### [BITE-SIZED POWER QUERY]

# SIMPLIFIED TABLE JOINS

### Merge these tables to add Category to the Sales rows

	Α	В	С	D	Ε	F	G	
1								
2		Table nam	ie: Sales			Table name	: Category	
3		Product	Date	Amount		Product	Category	
4		Apples	29-Jan-23	2585		Apples	Pome Fruit	
5		Lemons	29-Jan-23	4626		Lemons	Citrus Fruit	
6		Lemons	28-Jan-23	4680		Oranges	Citrus Fruit	
7		Oranges	29-Jan-23	4928		Pears	Pome Fruit	
8		Pears	29-Jan-23	3083				
9		Pears	28-Jan-23	4599				
10		Pears	27-Jan-23	2803				
11								

## **UI Approach**

```
UI Merge defaults to
                                        We then need to
       Table.NestedJoin,
                                         expand the table
       adding a new
                                         column. If we de-
        column containing a
                                         select "Prefix new
        Table on each row
                                         columns", a suffix is
let
                                         added to the new
    Source > Table.NestedJoin(
                                         column names
        Sales, {"Product"},
        Category, {"Product"},
        "Category", JoinKind.LeftOuter),
    #"Expanded Category" = Table.ExpandTableColumn(
        Source, "Category",
        {"Category"}, {"Category.1"}),
    #"Renamed Columns" = Table.RenameColumns(
        #"Expanded Category",
        {{"Category.1", "Category"}})
in
    #"Renamed Columns"
                                      We then need to
                                      rename the
                                      column to tidy up
```

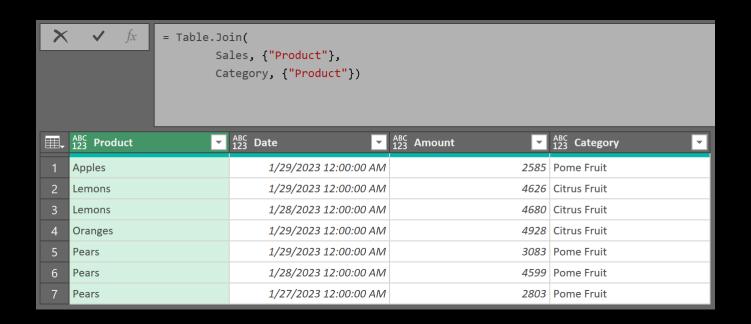
×	✓ fx = Table.Re	nameColumns(#"Expanded Co	ategory",{{"Category.1",	"Category"}})
<b>III</b> ~	ABC Product	ABC Date	ABC Amount	ABC 123 Category
1	Apples	1/29/2023 12:00:00 AM	2585	Pome Fruit
2	Lemons	1/29/2023 12:00:00 AM	4626	Citrus Fruit
3	Lemons	1/28/2023 12:00:00 AM	4680	Citrus Fruit
4	Oranges	1/29/2023 12:00:00 AM	4928	Citrus Fruit
5	Pears	1/29/2023 12:00:00 AM	3083	Pome Fruit
6	Pears	1/28/2023 12:00:00 AM	4599	Pome Fruit
7	Pears	1/27/2023 12:00:00 AM	2803	Pome Fruit

## Table.Join

Table.Join defaults to JoinKind.Inner. Since the join columns are identical *and* have identical names, only the non-join columns from the right-table are added to the left table



No prefixes or suffixes are added so long as columns added from the right table are not named the same as any columns in the left table



If we use JoinKind.LeftOuter, both Product columns must be in the output because the left Product column may have rows without a match in the right Product column



#### Duplicate column names will produce an error

#### CategoryWithDate



×	$\checkmark$ $f_X$	= Table.Ad	dColumn(Source, "Date",	each #date(2023,1,1))
	ABC 123 <b>Product</b>	~	ABC 123 Category	ABC Date
1	Apples		Pome Fruit	1/1/2023
2	Lemons		Citrus Fruit	1/1/2023
3	Oranges		Citrus Fruit	1/1/2023
4	Pears		Pome Fruit	1/1/2023

#### let

```
Source = Table.Join(
    Sales, {"Product"},
    CategoryWithDate, {"Product"})
```

#### in

#### Source

Because Date is in both tables, the join produces an error



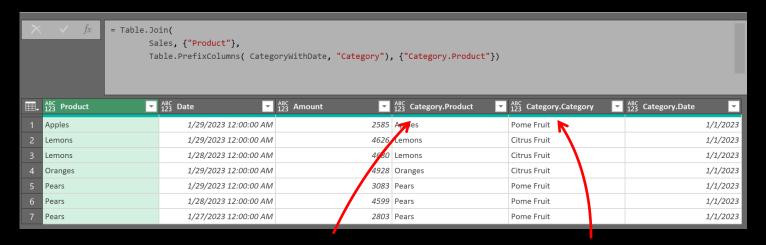
```
Table.Join(
Sales, {"Product"},
CategoryWithDate, {"Product"})

Expression.Error: A join operation cannot result in a table with duplicate column names ("Date").
Details:
[Type]
```

We can prefix all the columns in the right table by using Table.PrefixColumns

We must remember to add the prefix to the right-table's join column(s) list in Table.Join

Source



Unfortunately, we now have the Category.Product column in the result as well as the Product column

The category is also prefixed, even though it doesn't need to be.

# Custom function "SimpleJoin"

```
SimpleJoin =
(
    LeftTable as table,
    JoinColumns as list,
    RightTable as table,
    optional JoinColumnsRight as list,
    optional DuplicateColumnPrefix as text,
    optional JoinKind as number
) =>
let (etc...)
```

- Only specify join columns once if join columns are the same in both tables
- Automatically handle column renaming in case of duplicate column names
- Only rename duplicate columns, leave non-duplicate columns alone
- Allow an optional column prefix for renamed duplicate columns (default "New ")
- Specify JoinKind when called (default Inner)

Table name: SalesMulti					
Product	Date	Amount			
Apples	1-Jan-23	2585			
Lemons	1-Jan-23	4626			
Lemons	1-Feb-23	4680			
Oranges	1-Feb-23	4928			
Pears	1-Jan-23	3083			
Pears	1-Feb-23	4599			
Pears	1-Dec-22	2803			

Table name: MonthPrice					
Product	Date	Price			
Apples	1-Jan-23	24.57			
Lemons	1-Jan-23	15.03			
Oranges	1-Jan-23	22.14			
Pears	1-Jan-23	23.16			
Apples	1-Feb-23	18.24			
Lemons	1-Feb-23	17.81			
Oranges	1-Feb-23	23.41			
Pears	1-Feb-23	24.35			
Apples	1-Dec-22	18.88			
Lemons	1-Dec-22	18.95			
Oranges	1-Dec-22	18.14			
Pears	1-Dec-22	22.68			

### A simple join with join columns only specified once. Adds Price to left table

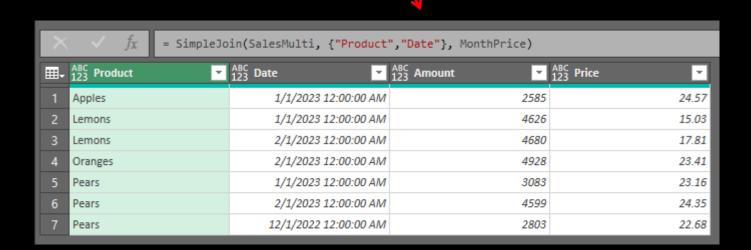


Table name: SalesMulti					
Product	Date	Amount			
Apples	1-Jan-23	2585			
Lemons	1-Jan-23	4626			
Lemons	1-Feb-23	4680			
Oranges	1-Feb-23	4928			
Pears	1-Jan-23	3083			
Pears	1-Feb-23	4599			
Pears	1-Dec-22	2803			

Table name: MonthPrice					
Product	Date	Price			
Apples	1-Jan-23	24.57			
Lemons	1-Jan-23	15.03			
Oranges	1-Jan-23	22.14			
Pears	1-Jan-23	23.16			
Apples	1-Feb-23	18.24			
Lemons	1-Feb-23	17.81			
Oranges	1-Feb-23	23.41			
Pears	1-Feb-23	24.35			
Apples	1-Dec-22	18.88			
Lemons	1-Dec-22	18.95			
Oranges	1-Dec-22	18.14			
Pears	1-Dec-22	22.68			

If JoinKind is not inner, the join columns are renamed according to the supplied prefix. Price is not renamed because it's not a duplicate

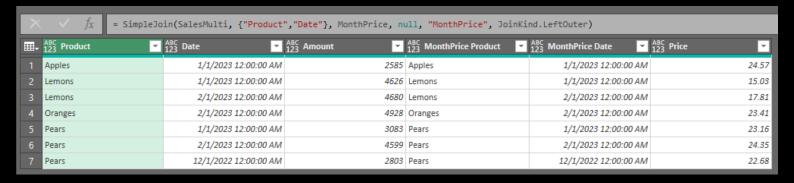
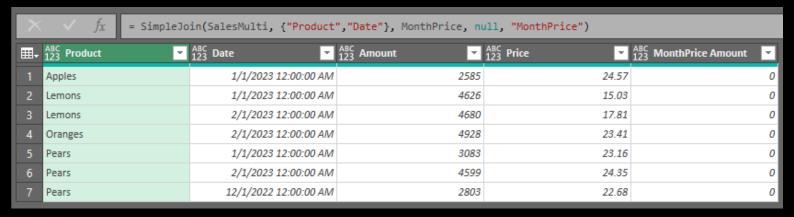


Table name: SalesMulti					
Date	Amount				
1-Jan-23	2585				
1-Jan-23	4626				
1-Feb-23	4680				
1-Feb-23	4928				
1-Jan-23	3083				
1-Feb-23	4599				
1-Dec-22	2803				
	1-Jan-23 1-Jan-23 1-Feb-23 1-Feb-23 1-Jan-23 1-Feb-23				

Table name: MonthPrice						
Product	Date	Price	Amount			
Apples	1-Jan-23	24.57	0			
Lemons	1-Jan-23	15.03	0			
Oranges	1-Jan-23	22.14	0			
Pears	1-Jan-23	23.16	0			
Apples	1-Feb-23	18.24	0			
Lemons	1-Feb-23	17.81	0			
Oranges	1-Feb-23	23.41	0			
Pears	1-Feb-23	24.35	0			
Apples	1-Dec-22	18.88	0			
Lemons	1-Dec-22	18.95	0			
Oranges	1-Dec-22	18.14	0			
Pears	1-Dec-22	22.68	0			

Amount in the right table is renamed because it's a duplicate. Price is not. Because JoinKind is not specified, default is Inner and Product and Date from the right table are not returned



# For a walkthrough of how SimpleJoin works and to grab the code:

https://www.youtube.com/@flexyourdata