

Power BI - Data Transformation - p3

Power BI - Data Transformation - p3

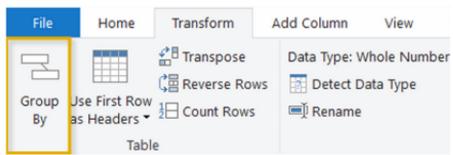
The screenshot shows the Power BI Data Transformation interface. On the left, there's a 'Queries [6]' pane with several items listed, including 'Sales Data 2020' which is currently selected. The main area displays a table named 'Index' with columns: OrderDate, StockDate, OrderNumber, ProductKey, CustomerKey, and Type. A tooltip for the 'Type' column indicates it is a 'Text' type. Below the table, a histogram shows 1000 distinct values. An 'Append' dialog box is open, asking to 'Concatenate rows from three or more tables into a single table.' The 'Three or more tables' option is selected. Under 'Available tables', 'Sales Data 2020', 'Sales Data 2021', and 'Sales Data 2022' are listed. A button labeled 'Add >>' is present. To the right of the dialog, a vertical list of unique values for the 'Type' column is shown, ranging from 14657 down to 29174. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

The screenshot shows the 'Data' pane on the left side of the Power BI interface. It includes a search bar and a list of available tables: 'Calendar Lookup', 'Customer Lookup', 'Product Lookup', 'Sales Data 2020', 'Sales Data 2021', 'Sales Data 2022', and 'Sales Record 2020-2...'. Ellipsis (...) and 'More' buttons are also visible.

Aggregation : Min , Max , Count , Sum , Average , etc..... Department wise - Group

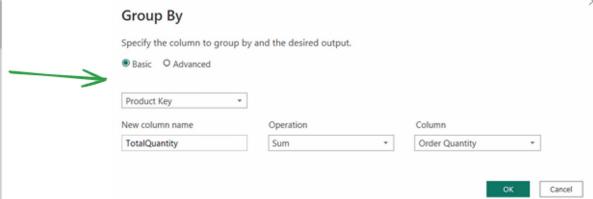


GROUPING & AGGREGATING



Group By allows you to aggregate data at a different level (i.e. group daily records into monthly, aggregate transactions by store, etc.)

	Order Date	Product Key	Customer Key	Order Quantity
1	6/25/2022	214	14719	1
2	10/8/2021	214	21990	1
3	12/30/2021	214	22098	1
4	6/29/2022	214	22748	1
5	8/16/2021	214	27821	1
6	10/9/2021	214	15685	1
7	8/9/2021	214	14951	1
8	1/19/2022	214	23101	1
9	9/23/2021	214	17158	1
10	1/19/2022	214	24296	1
11	6/29/2022	214	12963	1
12	9/13/2021	214	12715	1
13	10/2/2021	214	14846	1
14	7/31/2021	214	11290	1
15	11/24/2021	214	22103	1
16	8/1/2021	214	16982	1
17	10/12/2021	214	20410	1
18	9/10/2021	214	14217	1
19	10/22/2021	214	19642	1
20	8/11/2021	214	11666	1



	Product Key	TotalQuantity
1	214	2099
2	215	1940
3	220	1995
4	223	4151
5	226	392
6	229	408
7	232	424
8	235	381
9	310	169
10	311	139
11	312	179
12	313	168
13	314	157
14	320	65
15	322	39
16	324	72
17	326	65

Here we're transforming a daily, transaction-level table into a summary of Total Quantity by Product Key

NOTE: Any fields not specified in the Group By settings are lost

The screenshot shows a table of daily transaction data with columns: Order Date, Product Key, Customer Key, and Order Quantity. To the right is a 'Group By' dialog box where 'Advanced' mode is selected. It shows grouping by Product Key and Customer Key, with an aggregation step where 'TotalQuantity' is summed from 'Order Quantity'. A green arrow points from the 'Group By' dialog to the 'TotalQuantity' row in the table.

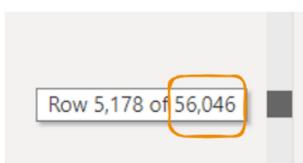
	Order Date	Product Key	Customer Key	Order Quantity
1	6/25/2022	214	14719	1
2	10/6/2021	214	21990	1
3	12/30/2021	214	22098	1
4	6/29/2022	214	22748	1
5	8/16/2021	214	27821	1
6	10/9/2021	214	15685	1
7	8/9/2021	214	14951	1
8	1/19/2022	214	23101	1
9	9/23/2021	214	17158	1
10	1/19/2022	214	24196	1
11	6/29/2022	214	12963	1
12	9/13/2021	214	12715	1
13	10/2/2021	214	14846	1
14	7/31/2021	214	11290	1
15	11/24/2021	214	22103	1
16	8/1/2021	214	16982	1
17	10/12/2021	214	20410	1
18	9/10/2021	214	14217	1
19	10/22/2021	214	19642	1
20	8/11/2021	214	11666	1

The screenshot shows a summary table with columns: Product Key, Customer Key, and TotalQuantity. The data is identical to the original table above, but it represents a grouped summary. A green arrow points from the 'TotalQuantity' column in the 'Group By' dialog to the 'TotalQuantity' column in this table.

	Product Key	Customer Key	TotalQuantity
1	214	19356	1
2	214	15101	1
3	214	12473	1
4	214	12963	1
5	214	26986	1
6	214	13202	1
7	214	14951	1
8	214	11201	1
9	214	19538	1
10	214	22749	1
11	214	15815	1
12	214	19252	1
13	214	14849	1
14	214	11290	1
15	214	27851	1
16	214	16982	1
17	214	21863	1
18	214	19725	1
19	214	15684	1
20	214	11666	1
21	214	26941	1

This time we're transforming the daily, transaction-level table into a summary of Total Quantity grouped by both Product Key and Customer Key (using the "Advanced" option)

NOTE: This is like creating a PivotTable in Excel and pulling in Sum of Order Quantity with Product Key and Customer Key as row labels



Grouping done using

- Aggregation on Next Column :
Sum of Order Quantit.u.

Grouping done using Product Key

Next Column : Sum of Order Quantity.

Total order Qty.

The screenshot shows the Power BI ribbon with the 'Transform' tab selected. Below it, the 'Group By' dialog is open. In the 'Group By' dialog, 'ProductKey' is selected for grouping. A new column 'Total Order Quantity' is being created with the operation 'Sum'. The 'Column' dropdown shows various columns from the current table, including 'OrderQuantity'. To the right, a preview pane shows the transformed data with a column labeled '1.2 Total Order Quantity'.

Existing Column on a Selected Table.

Whatever aggregation you want to perform . Just Select : Statistical Tool

The screenshot shows the 'Group By' dialog with a green box highlighting the 'Grouping' section where 'ProductKey' is selected. A green arrow points from this section to the 'Aggregation' section, which shows 'Total Order Quantity' being aggregated with the 'Sum' operation. The 'Column' dropdown is also visible.

Transpose Changing row to column, Column to Row : is transpose

The screenshot shows the 'Transpose' dialog. It has a green box around the 'Group By' section where 'ProductKey' is selected. A green arrow points from this section to the 'Transpose' section, which shows 'Total Order Quantity' being transposed. The 'Column' dropdown is also visible. To the right, a preview pane shows the transposed data.

PIVOTING & UNPIVOTING

Pivoting describes the process of turning distinct row values into columns, and unpivoting describes the process of turning distinct columns into rows

The diagram illustrates the relationship between two tables. The top table is a pivoted version of the bottom table. Arrows labeled "PIVOTING" point from the bottom table to the top table, and arrows labeled "UNPIVOTING" point from the top table back to the bottom table.

	Date	Product Category	North Region	Central Region	South Region
1	7/1/2022	Bikes	10	19	25
2	7/1/2022	Components	14	31	16
3	7/1/2022	Clothing	35	32	46

	Date	Product Category	Region	Quantity Sold
1	7/1/2022	Bikes	North Region	10
2	7/1/2022	Bikes	Central Region	19
3	7/1/2022	Bikes	South Region	25
4	7/1/2022	Components	North Region	14
5	7/1/2022	Components	Central Region	31
6	7/1/2022	Components	South Region	16
7	7/1/2022	Clothing	North Region	35
8	7/1/2022	Clothing	Central Region	32
9	7/1/2022	Clothing	South Region	46

Imagine the table on a hinge; pivoting rotates it from vertical to horizontal, and unpivoting rotates it from horizontal to vertical

NOTE: Transpose works very similarly, but doesn't recognize unique values; instead, the entire table is transformed so that each row becomes a column and vice versa

The diagram shows how individual rows from a table can be converted into new columns. A green box highlights a row in the main table, which is then shown as a new column in a secondary table.

	North Region	Central Region	South Region
10	19	25	
14	31	16	
35	32	46	
13	42	26	
47	23	41	
41	36	9	
13	14	27	
45	21	28	
18	24	12	
11	32	18	
43	35	35	
49	8	15	
36	33	43	
16	17	30	
15	18	28	
34	28	31	
16	43	14	

	Date	Product Category
7/1/2022	Bikes	
7/1/2022	Components	
7/1/2022	Clothing	
7/1/2022	Accessories	
7/2/2022	Bikes	
7/2/2022	Components	

1 column - Region

Individual Row, Can be converted to another new column - "Quantity Sold"

MERGING QUERIES

The screenshot shows the 'Merge' dialog box in Power BI. On the left, there's a sidebar with options: 'Merge Queries' (selected), 'Append Queries', 'Combine Files', and 'Combine'. The main area is titled 'Merge' with the instruction 'Select a table and matching columns to create a merged table.' Below this, there are two tables: 'Sales Data' and 'Product Lookup'. The 'Sales Data' table has columns: Order Date, Product Key, Customer Key, Order Quantity, Index, Stock Date, Order Number, and Territory. The 'Product Lookup' table has columns: Product Key, Product Subcategory Key, Product S K U, Product Name, and Model Name. A blue arrow points from the text 'P.K connects with F.K to build a relation.' to the 'Product Key' column in the 'Sales Data' table. A blue box highlights the 'Product Key' column in both tables. At the bottom, there's a 'Join Kind' dropdown set to 'Left Outer (all from first, matching from second)', a checkbox for 'Use fuzzy matching to perform the merge', and a note: '✓ The selection matches 56046 of 56046 rows from the first table.' There are 'OK' and 'Cancel' buttons at the bottom right.

Merging queries allows you to join tables based on a common column (like a lookup in Excel)

In this case we're merging the Sales Data table with the Product Lookup table, which share a common Product Key column

NOTE: Merging adds columns to an existing table/query

Just because you can merge tables, doesn't mean you should!

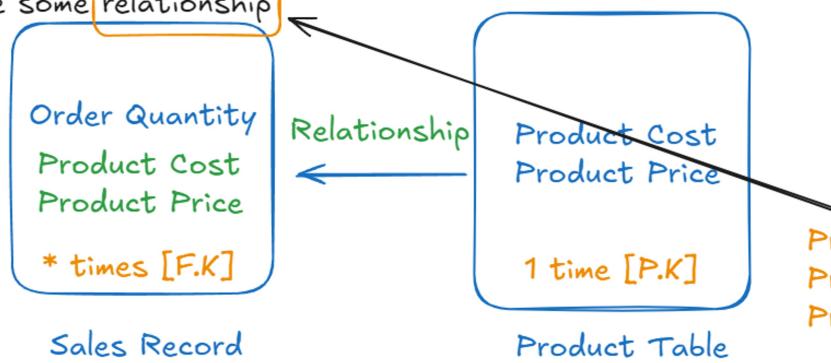
In many cases, it's better to keep tables separate and define relationships between them in the data model (more on that soon!)

Table.ColumnName

Less Efficient,
High Computational Power

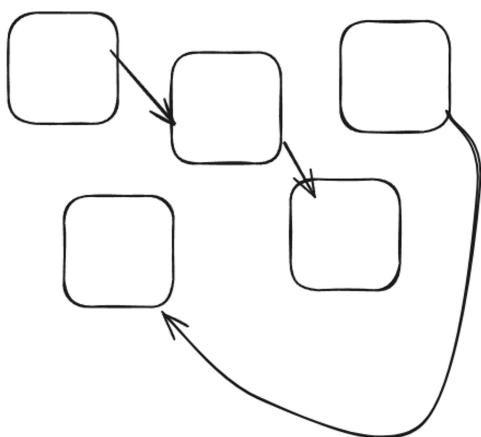
Customer, Product,
Calendar, Sales

At any point of time we can join, but atleast they should have some relationship



We can't store every individual details on a single Table

Problem Statement:
Product Cost/Price from
Product Table , To Sales Table.



Merge

Select a table and matching columns to create a merged table.

Sales Record 2020-2022

Index	OrderDate	StockDate	OrderNumber	ProductKey	CustomerKey	TerritoryKey	OrderLineItem
1	01-01-2020	21-09-2019	SO45080	332	14657	1	
2	01-01-2020	05-12-2019	SO45079	312	29255	4	
3	01-01-2020	29-10-2019	SO45082	350	11455	9	
4	01-01-2020	16-11-2019	SO45081	338	26782	6	

Product Lookup

ProductKey	ProductSubcategoryKey	ProductSKU	SKU Type	ProductName	ModelName
214	31	HL-U509-R	HL	Sport-100 Helmet, Red	Sport-100
215	31	HL-U509	HL	Sport-100 Helmet, Black	Sport-100
218	23	SO-B909-M	SO	Mountain Bike Socks, M	Mountain Bike Socks
219	23	SO-B909-L	SO	Mountain Bike Socks, L	Mountain Bike Socks

Join Kind: Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

Fuzzy matching options

✓ The selection matches 56046 of 56046 rows from the first table.

OK Cancel

Queries [8]

- Product Lookup
- Customer Lookup
- Calendar Lookup
- Sales Data 2020
- Sales Data 2021
- Sales Data 2022
- Sales Record 2020-2022**
- Product Category Sales ...

= Table.NestedJoin(Source, {"ProductKey"}, #“Product Lookup”, {"ProductKey"}, “Product Lookup”, JoinKind.LeftOuter)

	CustomerKey	TerritoryKey	OrderLineItem	OrderQuantity	Quantity Type
1	332	14657	1	1	1 Single Item
2	312	29255	4	1	1 Single Item
3	350	11455	9	1	1 Single Item
4	338	26782	6	1	1 Single Item
5	312	14947	10	1	1 Single Item
6	310	29143	4	1	1 Single Item
7	314	18747	9	1	1 Single Item
8	312	18746	9	1	1 Single Item
9	312	18906	9	1	1 Single Item
10	310	29170	4	1	1 Single Item
11	345	11398	10	1	1 Single Item
12	313	18899	9	1	1 Single Item

Quantity Type Product Lookup

(Select All Columns) ProductKey ProductSubcategoryKey ProductSKU SKU Type ProductName ModelName ProductDescription ProductColor ProductSize ProductStyle ProductCost ProductPrice

Valid: 100% Error: 0% Empty: 0%

4 distinct, 0 unique

413.15 699.10

413.15 699.10

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

2,171.29 3,578.27

OK Cancel

Table Reference .



Transform :

Query Settings

Properties

Name: Product Categories Lookup

Applied Steps

Source: Promoted Headers

Changed Type



Query Settings

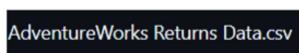
Properties

Name: AdventureWorks Product Subcategories

Applied Steps

Source: Promoted Headers

Changed Type



Query Settings

Properties

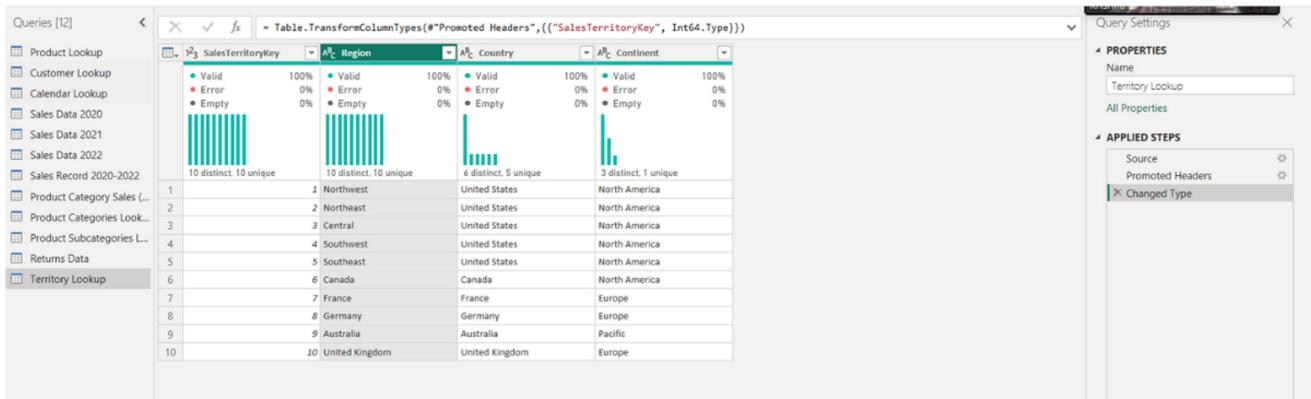
Name: Returns Data

Applied Steps

Source: Promoted Headers

Changed Type

AdventureWorks Territory Lookup.csv



The screenshot shows the Power Query Editor interface. On the left, there's a list of queries: Product Lookup, Customer Lookup, Calendar Lookup, Sales Data 2020, Sales Data 2021, Sales Data 2022, Sales Record 2020-2022, Product Category Sales, Product Categories Look..., Product Subcategories L..., Returns Data, and Territory Lookup. The 'Territory Lookup' query is selected. The main area displays a table with four columns: SalesTerritoryKey, Region, Country, and Continent. The preview pane shows 10 distinct rows. The 'APPLIED STEPS' pane on the right lists 'Source', 'Promoted Headers', and 'Changed Type'.

POWER QUERY BEST PRACTICES

Get organized before connecting and loading data

- Define clear and intuitive table/query names from the start, and establish an organized file/folder structure if you are working with local flat files to avoid changes to file names or paths.

Disable report refresh for any static data sources

- There's no need to constantly refresh data sources that don't change, like lookups or static data tables

When working with large tables, only load the data you need

- Don't include hourly data when you only need daily, or transaction-level data when only need a product-level summary (extra data will only slow your report down!)

Exploring Web Connector : Using Cricbuzz Example;

Queries [14]

- Product Lookup
- Customer Lookup
- Calendar Lookup
- Sales Data 2020
- Sales Data 2021
- Sales Data 2022
- Sales Record 2020-2022
- Product Category Sales ...
- Product Categories Look...
- Product Subcategories L...
- Returns Data
- Territory Lookup
- Batting Record
- Bowling Record**

Table.RenameColumns(#"Promoted Headers", {"O", "Over"}, {"W", "Wicket"}, {"NB", "No Ball"}, {"H", "Maiden Over"}, {"R", "Runs"}, {"ECO", "Economy"})

A ₁ Over	A ₂ Wicket	A ₃ No Ball	A ₄ Bowler	A ₅ Maiden Over	A ₆ Runs	A ₇ Economy
Valid 100%	Valid 100%	Valid 100%	Valid 100%	Valid 100%	Valid 100%	Valid 100%
Error 0%	Error 0%	Error 0%	Error 0%	Error 0%	Error 0%	Error 0%
Empty 0%	Empty 0%	Empty 0%	Empty 0%	Empty 0%	Empty 0%	Empty 0%
4 distinct, 1 unique	4 distinct, 1 unique	3 distinct, 2 unique	13 distinct, 13 unique	2 distinct, 1 unique	12 distinct, 11 unique	11 distinct, 9 unique
1 4	0	0	Mahedi Hasan	0	45	11.20
2 4	1	3	Tasmin Ahmed	0	51	12.80
3 4	3	0	Tanzim Hasan Sakib	0	66	16.50
4 4	1	0	Mustafizur	0	52	13.00
5 2	0	0	Rishad Hossain	0	46	23.00
6 2	1	0	Mahmudullah	0	26	13.00
7 4	2	1	Mayank Yadav	0	32	8.00
8 3	0	0	Hardik Pandya	0	32	10.70
9 1	1	0	Washington Sundar	0	4	4.00
10 3	1	0	Nitish Reddy	0	31	10.30
11 4	3	0	Ravi Bishnoi	1	30	7.50
12 4	0	0	Varun Chakaravarthy	0	23	5.80
13 1	0	0	Abhishek Sharma	0	8	8.00

Query Settings

- PROPERTIES**
 - Name: Bowling Record
 - All Properties
- APPLIED STEPS**
 - Source: Extracted Table From Html
 - Removed Duplicates
 - Promoted Headers
 - Renamed Columns**

Queries [14]

- Product Lookup
- Customer Lookup
- Calendar Lookup
- Sales Data 2020
- Sales Data 2021
- Sales Data 2022
- Sales Record 2020-2022
- Product Category Sales ...
- Product Categories Look...
- Product Subcategories L...
- Returns Data
- Territory Lookup
- Batting Record
- Bowling Record**

Table.TransformColumnTypes(#"Renamed Columns", {"Runs": Int64.Type}, {"Balls": Int64.Type}, {"4s": Int64.Type})

A ₁ Batsmen Name	A ₂ Wicket Reason	A ₃ Runs	A ₄ Balls	A ₅ 4s
Samson (wk)	c Mahedi Hasan b Mustafizur	112	47	21
2 Abhishek Sharma	c Mahedi Hasan b Tanzim Hasan Sakib	4	4	1
3 Suryakumar Yadav (c)	c Rishad Hossain b Mahmudullah	75	35	8
4 Riyam Parag	c Litton Das b Tasmin Ahmed	34	13	1
5 Hardik Pandya	c Rishad Hossain b Tanzim Hasan Sakib	47	18	4
6 Rinku Singh	not out	8	4	0
7 Nitish Reddy	c Mahedi Hasan b Tanzim Hasan Sakib	0	1	0
8 Washington Sundar	not out	1	1	0
9 Parvez Hossain Emon	c Riyam Parag b Mayank Yadav	0	1	0
10 Tanzim Hasan	c Varun Chakaravarthy b Washington Sundar	15	12	3
11 Najmul Hossain Shanto (c)	c Samson b Ravi Bishnoi	14	11	1
12 Litton Das (wk)	c (sub)Tilak Varma b Ravi Bishnoi	42	25	8
13 Towhid Hridoy	not out	63	42	5
14 Mahmudullah	c Riyam Parag b Mayank Yadav	8	9	1

Query Settings

- PROPERTIES**
 - Name: Batting Record
 - All Properties
- APPLIED STEPS**
 - Source: Extracted Table From Html
 - Renamed Columns
 - Changed Type**