

Lesson - 4 homework

Question 1: What is the difference between "Merge" and "Append" in Power Query?

Merge is for combining tables side-by-side (like a VLOOKUP). You use it to add new columns from one table to another using a shared key, like adding customer names to an orders table using CustomerID.

Merge

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

Orders

CustID	Name	OrderDate	Product	Quantity	Price
101	Alice	1/10/2023	Laptop	1	1200
102	Bob	1/15/2023	Mouse	3	25
101	Alice	1/20/2023	Keyboard	2	80
103	Charlie	1/25/2023	Monitor	1	300

Customer_Orders

CustID	Name	Email
101	Alice	alice@example.com
102	Bob	bob@example.com
103	Charlie	charlie@example.com

Join Kind

Left Outer (all from first, matching from second)

☐ Use fuzzy matching to perform the merge

▸ Fuzzy matching options

⚠ Select the same number of columns from both visible tables to continue.

OK Cancel

Queries [3]		= Table.ExpandTableColumn(Source, "Customer_Orders", {"Email"}, {"Customer_Orders.Email"})						
Orders	OrderDate	Product	Quantity	Price	Customer_Orders.Email			
Valid	100%	Valid	100%	Valid	100%	Valid	100%	
Error	0%	Error	0%	Error	0%	Error	0%	
Empty	0%	Empty	0%	Empty	0%	Empty	0%	
1	1/10/2023	Laptop	1	1200	alice@example.com			
2	1/20/2023	Keyboard	2	80	alice@example.com			
3	1/15/2023	Mouse	3	25	bob@example.com			
4	1/25/2023	Monitor	1	300	charlie@example.com			

Append is for stacking tables on top of each other. You use it to add new rows, like combining monthly sales tables into one yearly sales table.

Append

Concatenate rows from two tables into a single table.

☒ Two tables ☐ Three or more tables

First table
Orders

Second table
Orders

OK Cancel

Queries [4] ✕ ✓ fx = Table.Combine({Orders, Orders})

	123 CustID	AB_C Name	AB_C OrderDate	AB_C Product	123 Quantity	123
	Valid 100%	Valid 100%	Valid 100%	Valid 100%	Valid 100%	Valid 100%
	Error 0%	Error 0%	Error 0%	Error 0%	Error 0%	Error 0%
	Empty 0%	Empty 0%	Empty 0%	Empty 0%	Empty 0%	Empty 0%
1	101	Alice	1/10/2023	Laptop	1	1
2	102	Bob	1/15/2023	Mouse	3	3
3	101	Alice	1/20/2023	Keyboard	2	2
4	103	Charlie	1/25/2023	Monitor	1	1
5	101	Alice	1/10/2023	Laptop	1	1
6	102	Bob	1/15/2023	Mouse	3	3
7	101	Alice	1/20/2023	Keyboard	2	2
8	103	Charlie	1/25/2023	Monitor	1	1

Question 2: How do you split a "Full Name" column into "First Name" and "Last Name"?

1. Select the Column: Click on the "Full Name" column header.
2. Go to the Transform Tab: Click this tab at the top.
3. Click 'Split Column':
4. Choose 'By Delimiter':
5. Select the Space: Choose Space as the delimiter.
6. Choose 'Each occurrence of the delimiter' (This is the key step for names with middle names).

Before:

	123 CustomerID	AB_C Name	AB_C Age	AB_C Email	AB_C PurchaseAmount
	Valid 100%	Valid 100%	Valid 96%	Valid 100%	Valid 98%
	Error 0%	Error 0%	Error 0%	Error 0%	Error 0%
	Empty 0%	Empty 0%	Empty 4%	Empty 0%	Empty 2%
1	1001	eva martin	70	henry@example.com	800.5
2	1002	mary jones	sixty	betty@mail.com	200
3	1003	Mark Anderson	Thirty-five	helen@	350
4	1004	Charlie Anderson	60	helen@	NaN
5	1005	Helen Jones	Twenty-two	mark@.mail.com	300
6	1006	CHRISTOPHER GARCIA	60	frank@mail.com	one hundred
7	1007	D@niel Thomas	Thirty-five	robert.com	410
8	1008	Frank Johnson	60	paul@mail	800.5
9	1009	Mary Martin	29	barbara@mail.com	430
10	1010	J@ckson Davis	Twenty-two	frank@mail.com	650
11	1011	HELEN MARTIN	70	henry@example.com	800.5

Process:

The screenshot displays a data transformation tool interface. On the left, a dark-themed context menu is open, listing various actions for a selected column. The menu items include: Copy, Remove, Remove Other Columns, Duplicate Column, Add Column From Examples..., Remove Duplicates, Remove Errors, Change Type, Transform, Replace Values..., Replace Errors..., Split Column, Group By..., Fill, Unpivot Columns, Unpivot Other Columns, Unpivot Only Selected Columns, Rename..., Move, Drill Down, and Add as New Query. The 'Split Column' option is currently selected, and its sub-menu is visible on the right. This sub-menu lists several splitting methods: By Delimiter..., By Number of Characters..., By Positions..., By Lowercase to Uppercase, By Uppercase to Lowercase, By Digit to Non-Digit, and By Non-Digit to Digit. In the background, a data table is visible with columns labeled 'A', 'C', 'Email', and 'A'. The 'Email' column contains several email addresses, and the 'A' column contains numerical values. A summary row at the top of the table shows statistics for the 'Email' column: 96% Valid, 0% Error, 4% Empty, and 100% for the 'A' column.

	A	C	Email	A
Summary	96%	Valid	100%	Valid
	0%	Error	0%	Error
	4%	Empty	0%	Empty
		henry@example.com	80	
		betty@mail.com	20	
		helen@	35	
		helen@	N	
		mark@@mail.com	30	
		frank@mail.com	or	
		robert.com	41	
		paul@mail	80	
		nancy @mail.com	28	

Split Column by Delimiter

Specify the delimiter used to split the text column.

Select or enter delimiter

Space

Split at

- ☐ Left-most delimiter
- ☐ Right-most delimiter
- ☒ Each occurrence of the delimiter

Advanced options

Quote Character

"

☐ Split using special characters

Insert special character

OK

After:

<div><div>Valid</div><div>Error</div><div>Empty</div></div>	<div><div>100%</div><div>0%</div><div>0%</div></div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div>	<div><div>91%</div><div>0%</div><div>9%</div></div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div>	<div><div>100%</div><div>0%</div><div>0%</div></div>
1012	Bob	Jones			
1013	James	Williams			
1014	David	Williams			
1015	Thomas	Thompson			
1016	Sophia	Anderson			
1017	Thom@S	Davis			
1018	N@Ncy	Davis			
1019	Patricia	Jones			
1020	Kevin	Thompson			
1021	Fr@Nk	Smith			
1022	Charles	Thompson			

Question 3: What is "Pivot Columns" used for?

Pivot Columns is used to reshape your data from a "long" format into a "wide" format. It takes the unique values from one column and transforms them into multiple new columns, summarizing another column's values for each of those new categories.

Imagine you have a tall, long table like this:

Product	Month	Sales
Apples	Jan	100
Apples	Feb	150
Oranges	Jan	75
Oranges	Feb	120

After pivoting the "Month" column, you get a wide, summary table:

Product	Jan	Feb
Apples	100	150
Oranges	75	120

Question 4: What is "Pivot Columns" used for?

1. In the Power Query Editor, look at the "Query Settings" pane on the right.
2. You'll see a list called "Applied Steps". This is a history of everything you've done.
3. To undo the last thing you did, simply click on the step above it in the list.
4. The query will instantly revert to how it was at that point, effectively "deleting" all the steps that came after.
5. To delete a specific step: Click the X next to the step's name. Be careful, as this will also delete all steps that come after it, since each step builds on the previous one.

Question 5: What is "Pivot Columns" used for?

Duplicate: Creates a full, independent copy of the query. Changing the original or the copy does not affect the other. Use it to experiment without messing up your original.

Reference: Creates a new query that points to the original. If you change the original query later (like adding a step), the reference will automatically get those changes too. Use it to build different tables (like a summary) from the same cleaned data source.

Analogy:

Duplicate is like making a photocopy of a document. Editing the original or the copy doesn't change the other.

Reference is like creating a shortcut or a link to the original document. If you update the original, the shortcut shows the latest version.

Question 6: Merge Orders.csv and Customers.xlsx on CustID (inner join).

Merge

Select tables and matching columns to create a merged table.

Orders

CustID	Name	OrderDate	Product	Quantity	Price
101	Alice	1/10/2023	Laptop	1	1200
102	Bob	1/15/2023	Mouse	3	25
101	Alice	1/20/2023	Keyboard	2	80
103	Charlie	1/25/2023	Monitor	1	300

Customer_Orders

CustID	Name	Email
101	Alice	alice@example.com
102	Bob	bob@example.com
103	Charlie	charlie@example.com

Join Kind

Inner (only matching rows)

☐ Use fuzzy matching to perform the merge

Fuzzy matching options

✓

The selection matches 4 of 4 rows from the first table, and 3 of 3 rows from...

OK

Cancel

Product	Quantity	Price	Customer_Orders.Name	Customer_Orders.Email
Laptop	1	1200	Alice	alice@example.com
Keyboard	2	80	Alice	alice@example.com
Mouse	3	25	Bob	bob@example.com
Monitor	1	300	Charlie	charlie@example.com

Question 7: Pivot the Product column to show total Quantity per product.

1. Select the Product column.
2. Go to the Transform tab.
3. Click Pivot Column.
4. In the window that pops up:
5. Values Column: Choose Quantity from the dropdown.
6. Aggregate Value Function: Ensure Sum is selected (it usually is by default).

Pivot Column

Use the names in column "Product" to create new columns.

Values Column ⓘ

Quantity

▶ Advanced options

[Learn more about Pivot Column](#)

OK Cancel

1 ² 3 Laptop	1 ² 3 Keyboard	1 ² 3 Mouse	1 ² 3 Monitor
<div><div>Valid 25%</div><div>Error 0%</div><div>Empty 75%</div></div>	<div><div>Valid 25%</div><div>Error 0%</div><div>Empty 75%</div></div>	<div><div>Valid 25%</div><div>Error 0%</div><div>Empty 75%</div></div>	<div><div>Valid 25%</div><div>Error 0%</div><div>Empty 75%</div></div>
1	null	null	null
null	2	null	null
null	null	3	null
null	null	null	1

Question 8: Append two tables with identical columns (e.g., Orders_Jan.csv + Orders_Feb.csv).

1. Go to the Home tab.
2. Click Append Queries (As new)
3. In the window, select your second table (e.g., Orders_Feb).

Append

Concatenate rows from two tables into a single table.

☒ Two tables ☐ Three or more tables

First table

Orders_Jan

Second table

Orders_Feb

OK Cancel

	123 CustID	A8C Name	ABC 123 OrderDate	A8C Product	123 Quantity	12
	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	
1	101	Alice	1/10/2023	Laptop	1	
2	102	Bob	1/15/2023	Mouse	3	
3	101	Alice	1/20/2023	Keyboard	2	
4	103	Charlie	1/25/2023	Monitor	1	
5	101	Alice	2023-02-10	Laptop	1	
6	102	Bob	2023-02-15	Mouse	3	
7	101	Alice	2023-02-20	Keyboard	2	
8	103	Charlie	2023-0-25	Monitor	1	

Question 9: Use "Fill Down" to replace nulls in the Email column with the previous value.

1. Select the Email column.
2. Go to the Transform tab.
3. Click Fill → Down.

All the nulls in the Email column will be replaced with the value from the cell above them. Done.

Copy	
Remove	
Remove Other Columns	
Duplicate Column	
Add Column From Examples...	
Remove Duplicates	
Remove Errors	
Change Type	
Transform	
Replace Values...	
Replace Errors...	
Split Column	
Group By...	
Fill	Down
Unpivot Columns	Up
Unpivot Other Columns	
Unpivot Only Selected Columns	
Rename...	
Move	
Drill Down	
Add as New Query	

	123 CustID	A8C Name	A8C Email
	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>
1	101	Alice	alice@example.com
2	102	Bob	bob@example.com
3	103	Charlie	charlie@example.com

Question 10: Extract the domain (e.g., "example.com") from the Email column.

- Select the Email column.
- Go to the Add Column tab.
- Click Extract → Text After Delimiter.

Text After Delimiter

Enter the delimiter that marks the beginning of what you would like to extract.

Delimiter

ABC

@

Advanced options

OKCancel

	123 CustID	ABC Name	ABC Email	ABC domain
	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>
1	101	Alice	alice@example.com	example.com
2	102	Bob	bob@example.com	example.com
3	103	Charlie	charlie@example.com	example.com

Question 11: Write M-code to merge queries dynamically based on a parameter (e.g., JoinType = "Inner").

```
1 let
2   ...// This converts your JoinType parameter to the correct JoinKind
3   ... GetJoinKind =
4   ...   if JoinType = "Inner" then JoinKind.Inner
5   ...   else if JoinType = "Left" then JoinKind.LeftOuter
6   ...   else if JoinType = "Right" then JoinKind.RightOuter
7   ...   else if JoinType = "Full" then JoinKind.FullOuter
8   ...   else JoinKind.LeftOuter, // Fallback to LeftOuter if something goes wrong
9   ...
10  ...// Your merge is now dynamic!
11  ... Source = Table.NestedJoin(
12  ...   Orders,
13  ...   {"CustID"},
14  ...   Customer_Orders,
15  ...   {"CustID"},
16  ...   "Customer_Orders",
17  ...   GetJoinKind // Using the parameter here instead of hardcoded JoinKind.LeftOuter
18  ... )
19  in
20  ... Source
```

	123 Price	123 Customer_Orders.CustID	ABC Customer_Orders.Name	ABC Customer_Orders.Email	ABC Customer_Orders
	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid</div> <div>Error</div> <div>Empty</div>
1	1200	101	Alice	alice@example.com	example.com
2	80	101	Alice	alice@example.com	example.com
3	25	102	Bob	bob@example.com	example.com
4	300	103	Charlie	charlie@example.com	example.com

Question 12: Unpivot a table with columns like "Jan_Sales," "Feb_Sales" into a "Month" and "Sales" format.

```
1 let
2     Source = Jan-Feb,
3     //Convert the text dates to proper date format first
4     ChangeType = Table.TransformColumnTypes(Source,{{"Month", type date}}),
5     //Extract month name
6     ExtractMonth = Table.TransformColumns(
7         ChangeType,
8         {"Month", each Date.MonthName(_), type text}
9     ),
10    //Get short month names (optional)
11    ShortMonth = Table.TransformColumns(
12        ExtractMonth,
13        {"Month", each Text.Start(_, 3), type text}
14    )
15 in
16    ShortMonth
```

Question 13: Handle errors in a custom column (e.g., division by zero) using try...otherwise.

```
1 let
2     Source = Csv.Document(File.Contents("C:\Users\apple-service\Documents\Data_Analytics\Power_BI\
3         Classes\bi_class_4\powerbi_dirty_dataset_fullnames_100.csv"),[Delimiter=",", Columns=7, Encoding=1252,
4         QuoteStyle=QuoteStyle.None]),
5     #"Promoted Headers" = Table.PromoteHeaders(Source, [PromoteAllScalars=true]),
6     #"Removed Other Columns" = Table.SelectColumns("#Promoted Headers",{"CustomerID", "PurchaseAmount"}),
7     #"Changed Type" = Table.TransformColumnTypes("#Removed Other Columns",{{"PurchaseAmount", Currency.Type}}),
8     #"Replaced Errors" = Table.ReplaceErrorValues("#Changed Type", {{"PurchaseAmount", null}}),
9     #"Added Custom" = Table.AddColumn(
10         #"Replaced Errors",
11         "AdjustedPurchase",
12         each try [PurchaseAmount] * 1.1 otherwise 0,
13         type number
14     )
15 in
16     #"Added Custom"
```

	AB C CustomerID	\$ PurchaseAmount	1.2 AdjustedPurchase
	<div><div>Valid</div><div>100%</div></div>	<div><div>Valid</div><div>81%</div></div>	<div><div>Valid</div><div>81%</div></div>
	<div><div>Error</div><div>0%</div></div>	<div><div>Error</div><div>0%</div></div>	<div><div>Error</div><div>0%</div></div>
	<div><div>Empty</div><div>0%</div></div>	<div><div>Empty</div><div>19%</div></div>	<div><div>Empty</div><div>19%</div></div>
24	1024	-150.00	-165
25	1025	900.00	990
26	1026	400.75	440.825
27	1027	285.00	313.5
28	1028	null	null
29	1029	null	null
30	1030	-150.00	-165
31	1031	450.75	495.825
32	1032	800.50	880.55
33	1033	300.00	330

Question 14: Create a function in Power Query to clean phone numbers (e.g., remove dashes).

(phoneNumber as text) as text =>

let

Cleaned = Text.Remove(phoneNumber, {" ", "-", "(", ")", ".", "+"}),

Result = Cleaned

in

Result

Question 15: Optimize a query with 10+ steps—identify bottlenecks and simplify.

Applied Steps:

- 1. Source: Import from C:\Data\Customers.csv (has 20 columns)**
- 2.**
- 3. Promoted Headers: First row contains column names.**
- 4.**
- 5. Changed Type: Changed all columns to type text to prevent errors.**
- 6.**
- 7. Renamed Columns: Renamed Phone Number to Phone.**
- 8.**
- 9. Added Custom Column: Created FullName by combining FirstName and LastName columns.**
- 10.**
- 11. Removed Columns: Removed FirstName and LastName (now that we have FullName).**
- 12.**
- 13. Added Custom Column: Used a complex Text.Remove formula to clean the Phone column.**
- 14.**
- 15. Filtered Rows: Kept only rows where Country = "USA".**
- 16.**
- 17. Changed Type (again): Changed SignUpDate from text to a date type.**
- 18.**
- 19. Replaced Errors: For any rows where the date conversion failed, replaced with null.**
- 20.**
- 21. Filtered Rows (again): Kept only rows where SignUpDate is not null.**
- 22.**
- 23. Reordered Columns: Moved FullName to the first column.**