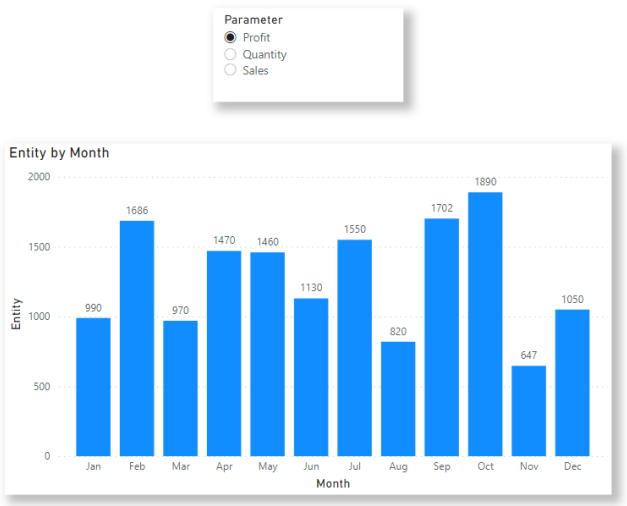
ID	Region	State	Sales
1	South	Brighton and Hove	1500
2	South	Milton Keynes	2600
3	South	Southampton	3600
4	South	Portsmouth	4100
5	South	Slough	5100
6	South	Reading	3600
7	South	Oxford	7400
8	South	High Wycombe	8000
9	South	Basingstoke	6300
10	South	Maidstone	9000
11	South	Crawley	4000
12	South	Worthing	3200
13	South	Gillingham	1253
14	South	Eastbourne	2000
Total			**61653**

Answer:

We use the summarize function to create a new table from the existing table and use Filter to make the region ‘South’.

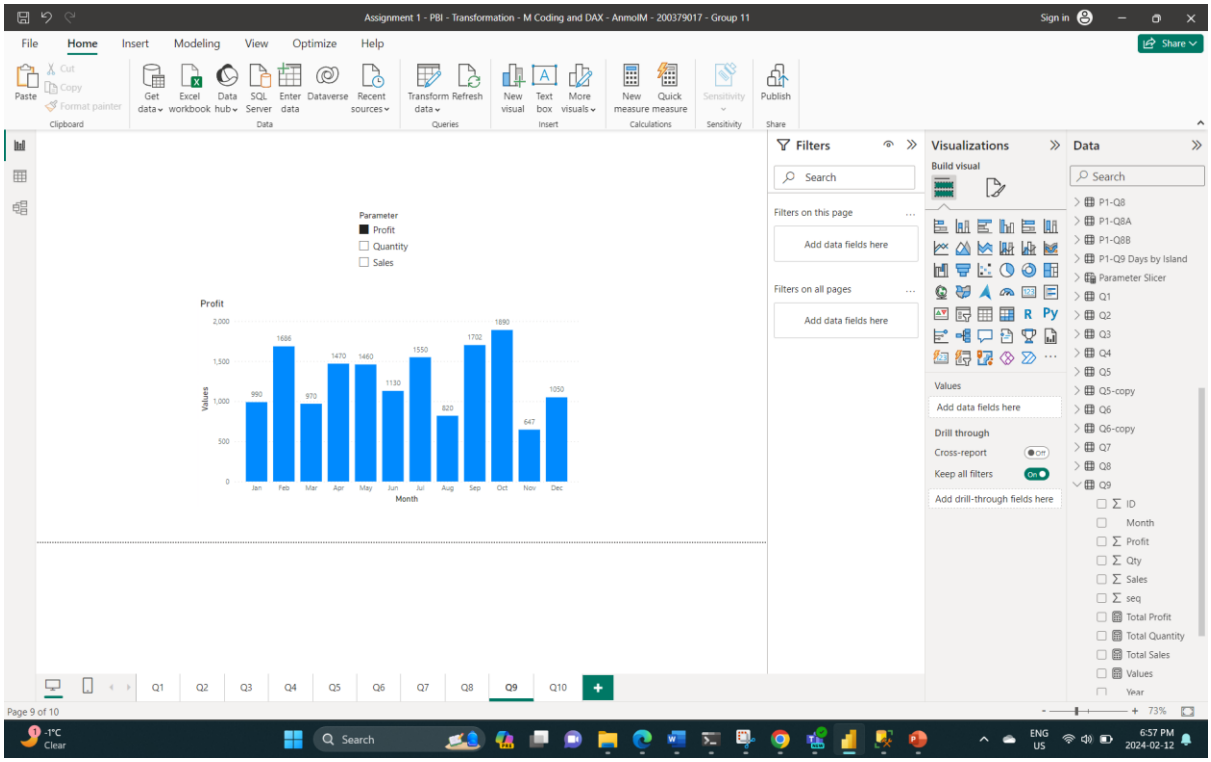


9. Create a bar graph as shown below with swapping axis as profit, sales, or quantity on a parameter(slicer). [Use Q9 sheet from DAX_Data.xlsx].



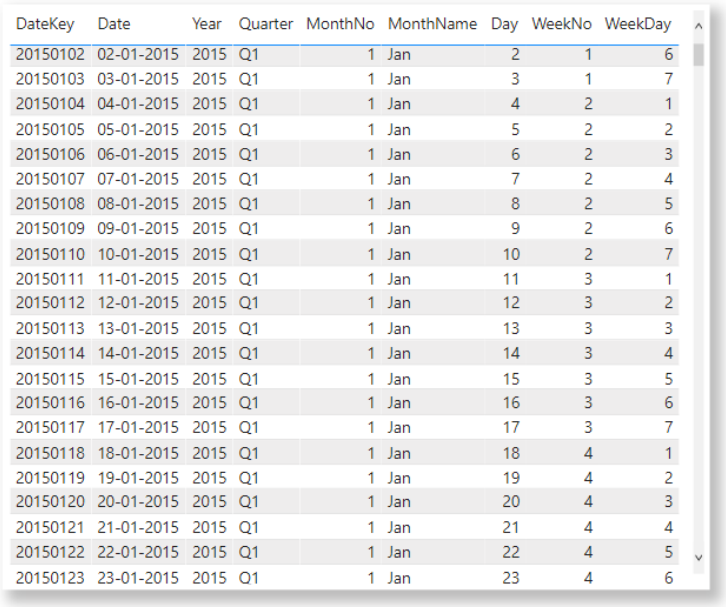
Answer:

We create sum/total measures for each entity to take data from Sales, Profit and Qty columns, then create an index table which we use as our parameter slicer. Then we create a new measure that use SWITCH function to switch index values from our slicer table with our SUM/total measures which we use to plot our chart as shown below.



10. Generate a calendar table that has the following columns-

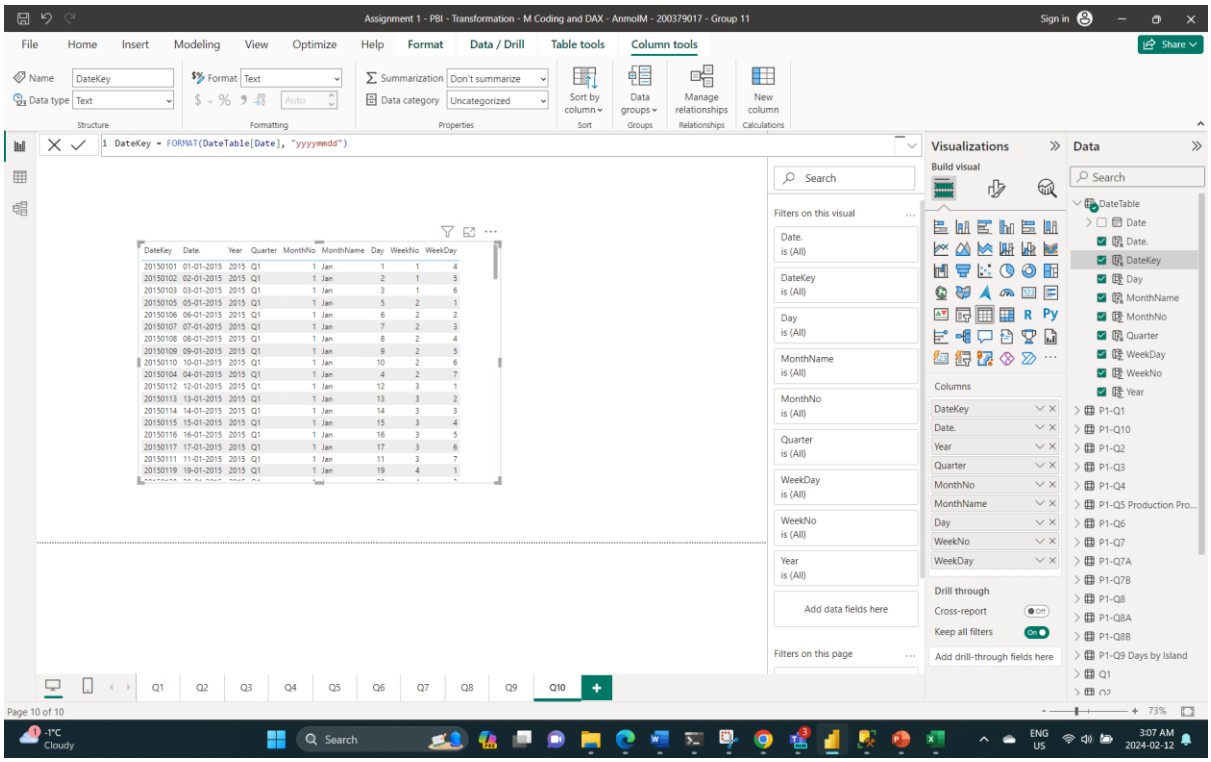
Date: containing a date value
DateKey: containing a date in integer format, ex- 26-05-2022→ 20220626
Year: containing a year value from date
MonthNo: containing a month number from date
MonthName: containing month name from date
Day: containing day from date
Quarter: containing quarter from date
WeekNo: containing week no from date
WeekDay: containing week day from day



DateKey	Date	Year	Quarter	MonthNo	MonthName	Day	WeekNo	WeekDay
20150102	02-01-2015	2015	Q1	1	Jan	2	1	6
20150103	03-01-2015	2015	Q1	1	Jan	3	1	7
20150104	04-01-2015	2015	Q1	1	Jan	4	2	1
20150105	05-01-2015	2015	Q1	1	Jan	5	2	2
20150106	06-01-2015	2015	Q1	1	Jan	6	2	3
20150107	07-01-2015	2015	Q1	1	Jan	7	2	4
20150108	08-01-2015	2015	Q1	1	Jan	8	2	5
20150109	09-01-2015	2015	Q1	1	Jan	9	2	6
20150110	10-01-2015	2015	Q1	1	Jan	10	2	7
20150111	11-01-2015	2015	Q1	1	Jan	11	3	1
20150112	12-01-2015	2015	Q1	1	Jan	12	3	2
20150113	13-01-2015	2015	Q1	1	Jan	13	3	3
20150114	14-01-2015	2015	Q1	1	Jan	14	3	4
20150115	15-01-2015	2015	Q1	1	Jan	15	3	5
20150116	16-01-2015	2015	Q1	1	Jan	16	3	6
20150117	17-01-2015	2015	Q1	1	Jan	17	3	7
20150118	18-01-2015	2015	Q1	1	Jan	18	4	1
20150119	19-01-2015	2015	Q1	1	Jan	19	4	2
20150120	20-01-2015	2015	Q1	1	Jan	20	4	3
20150121	21-01-2015	2015	Q1	1	Jan	21	4	4
20150122	22-01-2015	2015	Q1	1	Jan	22	4	5
20150123	23-01-2015	2015	Q1	1	Jan	23	4	6

Answer:

We use YEAR, QUARTER, MONTH, WEEKDAY, WEEKNUM, DAY and FORMAT functions to create the Calendar table as required.



Achievement:

With this assignment, we learned various functions of M-Code that enabled me to understand how to add index columns, conditional columns, custom columns and adding increments to index columns. We also learn how to split columns by delimiters, append queries, merge queries, pivot-unpivot and transpose tables and add parameters. We also receive a thorough learning exercise to practice PowerBI DAX and understand various DAX functions like IF, NOT, CONCATENATE, ISFILTERED, LEFT, MID, RIGHT, SUMX, MAXX, various functions related to date. Although this assignment was lengthy, it was a good exercise to practice and improve my PowerBI skills. Especially, Q7 and Q9 from DAX were highly helpful in making me aware of new PowerBI functions.