

K2's Excel's Latest And Greatest Features

Microsoft Excel offers hundreds of features, functions, tools, and options to business professionals, including accounting and financial professionals – so many that it can be challenging to identify the most useful ones. In this session, you will learn which features to focus on and how to use them to maximize your return on investment in Excel. PivotTables, Power Query, Power Pivot, tables, and new features and functions are all covered in this indispensable session.

Introduction

Microsoft Excel offers hundreds of features, functions, tools, and options to business professionals, including accounting and financial professionals – so many that it can be challenging to identify the most useful ones. In this session, you will learn which features to focus on and how to use them to maximize your return on investment in Excel. PivotTables, Power Query, Power Pivot, tables, and new features and functions are all covered in this indispensable session.

Learning Objectives

Upon completing this session, you should be able to:

- Create formulas that incorporate new Excel functions, including XLOOKUP and LAMBDA;
- Generate PivotTables quickly and easily;
- Utilize Power Query to link external data to Excel and apply transformations to the linked data;
- Apply formatting to Excel spreadsheets quickly and efficiently; and
- Identify the benefits associated with Excel's Tables feature and incorporate Tables into your daily routines

Manipulating Text In Excel

Most business professionals – especially accounting and financial professionals – use Excel primarily to work with numerical data. But, on occasion, they may also need to manipulate text in Excel. Fortunately, Excel provides legacy and new options to assist with those efforts.

Text To Columns Wizard

Excel's **Text To Columns Wizard** is a legacy feature that can help split a single data column into multiple columns. This feature is straightforward and often adequately handles the designated task. To illustrate, consider the example pictured in **Figure 1**. Further, suppose you needed to quickly split the combined data in column B into separate columns – one for the name, another for the address, another for the city, etc. Excel's Text To Columns Wizard is often the right tool for tasks like these without injecting additional formulas into the workbook.

You can perform the following steps to split the data into individual columns using Text To Columns.

1. Select the source range of data, which in this case is cells **A2:A13**.
2. Click **Text To Columns** in the Data Tools group on the Ribbon's **Data** tab.
3. Indicate whether a specific character delimits the data or whether to split the data based on its length. In this example, because a comma separates each field, we choose the "Delimited" option and click **Next**.

4. Select **Comma** in **Step 2** of the **Text To Columns Wizard** and click **Next**. Note that the wizard previews how the data will split after selecting “Comma.”
5. Finally, Step 3 of the Wizard offers options to apply formats if necessary.

	A	B	C
1			
2		Nancy Freehafer, 123 Main St, Atlanta, GA	
3		Andrew Cencini, 456 Elm St, Houston, TX	
4		Jan Kotas, 789 Oak Ave, Vancouver, BC	
5		Mariya Sergienko, 1234 Hill Blvd, Los Angeles, CA	
6		Steven Thorpe, 6789 Biscayne Ln, Miami, FL	
7		Michael Neipper, 135 Ave K, Calgary, AB	
8		Robert Zare, 79 B Central Ave, Minneapolis, MN	
9		Laura Giussani, 246 Park Ave, Dallas, TX	
10		Anne Hellung-Larsen, 468 Spuce Dr, Toronto, ON	
11		Anna Bedecs, 689 Pine Ave, Portland, OR	
12		Antonio Gratacos Solsona, 297 Vista Dr, Denver, CO	
13		Thomas Axen, 638 Golden Peak, Salt Lake City, UT	
14			

Figure 1 - Sample Data For Manipulating Text

Flash Fill

Flash Fill is a formula-free alternative to Text To Columns. Flash Fill utilizes patterns in your data and manipulates the data based on those patterns. Therefore, you will specify the patterns Flash Fill should use when modifying text and Flash Fill will do the heavy lifting for you. To illustrate, consider the data set pictured above in Figure 1. To use Flash Fill to split the data into multiple columns, type in a pattern similar to that shown in **Figure 2**.

	A	B	C	D	E	F
1			Name	Address	City	State/Province
2		Nancy Freehafer, 123 Main St, Atlanta, GA	Nancy Freehafer	123 Main St	Atlanta	GA
3		Andrew Cencini, 456 Elm St, Houston, TX				
4		Jan Kotas, 789 Oak Ave, Vancouver, BC				
5		Mariya Sergienko, 1234 Hill Blvd, Los Angeles, CA				
6		Steven Thorpe, 6789 Biscayne Ln, Miami, FL				
7		Michael Neipper, 135 Ave K, Calgary, AB				
8		Robert Zare, 79 B Central Ave, Minneapolis, MN				
9		Laura Giussani, 246 Park Ave, Dallas, TX				
10		Anne Hellung-Larsen, 468 Spuce Dr, Toronto, ON				
11		Anna Bedecs, 689 Pine Ave, Portland, OR				
12		Antonio Gratacos Solsona, 297 Vista Dr, Denver, CO				
13		Thomas Axen, 638 Golden Peak, Salt Lake City, UT				

Figure 2 - Pattern Established For Flash Fill

With your data identified and a pattern established, click in cell **C3**, and then select **Flash Fill** from the **Data Tools** group on the Ribbon’s **Data** tab. Upon doing so, Flash Fill will attempt to fill the remaining rows with the appropriate data for the selected column, as shown in **Figure 3**.

	A	B	C	D	E	F
1			Name	Address	City	State/Province
2		Nancy Freehafer, 123 Main St, Atlanta, GA	Nancy Freehafer	123 Main St	Atlanta	GA
3		Andrew Cencini, 456 Elm St, Houston, TX	Andrew Cencini			
4		Jan Kotas, 789 Oak Ave, Vancouver, BC	Jan Kotas			
5		Mariya Sergienko, 1234 Hill Blvd, Los Angeles, CA	Mariya Sergienko			
6		Steven Thorpe, 6789 Biscayne Ln, Miami, FL	Steven Thorpe			
7		Michael Neipper, 135 Ave K, Calgary, AB	Michael Neipper			
8		Robert Zare, 79 B Central Ave, Minneapolis, MN	Robert Zare			
9		Laura Giussani, 246 Park Ave, Dallas, TX	Laura Giussani			
10		Anne Hellung-Larsen, 468 Spuce Dr, Toronto, ON	Anne Hellung-Larsen			
11		Anna Bedecs, 689 Pine Ave, Portland, OR	Anna Bedecs			
12		Antonio Gratacos Solsona, 297 Vista Dr, Denver, CO	Antonio Gratacos Solsona			
13		Thomas Axen, 638 Golden Peak, Salt Lake City, UT	Thomas Axen			

Figure 3 - Using Flash Fill To Extract And Insert Data

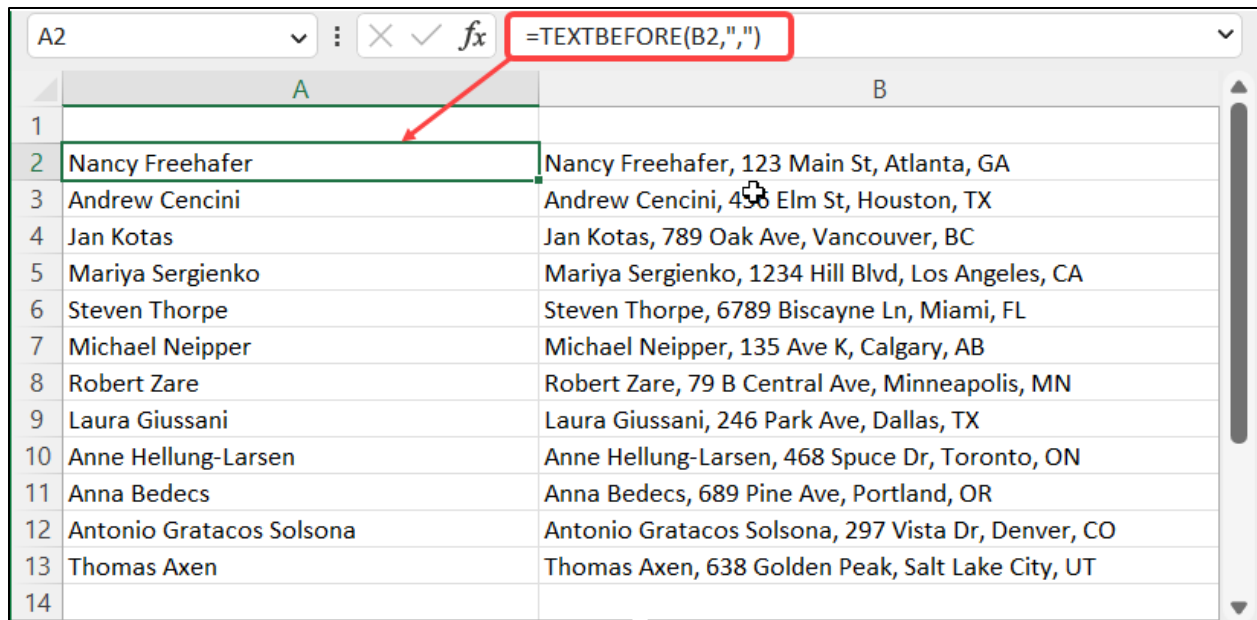
Next, perform the steps above on each additional column where you need to populate data. Flash Fill will attempt to use its' pattern recognition capabilities to fill the data. However, in some cases, Flash Fill may be unable to insert data based on specified patterns. In such cases, enter correct data in the empty cell, and Flash Fill will attempt to "learn" from the new data and fill the cells. Also, note that you can use Flash Fill to join cells together. The process for doing so is the same as for splitting data – specify your data's pattern and then use Flash Fill to concatenate the data.

Formula-Based Approaches For Manipulating Data

Many users will prefer the formula-free approaches discussed above. However, these approaches have one potential shortcoming – the results do not update as your data changes. Thus, if your data is subject to change, you want to use a formula-based approach to ensure that your results are always current.

"Legacy" functions such as **LEFT**, **RIGHT**, and **MID** can all be helpful when attempting to manipulate text via formula. However, a newer breed of functions may be easier for some to use and provide more power. Included in these functions are **TEXTBEFORE**, **TEXTAFTER**, and **TEXTSPLIT**. We will use **TEXTBEFORE** to illustrate these functions' power and ease of use.

Examine the highlighted **TEXTBEFORE** formulas shown below in **Figure 4**. The formula's simplicity is striking – it extracts all the text before the string's first comma, working from left to right.



	A	B
1		
2	Nancy Freehafer	Nancy Freehafer, 123 Main St, Atlanta, GA
3	Andrew Cencini	Andrew Cencini, 456 Elm St, Houston, TX
4	Jan Kotas	Jan Kotas, 789 Oak Ave, Vancouver, BC
5	Mariya Sergienko	Mariya Sergienko, 1234 Hill Blvd, Los Angeles, CA
6	Steven Thorpe	Steven Thorpe, 6789 Biscayne Ln, Miami, FL
7	Michael Neipper	Michael Neipper, 135 Ave K, Calgary, AB
8	Robert Zare	Robert Zare, 79 B Central Ave, Minneapolis, MN
9	Laura Giussani	Laura Giussani, 246 Park Ave, Dallas, TX
10	Anne Hellung-Larsen	Anne Hellung-Larsen, 468 Spuce Dr, Toronto, ON
11	Anna Bedecs	Anna Bedecs, 689 Pine Ave, Portland, OR
12	Antonio Gratacos Solsona	Antonio Gratacos Solsona, 297 Vista Dr, Denver, CO
13	Thomas Axen	Thomas Axen, 638 Golden Peak, Salt Lake City, UT
14		

Figure 4 - Illustrating Excel's TEXTBEFORE Function

If, however, you wanted the formulas to extract only the city and state/province, you could use the following formula to complete the task:

=TEXTAFTER(B2,\",\",2)

The formula illustrated above extracts all the text appearing after the second comma in the string.

Finally, another way to extract the first and last names would be to use the following formula based on Excel's **TEXTSPLIT** function.

=TEXTSPLIT(B2:B13,\",\")

Of course, any of these formula-based approaches will update results dynamically as the source data changes.

Excel's Little-Known LAMBDA Function

In 2020, Microsoft unveiled the new Lambda function in Excel. Perhaps due to the release's timing (Pandemic!), the function remains unknown and unused by most Excel users. Yet, for those who have discovered this function and its' power, Lambda provides the capacity to create and store user-defined functions (UDFs) in Excel. In this article, you will learn how to take advantage of Lambda and why doing so would be a good move for most Excel users.

First, What Is A User-Defined Function?

Almost all Excel users are familiar with functions, and presently, Excel contains hundreds of functions, such as **SUM**, **IF**, **NPV**, and **XLOOKUP**. These functions underlie the calculations in row

after row and column after column of today's generation of workbooks. Seemingly, there is a function for every need.

However, instances arise where Excel does not offer a pre-defined function to meet the specific needs of some users. For example, perhaps a user needs to allocate monthly overhead expenses to multiple operational departments, such as *Finance*, *Manufacturing*, and *Executive*. Without UDFs, someone must create a formula in Excel each month to handle the allocations. However, allocating the monthly overhead becomes quick and easy by creating and saving a UDF to handle the assignment.

Notably, the concept of UDFs in Excel is not new. Before the advent of Lambda, you could (and still can) create UDFs by writing **Visual Basic for Applications** (VBA) code and saving the code as an Excel add-in. Although this process works, it is too cumbersome for many, and learning to write VBA code is not usually high on most users' priority lists.

So, to summarize the answer to the question "What is a user-defined function," the short answer is simply this: *a user-defined function is a function – similar to other Excel functions – that you can create and use in your Excel workbooks.*

How Do I Create User-Defined Functions Using Lambda?

So, what is a better option if creating UDFs is clumsy and challenging using the VBA approach outlined above? **Lambda**! In short, the feature provides a simple method for creating and saving UDFs in Excel. Specifically, Lambda allows you to create customized functions in Excel that you can save and reuse anytime. Further, when you save your UDFs in Excel, you can assign easy-to-remember names to them, so accessing them later is painless.

The Mechanics OF Creating A UDF Using Lambda

Consider the following example. You must allocate corporate overhead expenses to the Finance, Manufacturing, and Executive departments each month. Further, the allocation percentages are 19%, 48%, and 33%, respectively. You create a formula like the one shown in **Figure 5** below to perform the allocation.

```
=IF(B5="Finance",0.19*C2,IF(B5="Manufacturing",0.48*C2,IF(B5="Executive",0.33*C2)))
```

Figure 5 - Formula Used To Allocate Overhead Expenses

Of course, you must re-create this formula monthly because you use a different workbook each month to complete your close-out tasks.

Alternatively, you can use Lambda to store your formulas for future use, eliminating the need to re-create them repeatedly. Perform the following general steps to create a Lambda.

1. Create and test the formula you want to save as a Lambda so you know it performs as intended and provides accurate calculations.

2. When you're ready to save the formula as a Lambda, click **Name Manager** on the **Formula** tab of the Ribbon and then click **New**. This action initiates the process of creating a new defined name that the formula you will save as your Lambda.
3. Enter the required fields in the **New Name** dialog box and click **OK** to save your new Lambda.
4. Once you create and save your formula as a Lambda, you can use it just like any other Excel function.

A Specific Example

Returning to the example above, suppose you wanted to automate overhead allocations. You can do this by creating Lambdas and using them to distribute the expenditures. Specifically, assume you want a formula to charge 19% of overhead to the Finance team, 48% to the Manufacturing organization, and 33% to the Executive group.

To create a Lambda to allocate 19% of overhead to the Finance team, click **Name Manager** on the Ribbon's **Formulas** tab. Next, enter the following data in the formulas dialog box, enter the following data.

- **Name:** Enter the name you are assigning to your Lambda. You will use this name to call the function in your Excel workbook.
- **Scope:** Identify whether the Lambda can be used throughout the workbook in which you save it or whether you want to restrict its use to a specific worksheet within the workbook. In general, you will likely want to establish workbook-level scopes.
- **Comment:** If desired, add an optional comment describing your Lambda.
- **Refers to:** Enter the *parameters* for your Lambda and the formula used to calculate the result. A parameter is information you need to pass to your Lambda. Often, a parameter is a cell reference, but it could also be a string or a number.

To illustrate, consider the Lambda pictured in **Figure 6**. That user-defined function allocates 19% of the overhead to the Finance group. Specifically, in the **Refers to** box, the Lambda passes the Overhead parameter and a formula that multiplies that parameter by 19%.

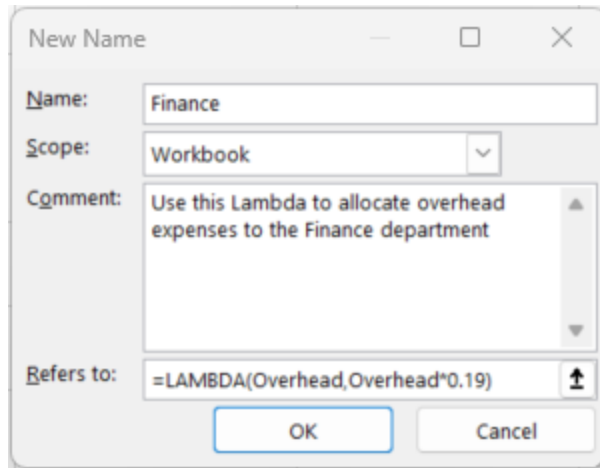


Figure 6 - Creating An Excel Lambda To Allocate Overhead

After creating a Lambda, you can view it by accessing Excel's **Name Manager**. To illustrate, consider the three Lambda's visible in the Name Manager shown in **Figure 7**. Further, you can edit or delete a Lambda from the Name Manager.

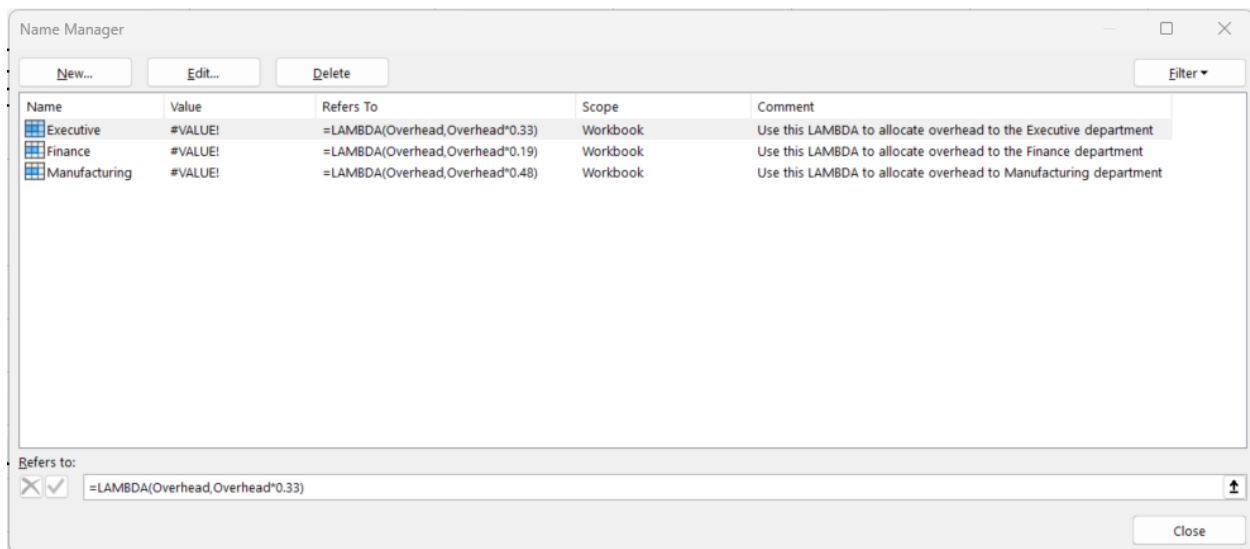


Figure 7 - Excel's Name Manager

Why Should I Create User-Defined Functions?

Hopefully, the usefulness of Lambda's is self-evident: *build a formula once and use it anywhere in a workbook without having to recreate it or copy-and-paste it*. This feature should save time and improve accuracy in your workbooks.

Further, you can enhance the usefulness of a Lambda by making it available in all future workbooks you create. If that's your desire, save your Lambdas in your *book.xltx* template file. Any Lambdas you save there will be available in all workbooks you create in the future, saving even more time and expanding the usefulness of this powerful feature.

Excel's Lambda feature is a little-known, little-used feature that offers tremendous promise and potential to eliminate creating the same formula repeatedly. With Lambda, you can create and save formulas and use them repeatedly without recreating or copying. Instead, you can use the Lambda functions you create with the same ease as you use other Excel functions, such as SUM. If you haven't yet discovered the incredible power of this feature, now is a great time to learn more about Excel's Lambda feature!

Creating Dynamic Array Formulas

The concept of *dynamic array* formulas first appeared in September 2018 and began to move mainstream for Microsoft 365 subscribers with the July 2019 release. Additionally, perpetual license users running Excel 2021 can also take advantage of dynamic arrays. Yet, regardless of the version in use, most business professionals do not use this feature or the related new functions appearing in the past five years. Those who master dynamic arrays will unlock tremendous productivity and likely work with greater degrees of accuracy in Excel. Bluntly, those who don't will fall behind.

Comparing Two Types Of Array Formulas

Generally speaking, a dynamic array formula can take action on multiple cells simultaneously by "spilling" results to multiple destination cells. Further – and very importantly, every Excel function can now generate dynamic arrays. Dynamic arrays stand in contrast to *legacy array* formulas, which require(d) users to enter a **CTRL+SHIFT+ENTER** keyboard sequence to enter a legacy array formula in Excel. Further, legacy array formulas offer no option for spilling results to multiple cells. Thus, if you want to use legacy array formulas on, for example, 1,000 rows of data, you must copy that formula 1,000 times. However, with dynamic arrays, you need only to enter the formula into one cell, and it will take action on the 1,000 rows you desire.

To illustrate, consider the dataset pictured in the Excel worksheet shown in **Figure 8**. We have 100 rows of data that we would like to round to 2 decimal places. In a legacy approach, we would likely create a formula using the ROUND function and copy that formula to all 100 rows in the worksheet.

	A	B	C	D
1				
2		984.03122	=ROUND(B2,2)	
3		20.18242		
4		212.60731		
98		715.39191		
99		269.81289		
100		348.41305		
101		657.60939		
102				
103				

Figure 8 - Legacy Approach To Creating A Formula

On the other hand, we could take advantage of the functionality offered by dynamic arrays to use a more efficient approach. Specifically, consider the example pictured in **Figure 9**. In it, we enter only one formula to round all the data in cells B2 through B101 to two decimal places. With this approach, we reduce the number of formulas in the workbook, which, in turn, should lead to improved spreadsheet performance. Further, this approach should also lead to reduced opportunities for errors simply because the number of formulas in a given workbook should decline. Couple these benefits with other new functions – such as **SORT** and **FILTER** – and it becomes clear that dynamic array formulas should be the new spreadsheet standard.

	A	B	C	D
1				
2		984.03122	=round(B2:B101,2) ^I	
3		20.18242	ROUND(number, num_digits)	
4		212.60731		
98		715.39191		
99		269.81289		
100		348.41305		
101		657.60939		
102				

Figure 9 - Simple Example Of Dynamic Arrays And Formula Reduction

Move Over, VLOOKUP; XLOOKUP Is Here!

Many Excel users now have access to **XLOOKUP**, the successor for VLOOKUP, HLOOKUP, and potentially INDEX and MATCH. XLOOKUP provides greater functionality and overcomes many of the limitations of the previous functions. For example, XLOOKUP defaults to an exact match and can perform lookups from the bottom up or the top down. Further, you no longer need to sort the lookup column or row for approximate matches. It also allows users to specify a range of cells on which to perform the lookup instead of a column number in a lookup table. Hence, the order of the table columns in a lookup table does not matter. In addition, this feature allows XLOOKUP to look to the lookup column's left – something VLOOKUP cannot do. You can see the full syntax of XLOOKUP below.

**XLOOKUP(lookup value, lookup array, return array,
[if not found], [match mode], [search mode])**

where:

Lookup value represents the value to be looked up.

Lookup array is the range or table column in which to search.

Return array is the range or table column from which to return a related value.

If not found (optional) represents what to return if the lookup value is not found; defaults to #NA.

Match mode (optional) determines how to perform the search; defaults to (0) exact match.

Option Number	Option Behavior
0 or Omitted	Exact Match (default)
-1	Exact match or next smaller item (default for VLOOKUP)
1	Exact match or next larger item
2	Wildcard character match

Search mode (optional) determines the search order; defaults to first-to-last.

Option Number	Option Behavior
1	Search first-to-last (default)
-1	Search last-to-first
2	Binary search sorted in ascending order
-2	Binary search sorted in descending order

We will use XLOOKUP to perform a reverse lookup in this first example. Since XLOOKUP uses a lookup array (column or row) instead of a lookup table, you can execute the lookup on any column (or row) and return the result from any column (or row), even columns to the left of the lookup column (or rows above.)

Nature's Softness carries an extensive inventory of quality facial products. Owner Debbie Nelson often looks up a vendor's part number when restocking a product. She uses a simple VLOOKUP formula to accomplish this task. However, she often needs to look up an internal SKU from a vendor's part number. Because the SKUs are in a column to the left of the column containing vendor part numbers, VLOOKUP cannot perform this task. Although you could use the **MATCH** and **INDEX** functions to perform this task, Debbie has limited knowledge of their use. Her assistant pointed out that the XLOOKUP function is a Swiss Army knife for data lookups and built a simple formula to retrieve the desired information.

The following formula performs a reverse lookup using a vendor's part number to retrieve a Stock Keeping Unit (SKU) from the Inventory List.

=XLOOKUP(C5,InventoryList[Vendor No],InventoryList[SKU])

However, when XLOOKUP does not find a vendor's part number in the lookup array, it returns **#NA** just like VLOOKUP. A minor change to the formula cures that shortcoming without using IFERROR. The following formula adds the fourth optional argument, what to do when the lookup

does not find the item it seeks. The argument can be a value, formula, cell reference, defined name, or text (enclosed in quotation marks). In this case, Excel will display “Item Not Found” when it does not find the vendor’s part number in the lookup array.

=XLOOKUP(C6,InventoryList[Vendor No],InventoryList[SKU],"Item Not Found")

Debbie needs to retrieve the name of the vendor when reordering a product. Unfortunately, the vendor list is in another table. XLOOKUP overcomes this problem because it works across multiple worksheets and workbooks.

**=XLOOKUP(XLOOKUP(B2,InventoryList[Item No],InventoryList[Vendor No]),
VendorList[Vendor No],VendorList[Vendor])**

Item No	Vendor No	Description	UM	QOH	Pric	Cost
C001	E1634	9 oz Aloe Vera Hand Cream	EA	2,400	6.99	2.80
C002	809834CF	9 oz Xtra Moisturizing Cream	EA	1,800	7.49	3.00
C003	E1636					
L001	N0019B					
L002	N00116B					
L003	N00124B					
L201	708901CF					
L202	N0029B					
L203	N00216B					
M101	N0039B					
M102	FA550009					
M201	FA550016					
M202	FA550024					

Vendor No	Vendor	Vendor Description
708901CF	Crème Francais	Lotion Francais, Extra Moisturizing Body, 9 oz
809834CF	Crème Francais	Cream Francais, Extra Moisturizing Hand, 9 oz
E1634	ELM Enterprises	ELM Cream, Aloe Vera, 9 oz
E1636	ELM Enterprises	ELM Cream, Hand and Body, 16 oz
FA550009	Facial Artistry LLC	Artistry Wrinkle Reducing Facial, 9 oz
FA550016	Facial Artistry LLC	Artistry Wrinkle Reducing Facial, 16 oz
FA550024	Facial Artistry LLC	Artistry Wrinkle Reducing Facial, 24 oz
N00116B	Nature's Botanicals Inc	Lotion, Organic Body, 16 oz
N00124B	Nature's Botanicals Inc	Lotion, Organic Body, 24 oz
N0019B	Nature's Botanicals Inc	Lotion, Organic Body, 9 oz
N00216B	Nature's Botanicals Inc	Lotion, Organic Hand and Body, 16 oz
N0029B	Nature's Botanicals Inc	Lotion, Organic Hand and Body, 9 oz
N0039B	Nature's Botanicals Inc	Face Mask, Organic, 9 oz

Figure 10 – XLOOKUP Can Look Across Worksheets or Workbooks

The formula shown in **Figure 10** performs a double lookup to return the value from the vendor list. It first uses XLOOKUP to find the vendor product number in the Inventory List, which is then input to XLOOKUP to return the vendor name from the Vendor List on another worksheet.

XLOOKUP can return values (as in the previous formula), arrays (ranges of values), or references (a reference to a cell or range of cells, such as B3 or A1:A10). Because XLOOKUP can search first-to-last or last-to-first, you can use it to identify the cell references of a specific product's first and last sales transactions. You can then use these values as inputs to other functions, such as SUM, COUNT, AVERAGE, MIN, or MAX. For example, **Figure 11** displays a formula to sum sales revenue by product ID from a transaction table.

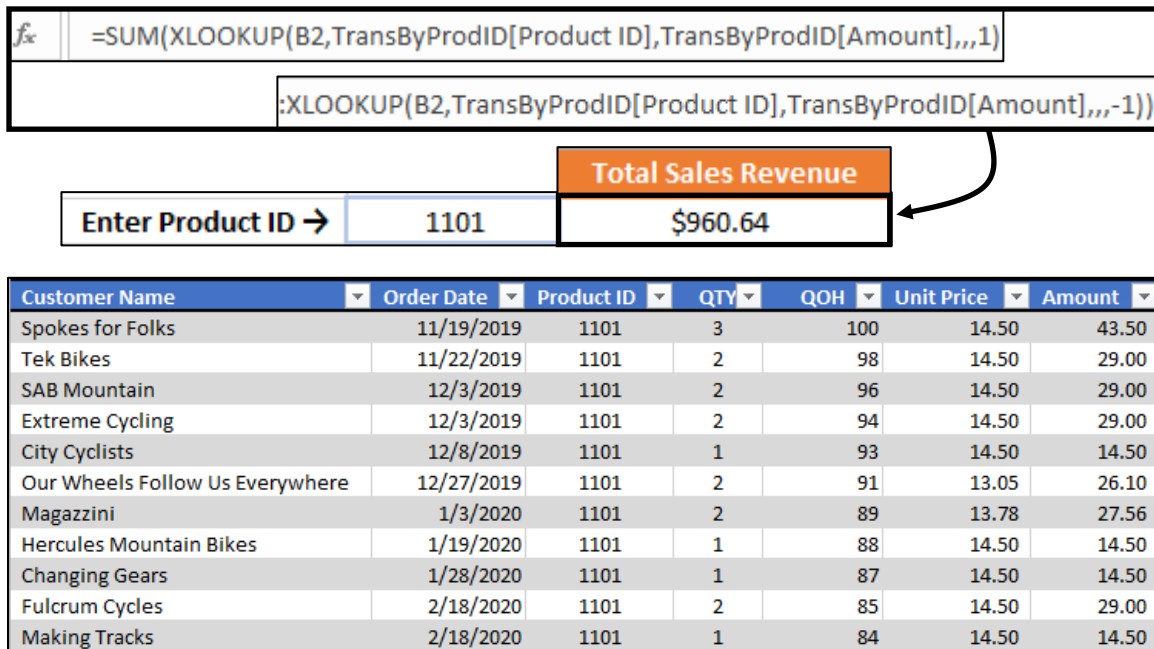


Figure 11 – Using XLOOKUP to Sum Sales of a Specified Product ID

Move over VLOOKUP; your day in the sun is over! XLOOKUP is here, and it can do everything VLOOKUP can and more. XLOOKUP and its companion XMATCH, with similar functionality advances, will revolutionize how you use lookup functions. They are easy to use, intuitive, and offer productivity-enhancing power.

Co-Authoring Using Microsoft Teams

The newest and perhaps best form of collaborating involves using **Microsoft Teams** as the platform for co-authoring. Microsoft created Teams around the idea that the most significant needs of information workers are communication and collaboration. In other words, these capabilities are not an afterthought. Instead, Microsoft incorporated them into the fundamental design of Teams.

The Teams platform takes advantage of the functionality provided by SharePoint Online to facilitate co-authoring but in a more user-friendly environment. More to the point, Microsoft enables co-authoring by default in Teams. Thus, if you open a workbook stored in Teams, and other users have the same workbook open, all users can edit the workbook and see each other's changes in real time. Notably, as shown in **Figure 12**, when you open a workbook in Teams, it opens in the Excel Web App. However, you can click **Open in Desktop App** if you require access to features not available in the Web App.

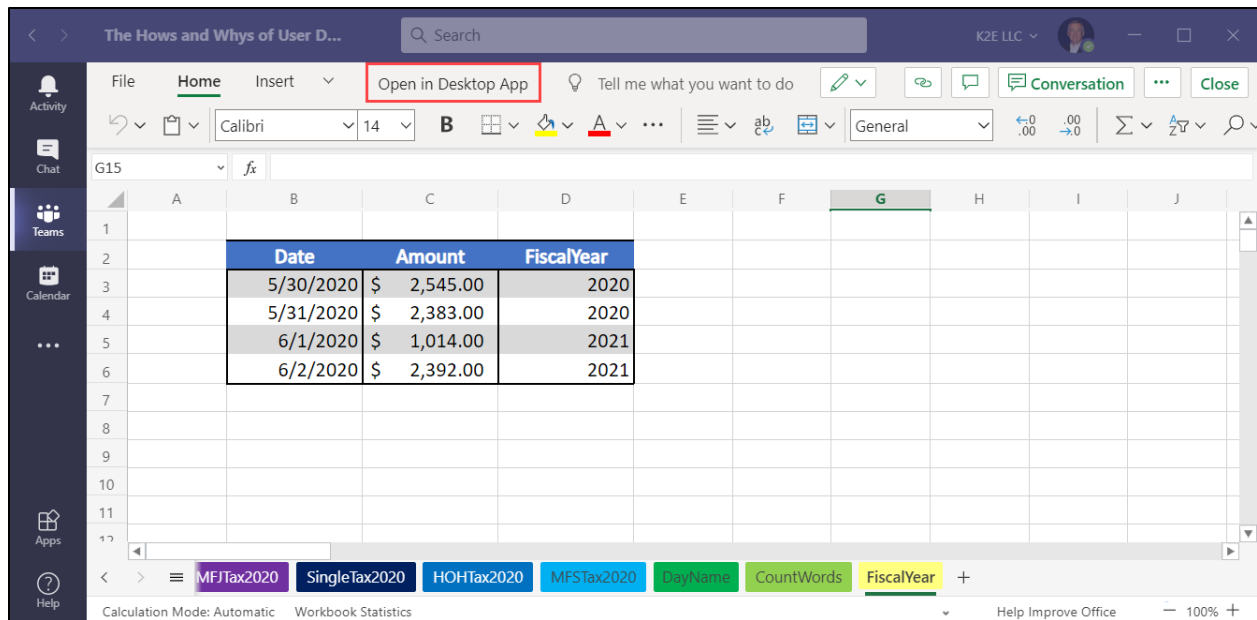


Figure 12 - Opening a Workbook in Teams Using the Excel Web App

In sum, as more users take advantage of the Teams platform, expect to see significant increases in collaboration because of the ease of doing so in Teams.

?

Suppose you must collaborate on a critical Excel document with a team member. What would be the relative merits and limitations of using the following methods?

- Sharing the document via email?
- Co-authoring the document via OneDrive or OneDrive for Business?
- Saving the document SharePoint?
- Using Excel's native Shared Workbook feature?
- Storing the workbook in Teams?

Optimizing Results With Solver

Excel's **Solver** add-in allows you to perform calculations you may have thought impossible in Excel. Specifically, Solver enables you to create models that identify optimal solutions based on the specified inputs and constraints. For example, you could use Solver in budgeting environments, resource calculations, and calculating optimal capital structures.

To illustrate Solver, consider the following example. Suppose you have a customer or client who sent a payment to you for \$14,663.10, but they did not provide a remittance advice. Further, suppose the customer has forty-two outstanding invoices with your company, meaning trying to guess which combination of invoices totals \$14,663.10 would be very time-consuming, at best. Instead, you could create a Solver model to identify which combination(s) of invoices sum to \$14,663.10.

To build your model, perform the following steps.

1. List all the invoice numbers in a single column in Excel.
2. Enter each outstanding invoice amount in an adjacent column.
3. Enter "0" in all the cells in another adjoining column.
4. Add a formula in another adjacent column, multiplying the invoice amount by the cell containing the zeroes.
5. Create a cell to rec the payment amount your customer sent you.
6. Also, create a cell that sums all the amounts of the values in the column created in Step 4 above.

Upon completing the steps above, your spreadsheet should resemble that shown in **Figure 13**.

	A	B	C	D
1	+			
2	Amount Of Payment Received	\$ 14,663.10		
3	Amount Of Payments Identified	\$ -		
4				
5	Invoice Number	Invoice Amount	Multiplier	Amount
6	5347	\$ 1,128.50	0	\$ -
7	5476	\$ 673.40	0	\$ -
44	10374	\$ 603.10	0	\$ -
45	10383	\$ 3,096.90	0	\$ -
46	10862	\$ 2,978.50	0	\$ -
47	10869	\$ 3,404.00	0	\$ -
48				

Figure 13 - Fundamental Structure Of A Solver Model

Click **Solver** on the Ribbon's **Data** tab. Note, if you do not see Solver on that tab, the most likely cause is that you need to activate the add-in. Upon clicking Solver, Excel opens the Solver Parameters dialog box pictured in

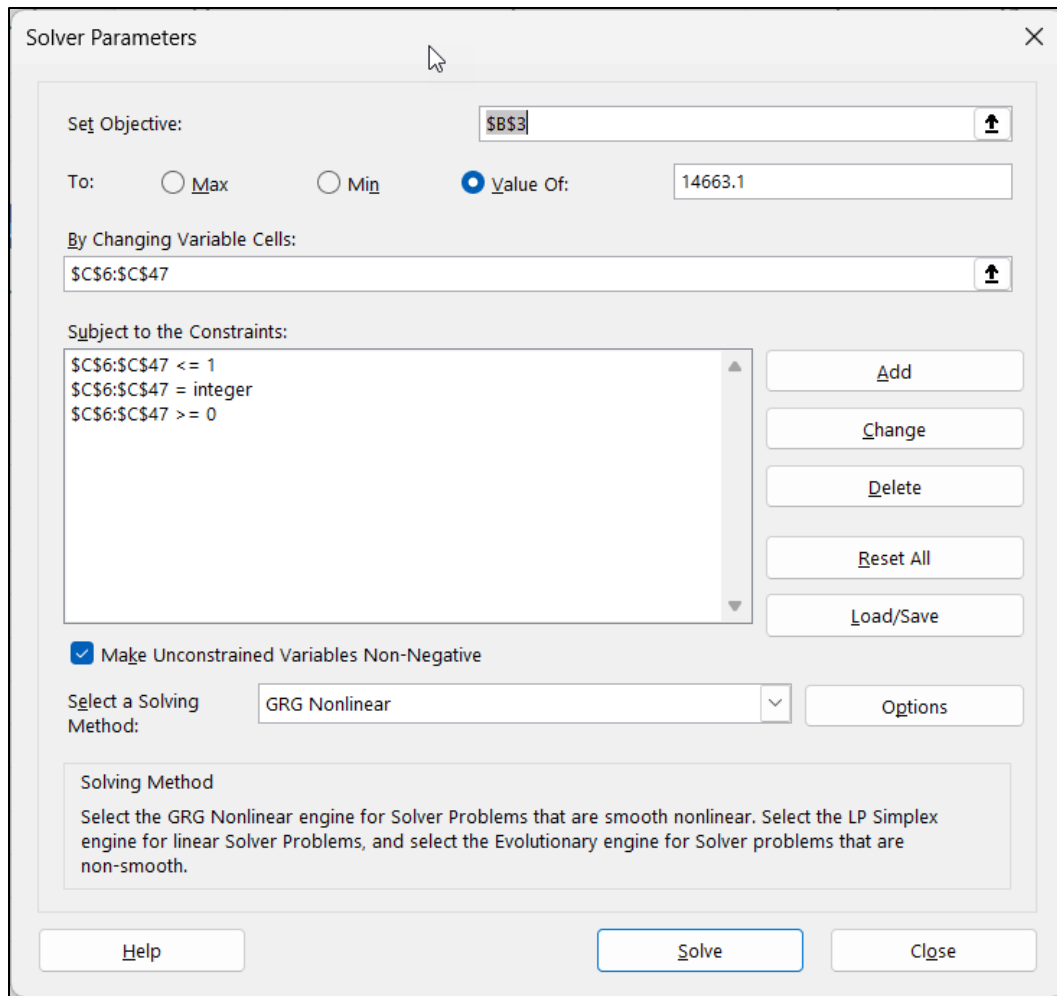


Figure 14 - Excel's Solver Parameters Dialog Box

Near the top of the dialog box, indicate that you want to set cell B3 equal to the specific value of \$14,663.10 by changing cells C6 through C47. Further, specify three constraints, as shown below.

1. Cells C6 through C47 must remain less than or equal to 1.
2. Cells C6 through C47 must remain greater than or equal to 0.
3. Cells C6 through C47 must be integers.

Upon indicating the constraints and clicking **Solve**, Excel identifies the solution shown in **Figure 15**, with some rows hidden for presentation purposes. In this case, the combination of the invoice amounts for 8033, 8340, 8449, 8509, 9373, 9433, and 9473 is the only combination that adds to \$14,663.10.

	A	B	C	D
1				
2	Amount Of Payment Received	\$ 14,663.10		
3	Amount Of Payments Identified	\$ 14,663.10		
4				
5	Invoice Number	Invoice Amount	Multiplier	Amount
6	5347	\$ 1,128.50	0	\$ -
24	8033	\$ 3,492.80	1	\$ 3,492.80
25	8105	\$ 1,287.60	0	\$ -
26	8243	\$ 2,364.30	0	\$ -
27	8340	\$ 2,038.70	1	\$ 2,038.70
28	8449	\$ 196.10	1	\$ 196.10
29	8509	\$ 2,493.80	1	\$ 2,493.80
33	9077	\$ 814.00	0	\$ -
34	9373	\$ 3,378.10	1	\$ 3,378.10
35	9433	\$ 155.40	1	\$ 155.40
36	9473	\$ 2,908.20	1	\$ 2,908.20
47	10869	\$ 3,404.00	0	\$ -

Figure 15 - Identified Solution For Solver Example

You could save your Solver solution in Excel's **Scenario Manager** if desired. Scenario Manager is helpful when you want to save multiple *virtual* versions of the same workbook instead of saving numerous *physical* copies of each workbook. You can use Scenario Manager to save assumptions and inputs associated with each workbook version. For example, in an Excel-based budgeting environment, you could save *Best Case*, *Worst Case*, and *Most Likely Case* scenarios representing the inputs associated with each budget version. Then, to switch from the workbook's Best Case version to its Worst Case version, you would not need open a different workbook or even a different worksheet. Instead, you would use Scenario Manager to insert all the necessary assumptions for the version you wish to view.

Forecasting The Future Easily With Excel

For those who engage in budgeting and forecasting activities or other activities where predicting with some degree of accuracy what will happen in the future is necessary, the **FORECAST.ETS** function added beginning with Excel 2016 is noteworthy. The original **FORECAST** function provided only linear forecasting capabilities. However, with FORECAST.ETS, you can generate forecasts with exponential smoothing, as pictured in **Figure 16**.

	A	B	C	D
28	3/31/2024	1,925,000		
29	4/30/2024	1,925,000		
30	5/31/2024	1,850,000		
31	6/30/2024	1,900,000		
32	7/31/2024	1,900,000		
33	8/31/2024	1,875,000		
34	9/30/2024	1,865,000		
35	10/31/2024	1,871,405		

Figure 16 - Using FORECAST.ETS in Excel

Additionally, you can use the new **Forecast Sheet** option on the **Ribbon's Data tab** to create a visualization of your forecast and modify parameters, including seasonality.

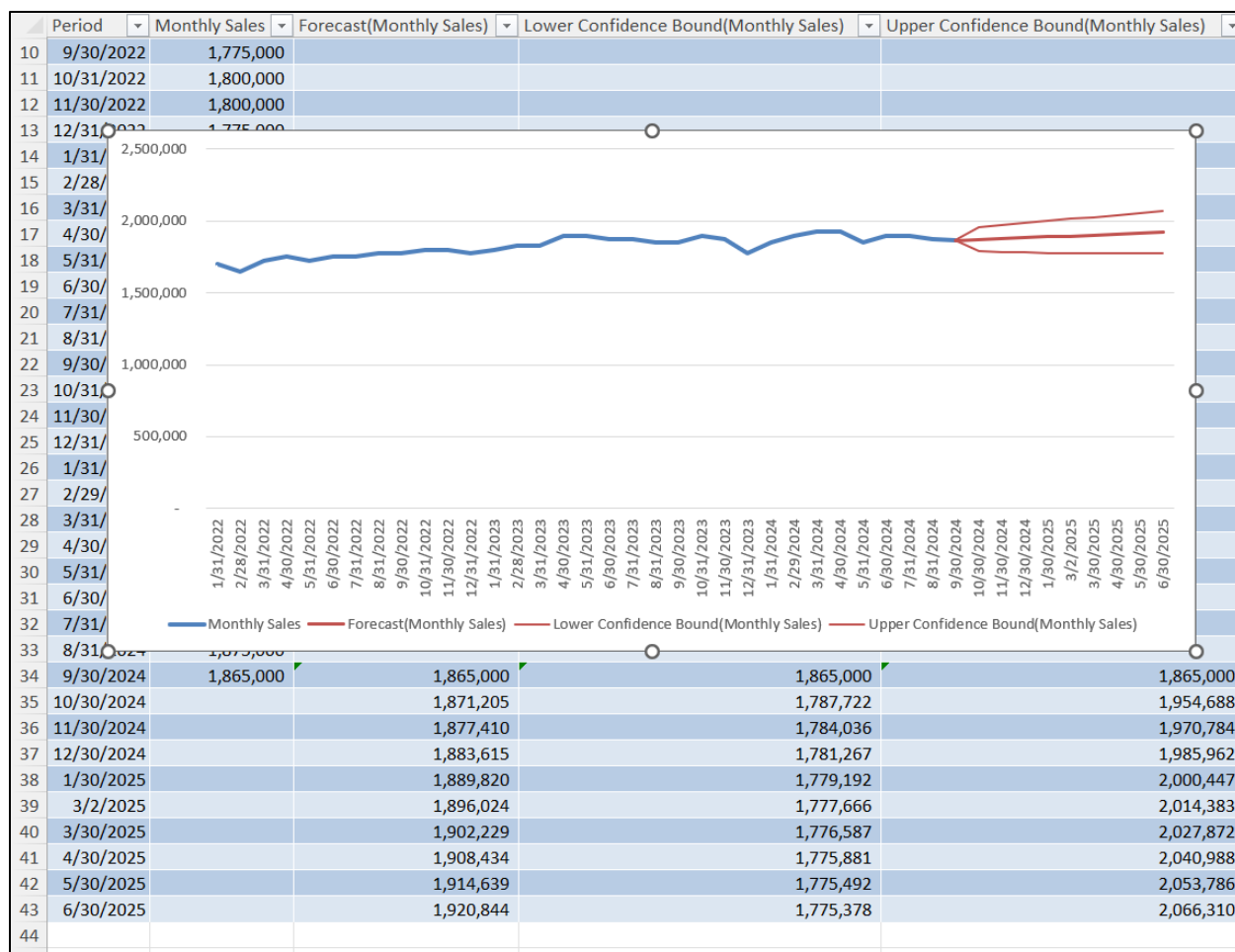


Figure 17 - Completed Forecast Sheet Generated in Excel

In addition to Forecast Sheet and Forecast.ETS, you can still access numerous Excel forecasting functions and regression analysis capabilities. For example, LINEST, LOGEST, and FORECAST are all examples of forecasting functions that might be appropriate for some users. So, for most, the forecasting options in Excel are more than adequate.

Building PivotTables The Easy Way

Many Excel users believe PivotTables are the application's singular most powerful feature. With PivotTables, we can summarize potentially vast volumes of data into clear, concise reports free of formulas. In this formula-free environment, we can work faster and with far less chance of errors. Thus, using PivotTables is widely considered a best practice for almost all Excel users. However, PivotTables can be intimidating to many users. Fortunately, Excel's Recommended PivotTables feature provides an easy way for all users – regardless of knowledge or skill level – to take advantage of PivotTables.

One way to create the desired PivotTable is to use Excel's **Recommended PivotTable** feature. With Recommended PivotTables, you can click anywhere in your data range and choose **Recommended**

PivotTables from the Ribbon's **Insert** tab. Upon doing so, Excel creates your PivotTable for you, even offering several arrangements for your report from which you can choose. This process can be highly effective when you need to create a simple PivotTable or if you are new to working with PivotTables.

Row Labels	Sum of Amount
Adams & Jefferson, CPAs	347,431.49
City Utilities	25,505.51
Farmer's Produce	171,917.97
Fresh Meats, Inc.	116,473.39
Gamble Insurance	5,725.66
RSI, Inc.	21,736.29
Washington Properties, LLC	28,800.00
Wholesale Beverage	60,434.88
Grand Total	778,025.19

Figure 18 - PivotTable Created Using Recommended PivotTables

Figure 18 shows the results of our efforts. The PivotTable summarizes the raw data into a formula-free report that summarizes expenditures by vendor. Further, all PivotTable options remain “on the table.” For example, you can add additional calculations, apply formats to the report, and even build a PivotChart from the underlying PivotTable. In other words, even though we created the PivotTable using the Recommended PivotTable approach, Excel does not limit us in any way as to how we can work with the report.

What is a PivotTable?

A PivotTable report is a cross-tabular, interactive report that extracts, organizes, and summarizes data. You can use a PivotTable to analyze data, such as making comparisons, detecting patterns and relationships, or uncovering trends. PivotTables are extremely useful for summarizing and analyzing substantial amounts of data efficiently and effectively. For example, an accountant might need to condense an Excel-based check register into a report of cash disbursements by month to prepare a financial statement. Alternatively, a public practitioner evaluating audit risk may need to summarize all bill payment checks by vendor and quarter. In both examples, users would like to drill into the summarized totals in the report to see the detailed transactions that underlie the calculations. PivotTables can provide this functionality – and much more! – quickly and with computational accuracy.

Primary Benefits Associated with PivotTables

PivotTables provide numerous benefits for those who choose to use them. Among these are:

- Automatic summarization of data, using eleven summarization options, reducing dependency on formulas;
- Drill-through capabilities to underlying details;
- The ability to summarize data across multiple dimensions;
- Automatic recognition of dates in PivotTables, with related automatic summarization by different date levels;
- The availability of predefined formatting options;
- The ability to add a user-defined calculation to PivotTables;
- Creating PivotCharts to express the data summarized in a PivotTable in a graphical environment; and
- Sorting and filtering data quickly and easily.

Although each of the benefits outlined above is beyond the scope of this session, it should become abundantly clear to all that PivotTables are Excel's best feature and, equally important, one of its most easy-to-use features.

The Six Main Elements of a PivotTable

Excel uses specific terms to identify the elements of a PivotTable report. The six primary components, shown in **Figure 19**, are filters, values fields, columns, items, rows, and the data area.

	A	B	C	D	E
1	Vendor	(All)			
2					
3	Sum of Amount	Quarter			
4	Account	Qtr1	Qtr2	Qtr3	Qtr4
5	Beverages	14,620.76	14,630.86	16,476.61	14,706.65
6	Food	72,545.64	70,502.27	69,699.27	75,644.18
7	Insurance	1,281.99	1,281.99	1,521.07	1,640.61
8	Payroll Expense	75,715.47	83,907.75	90,288.01	97,520.26
9	Rent	7,200.00	7,200.00	7,200.00	7,200.00
10	Supplies	4,778.96	6,032.00	5,235.41	5,689.92
11	Utility Expense	5,130.76	7,217.52	7,949.89	5,207.34

Figure 19 – The Six Primary Elements of a PivotTable

1. **Filters** are fields from the source data that act as filters in a PivotTable report. In the example presented in Figure 19, *Vendor* is a filter. You can use the *Vendor* field to display data from a single vendor or multiple vendors. Filters are optional, and you can use numerous filters if desired.
2. **Values Fields** are fields from the source data that contain the values you will summarize in your PivotTable. For example, *Amount* is a values field in Figure 19. You can choose how to compile the numerical data based on eleven options, including sum, count, and average. You can count the times a specific text entry appears in a field for text data.
3. **Columns** are fields from the source data you assign to a vertical layout. For example, *Quarter* is a column field in Figure 19.
4. **Items** are the subcategories of a row, column, or report filter. In this example, the *Quarter* field contains four elements: *Qtr1*, *Qtr2*, *Qtr3*, and *Qtr4*. Further, the *Account* field includes seven items: *Beverages*, *Food*, *Insurance*, *Payroll Expense*, *Rent*, *Supplies*, and *Utility Expense*.
5. **Rows** are fields from the source data that make up the row layout in a PivotTable. As an example, *Account* is a row field in Figure 19.

6. The **Data Area** is the range of cells in a PivotTable report containing summarized data. In Figure 19, cells B5 through E5 represent the *Data Area*. For example, the value in cell B5 totals sales for *Beverages* and *Qtr1*. In other words, cell B5 summarizes the cash disbursements for every row in the source data that contains disbursement data in the first quarter and beverage expenditures.

Now that we understand what a PivotTable is and are familiar with Microsoft's terminology to describe a PivotTable's various elements, let us turn our attention to creating PivotTables.

Creating Simple PivotTables

Even the simplest PivotTables provide a substantial portion of the tool's productivity and information analysis benefits. In other words, you need not be a PivotTable expert or know how to apply the advanced features of PivotTables to take advantage of their power and functionality. *The Law of Diminishing Returns applies!* As our PivotTables become more complex, the amount of learning required to take advantage of their more complex functionality increases. For most analyses, a simple PivotTable that auto-summarizes or auto-tabulates our data will benefit and provide all the power we need.

The table in **Figure 20** contains a restaurant owner's Excel-based check register maintained in Excel. Each row includes the details of checks written during the year. The spreadsheet consists of five data columns – *Date*, *Check Number*, *Amount*, *Vendor*, and *Account*. This data serves as the data source for our first PivotTable examples.

The sample data set includes the details of 430 checks, although other data sets may have thousands, tens of thousands, hundreds of thousands, or even millions of rows. Further, the data that serves as the foundation of PivotTables need not reside in Excel. Practically, there are no limits on the amount of data you can summarize in a PivotTable, but the more information there is, the greater the power of PivotTables. However, arranging your data in a tabular fashion – with column headers and, preferably, no blank rows – is imperative to facilitate quick-and-easy PivotTables.

	A	B	C	D	E
1	Date	Check Number	Amount	Vendor	Account
2	1/1/2021	2154	\$ 1,395.29	Fresh Meats, Inc.	Food
3	1/2/2021	2155	\$ 998.13	Wholesale Beverage	Beverages
4	1/2/2021	2156	\$ 1,221.47	Fresh Meats, Inc.	Food
5	1/2/2021	2157	\$ 6,234.56	Adams & Jefferson, CPAs	Payroll Expense
6	1/3/2021	2158	\$ 1,341.25	Farmer's Produce	Food
7	1/3/2021	2159	\$ 2,400.00	Washington Properties, LLC	Rent
426	12/27/2021	2578	\$ 1,245.76	Fresh Meats, Inc.	Food
427	12/27/2021	2579	\$ 439.59	RSI, Inc.	Supplies
428	12/28/2021	2580	\$ 1,584.08	Farmer's Produce	Food
429	12/28/2021	2581	\$ 7,039.73	Adams & Jefferson, CPAs	Payroll Expense
430	12/30/2021	2582	\$ 1,223.16	Fresh Meats, Inc.	Food
431	12/31/2021	2583	\$ 802.33	Fresh Meats, Inc.	Food

Figure 20 - Data for Simple PivotTable Examples

In our first example, we would like to tabulate cash disbursement figures by account and by date of expenditure to create a set of financial statements. Our completed PivotTable will have rows for the Account and Date for the columns.

One easy way to create the desired PivotTable is to use Excel's **Recommended PivotTable** feature. With Recommended PivotTables, you can click anywhere in your data range and choose **Recommended PivotTables** from the **Ribbon's Insert tab**. Upon doing so, Excel will automatically create your PivotTable for you, even offering several arrangements for your report from which you can choose. This process can be highly effective when hurrying to construct a simple PivotTable.

Alternatively, by performing the following steps, you can create a PivotTable using a more traditional method.

1. Place the cursor in the data and choose **PivotTable** from the **Insert** tab, as shown in **Figure 21**.

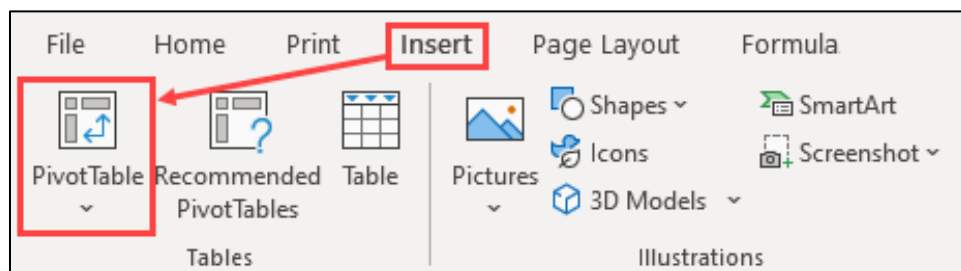


Figure 21 - Select PivotTable from the Insert Tab

2. In the resulting **Create PivotTable** dialog box, shown in

3. Figure 22, the data range should already reside in the dialog box. If not, click the **Collapse Dialog** button and select the data range. Ensure your entire data range is selected, including the field names at the top of the columns. Next, select **New Worksheet** and then click **OK**. Note that a Table can serve as a dynamic data range for a PivotTable.

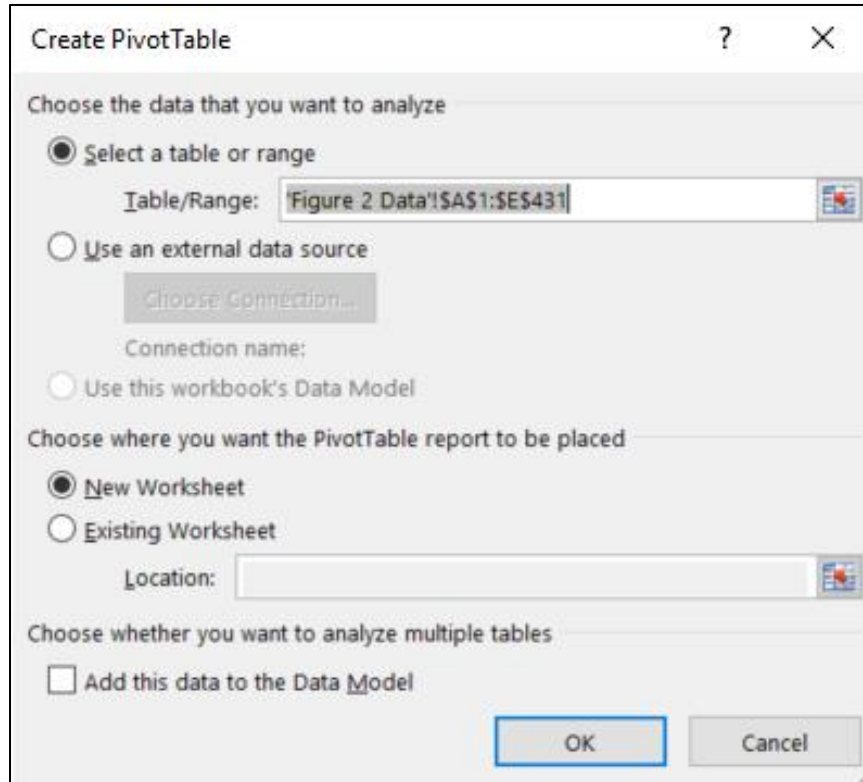


Figure 22 - Highlighting the Data Range and Selecting New Worksheet

4. A new worksheet like the one shown in **Figure 23** will appear in the workbook. On the left side of the worksheet is a placeholder for the PivotTable. On the right side is the **PivotTable Task Pane**. You can customize the layout of the task pane to meet your needs. To do so, click the drop-down button in the top right corner of the task pane just below the title bar.
5. The task pane contains the **PivotTable Field List** at the top and a quadrant of boxes into which fields are dragged from the list to create the PivotTable. The quadrants correspond to the four areas of a PivotTable report: *Filters*, *Columns*, *Rows*, and *Values*.

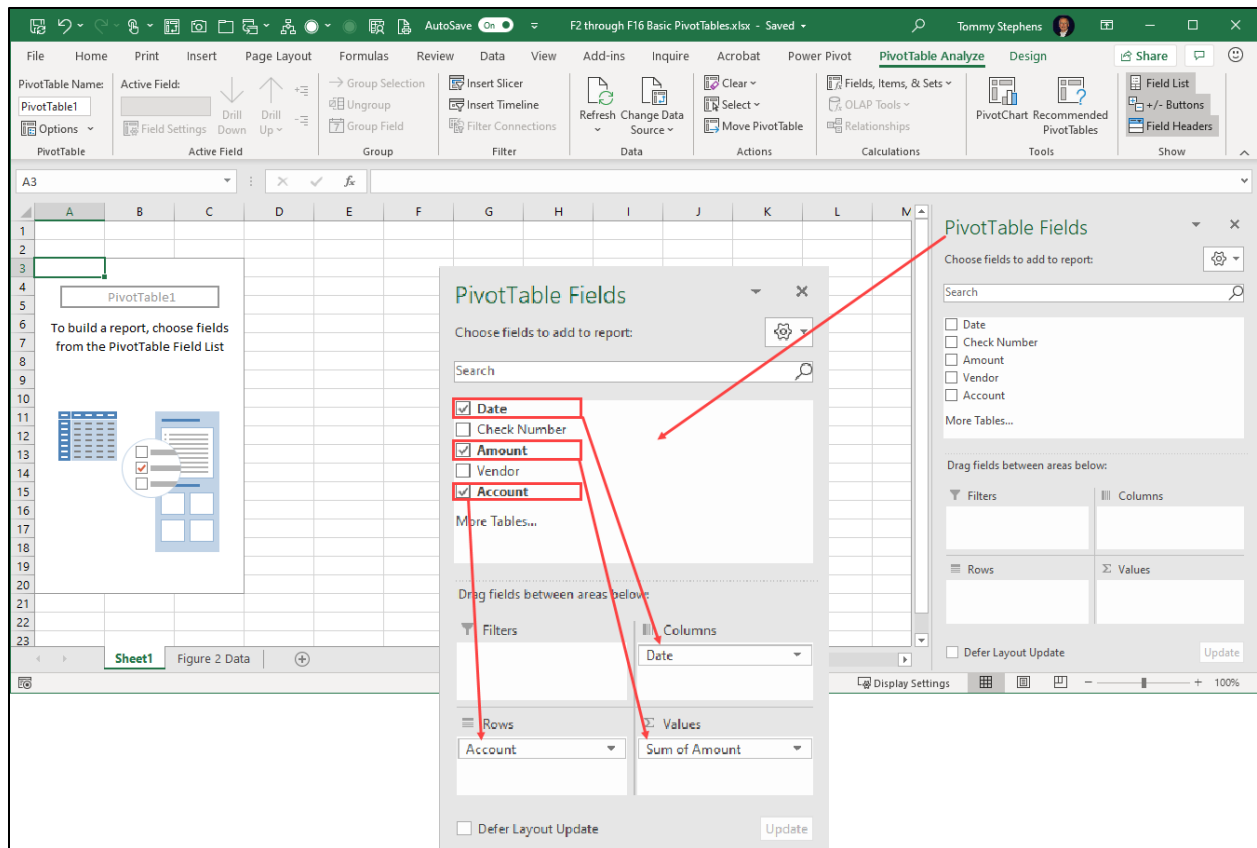


Figure 23 - Creating a PivotTable by Dragging Fields to the Quadrants

6. Drag each desired field from the list to the report quadrants, as shown in Figure 23.
7. Click on the **Date** field, hold down your left mouse button, drag and point to the **Columns** quadrant, and release your left mouse button.
8. Click on the **Account** field, hold down your left mouse button, drag and point to the **Rows** quadrant, and rerelease your left mouse button.
9. Click on the **Amount** field, hold down your left mouse button, drag and point to the **Values** quadrant, and release your left mouse button.

We have just created our first PivotTable using the traditional process. **Figure 24** shows the results of our efforts. The table summarizes the raw data into a two-dimensional table with *Date* across the top and *Account* down the left margin. Also, note how Excel totaled each row and column automatically. Further, once we add grouping options to the PivotTable to group the individual dates into months and quarters, the PivotTable will become more valuable.

	A	B	C	IS	IT	IU
1						
2						
3	Sum of Amount	Column Labels ▼				
4	Row Labels ▼	1-Jan	2-Jan	30-Dec	31-Dec	Grand Total
5	Beverages		998.13			60434.88
6	Food	1395.29	1221.47	1223.16	802.33	288391.36
7	Insurance					5725.66
8	Payroll Expense		6234.56			347431.49
9	Rent					28800
10	Supplies					21736.29
11	Utility Expense					25505.51
12	Grand Total	1395.29	8454.16	1223.16	802.33	778025.19

Figure 24 - Simple PivotTable Report Created with a Few Clicks

Grouping and Ungrouping Data

Now, let us examine some of the power of PivotTables. Let us group the dates into months and quarters to calculate and display monthly and quarterly totals for each account. Position the cursor in a cell containing one of the date column headings, such as **1-Jan**. Right-click and choose **Group** to open the **Grouping** dialog box. In the **By** box, click **Months** and **Quarters**, as shown in **Figure 25**. Click **OK** to complete the task.

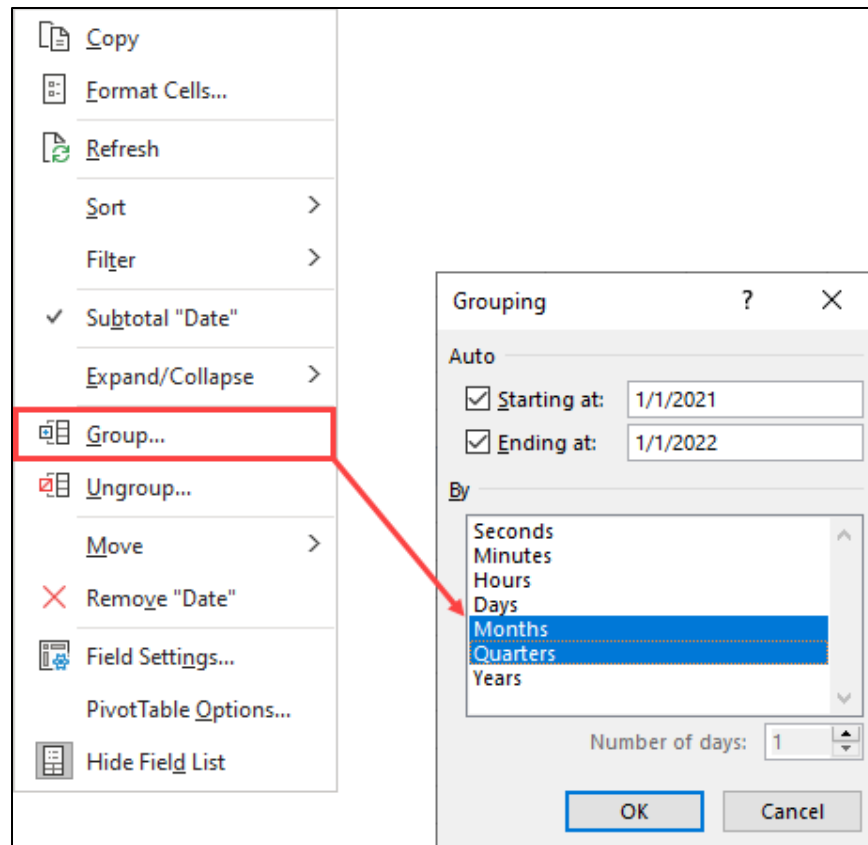


Figure 25 - Grouping Dates into Months and Quarters



In newer versions of Excel, you will notice that Excel offers the option of automatically creating date groupings whenever you add fields containing dates to your PivotTables. This feature can reduce or eliminate the need to group data manually, as described above.

Next, position the cursor in a cell containing one of the quarterly column headings, such as **Qtr1**. Then, right-click on the cell and select **Subtotal "Quarters"** from the context-sensitive menu to add a quarterly subtotal following each calendar year quarter. **Figure 26** shows the initial results, with only the first two quarters presented.

	A	B	C	D	E	F	G	H	I
1									
2									
3	Sum of Amount	Column Labels ▾							
4		⊖ Qtr1			Qtr1 Total	⊖ Qtr2			Qtr2 Total
5	Row Labels ▾	Jan	Feb	Mar		Apr	May	Jun	
6	Beverages	5098.91	4448.02	5073.83	14620.76	4966.58	4902.95	4761.33	14630.86
7	Food	25562.79	23221.12	23761.73	72545.64	24076.4	22816.73	23609.14	70502.27
8	Insurance	427.33	427.33	427.33	1281.99	427.33	427.33	427.33	1281.99
9	Payroll Expense	13658.24	32601.41	29455.82	75715.47	27649.05	27833.66	28425.04	83907.75
10	Rent	2400	2400	2400	7200	2400	2400	2400	7200
11	Supplies	1510.8	1632	1636.16	4778.96	2075.89	2218.97	1737.14	6032
12	Utility Expense	1541.68	2156.87	1432.21	5130.76	2794.58	1301.54	3121.4	7217.52
13	Grand Total	50199.75	66886.75	64187.08	181273.58	64389.83	61901.18	64481.38	190772.39

Figure 26 - Results of Grouping Months into Quarters

Again, position the cursor in a cell containing one of the quarterly column headings, such as **Qtr1**. Right-click on the cell and select **Expand/Collapse, Collapse Entire Field** from the context-sensitive menu. Months collapse into Quarters, and all the totals now reflect quarterly totals, as shown in **Figure 27**. You can control the detail shown in a field from the context-sensitive menu.

	A	B	C	D	E	F
1						
2						
3	Sum of Amount	Column Labels ▾				
4		⊕ Qtr1	⊕ Qtr2	⊕ Qtr3	⊕ Qtr4	Grand Total
5	Row Labels ▾					
6	Beverages	14620.76	14630.86	16476.61	14706.65	60434.88
7	Food	72545.64	70502.27	69699.27	75644.18	288391.36
8	Insurance	1281.99	1281.99	1521.07	1640.61	5725.66
9	Payroll Expense	75715.47	83907.75	90288.01	97520.26	347431.49
10	Rent	7200	7200	7200	7200	28800
11	Supplies	4778.96	6032	5235.41	5689.92	21736.29
12	Utility Expense	5130.76	7217.52	7949.89	5207.34	25505.51
13	Grand Total	181273.58	190772.39	198370.26	207608.96	778025.19

Figure 27 - Monthly Data Grouped into Quarters

Double-click on the cell of interest to see all the records underlying any calculation. Excel will insert a new sheet containing the detailed observations that created the cell contents. For example, if you click on cell D8, which includes an *Insurance* total of \$1,521.07 in our sample PivotTable, Excel will display the underlying detail, as shown in **Figure 28**. Note that Excel presents the data in Table form and format, making it easy to filter and analyze.

	A	B	C	D	E
1	Date ▾	Check Number ▾	Amount ▾	Vendor ▾	Account ▾
2	7/15/2021	2381	427.33	Gamble Insurance	Insurance
3	8/21/2021	2417	546.87	Gamble Insurance	Insurance
4	9/24/2021	2460	546.87	Gamble Insurance	Insurance

Figure 28 - Drilling to Detail from a PivotTable

Custom Groups

In the example, the PivotTable summarized data using built-in groups, such as consolidating months into quarters. However, Excel allows you to create user-defined groups. First, highlight the items you wish to group to accomplish this task. Then, right-click and select **Group** from the context-sensitive menu.

In the example PivotTable, we would like to group the accounts into two classifications for financial statement presentation – *Cost of Goods Sold* and *Operating Expenses*. Food and Beverages comprise the Cost of Goods Sold group, while all other items comprise the Operating Expenses group. Click on **Beverages**, hold down the **CTRL** key, and then click **Food** to create the groups. Next, right-click and choose **Group** from the context-sensitive menu. This action combines the first two accounts into one group entitled **Group1**. Rename the **Group1** label to **Cost of Goods Sold** to complete the first group, as shown in **Figure 29**. You can rename to rename by typing over it with the desired text.

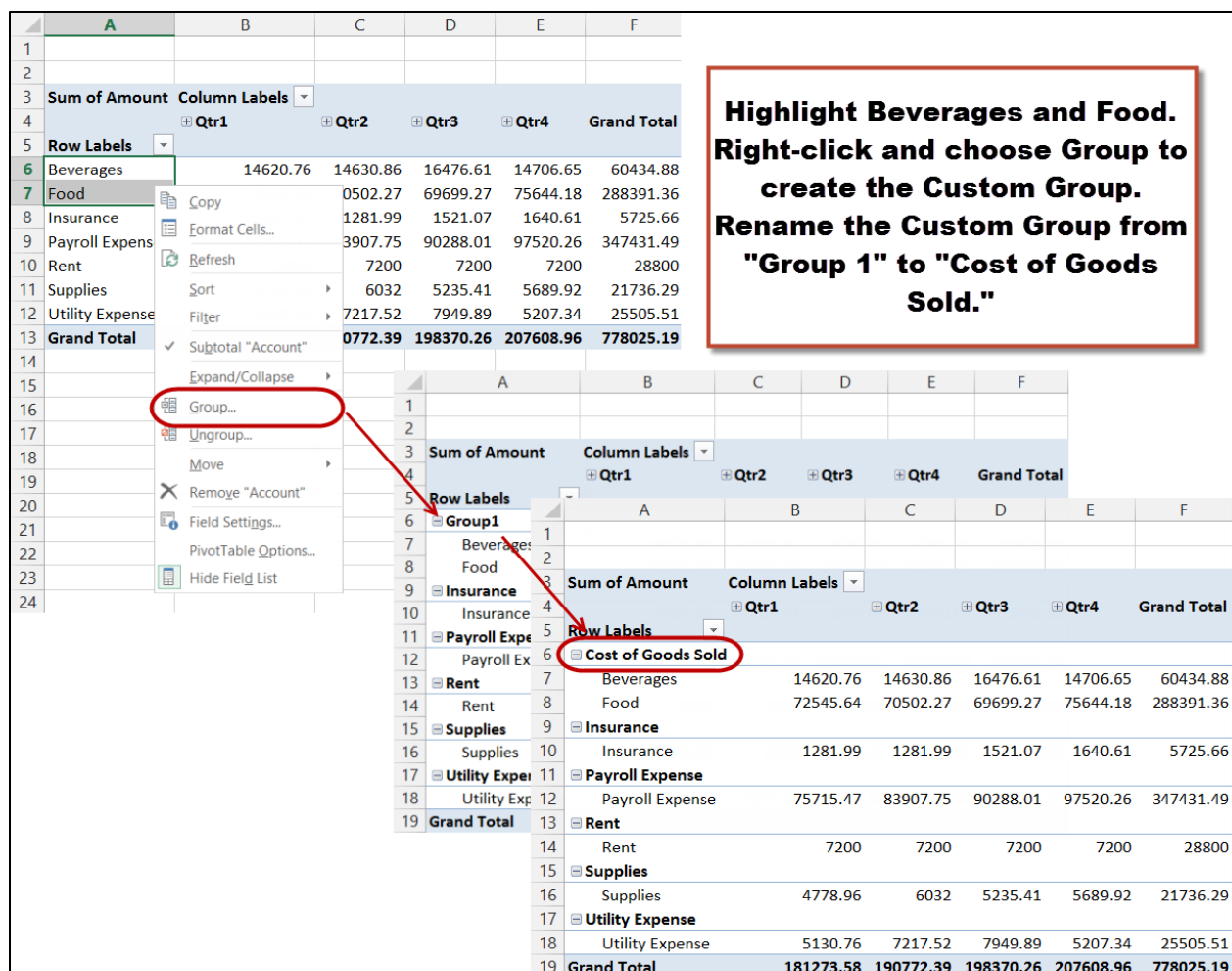


Figure 29 - Grouping Accounts into Account Groups

Repeat the steps for the remaining group, **Operating Expenses**. To complete the PivotTable, add subtotals for both groups. Position the cursor in a cell containing one of the group row headings, such as **Cost of Goods Sold**. Right-click on the cell and select **Subtotal "Account2"** from the context-sensitive menu to add a subtotal for the group. The completed report should resemble the one shown in **Figure 30**.

	A	B	C	D	E	F
1						
2						
3	Sum of Amount	Column Labels ▼				
4		⊕ Qtr1	⊕ Qtr2	⊕ Qtr3	⊕ Qtr4	Grand Total
5	Row Labels ▼					
6	▢ Cost of Goods Sold	87166.4	85133.13	86175.88	90350.83	348826.24
7	Beverages	14620.76	14630.86	16476.61	14706.65	60434.88
8	Food	72545.64	70502.27	69699.27	75644.18	288391.36
9	▢ Operating Expenses	94107.18	105639.26	112194.38	117258.13	429198.95
10	Insurance	1281.99	1281.99	1521.07	1640.61	5725.66
11	Payroll Expense	75715.47	83907.75	90288.01	97520.26	347431.49
12	Rent	7200	7200	7200	7200	28800
13	Supplies	4778.96	6032	5235.41	5689.92	21736.29
14	Utility Expense	5130.76	7217.52	7949.89	5207.34	25505.51
15	Grand Total	181273.58	190772.39	198370.26	207608.96	778025.19

Figure 30 - PivotTable with Account Groups and Subtotals



Note that you can rename the default field name – in this case, **Account2** – given by Excel to the account group in **Field Settings**. Right-click on any group label and select **Field Settings** from the context-sensitive menu. Type the desired field name, such as **Account Group**, in the **Custom Name** box and click **OK** in the **Field Settings** dialog box.

You can create custom groups by selecting and grouping any item within a field. Further, you may need to create custom groups when built-in groups are inconsistent with a company's practices. For example, when grouping months into quarters, the predefined groups use a calendar year to define the quarters. However, a custom grouping based on fiscal periods could be necessary if a company uses a fiscal year instead.

Pivoting Columns and Rows

You can re-arrange – “pivot” – the summarized table to show the data from virtually any perspective. This feature is among the most powerful capabilities of PivotTables. For example, if you wanted to display Accounts as columns and Quarterly Totals as rows, you would re-arrange the rows and columns by dragging and dropping the fields into the quadrant boxes, as detailed below.

1. Position the cursor in the PivotTable report to display the **PivotTable Task Pane**. In the field list at the top of the task pane, uncheck **Date** and **Account Group** to remove them from the report. Next, click **Quarters**, drag it into the **Rows** quadrant, and drop it below

Account. This action has the effect of reporting Quarters within Accounts, as shown in **Figure 31**.

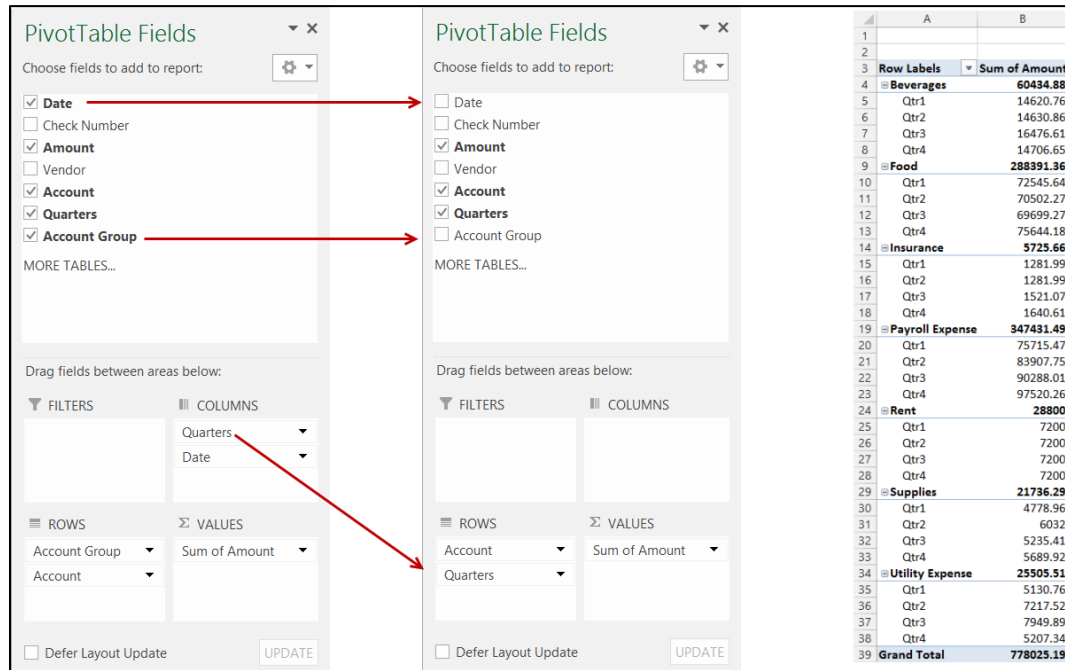


Figure 31 - Dragging Fields in the PivotTable Task Pane to Re-arrange a Report

- Now, click on **Account** and drag and drop it into the **Columns** quadrant. This step re-arranges the report so that Accounts are the columns and Quarters are the rows, as shown in **Figure 32**.

PivotTable Fields

Choose fields to add to report:

- ☐ Date
- ☐ Check Number
- ☒ Amount
- ☐ Vendor
- ☒ Account
- ☒ Quarters
- ☐ Account Group

MORE TABLES...

Drag fields between areas below:

FILTERS

COLUMNS

ROWS

VALUES

Account

Quarters

Sum of Amount

☐ Defer Layout Update

UPDATE

PivotTable Fields

Choose fields to add to report:

- ☐ Date
- ☐ Check Number
- ☒ Amount
- ☐ Vendor
- ☒ Account
- ☒ Quarters
- ☐ Account Group

MORE TABLES...

Drag fields between areas below:

FILTERS

COLUMNS

Account

ROWS

Quarters

VALUES

Sum of Amount

☐ Defer Layout Update

UPDATE

	A	B	C	D	E	F	G	H	I
1									
2									
3	Sum of Amount	Column Labels							
4	Row Labels	Beverages	Food	Insurance	Payroll Expense	Rent	Supplies	Utility Expense	Grand Total
5	Qtr1	14620.76	72545.64	1281.99	75715.47	7200	4778.96	5130.76	181273.58
6	Qtr2	14630.86	70502.27	1281.99	83907.75	7200	6032	7217.52	190772.39
7	Qtr3	16476.61	69699.27	1521.07	90288.01	7200	5235.41	7949.89	198370.26
8	Qtr4	14706.65	75644.18	1640.61	97520.26	7200	5689.92	5207.34	207608.96
9	Grand Total	60434.88	288391.36	5725.66	347431.49	28800	21736.29	25505.51	778025.19

Figure 32 - Completing the Drag-and-Drop Rearrangement of a Report

To review, we merely dragged the field buttons to different quadrants in the PivotTable Task Pane to re-arrange the report. The number of ways we can view a PivotTable is limited only by the number of fields we have to display and our imagination.

At first glance, the power of PivotTables to reduce the time and tedium of preparing routine reports is significant. However, the real power of PivotTables is not clerical; it is the added ability they provide to view data from different perspectives. By viewing data from different perspectives, PivotTables

increase the probability of uncovering relationships in the data that were previously unknown. With these relationships exposed, we can exploit them on behalf of the organizations we serve. Now, that's analytical power!

Managing Data Using Power Query

Excel's **Power Query** feature is one of the most significant innovations and additions made to Excel throughout the application's history. Power Query serves two primary purposes. First, Power Query easily links external data into Excel. Second, Power Query provides options for you to create transformations in the linked data to make the data more useful to you. And, because Power Query links the data – instead of copying and pasting it – whenever you refresh a query, all your transformations update. Collectively, these two features mean that Power Query helps you automate many of your Excel-based reporting and analytical tasks.

If you use Excel 2016 or newer, the functionality provided by Power Query is available in the **Get & Transform** group on the **Data** tab of the Ribbon. Microsoft supports Power Query in all versions of Excel 2016 and newer. Further, it does not matter whether you run a 32-bit or a 64-bit version of Excel; you have access to Power Query.

One of the most crucial advantages of Power Query is that it facilitates queries from many data sources, including most of the primary data sources in use by accounting and financial professionals. In addition, you can query data from “non-standard” data sources such as PDF documents, Exchange, and Salesforce. Further, and we stress this point, Power Query provides robust capabilities to extract and link data from each source. These capabilities mean that even users with limited technical skills can extract data from their accounting, ERP, CRM, practice management, or other line-of-business databases into Excel for reporting and data analysis quickly and with relative ease. Moreover, Power Query's capabilities allow you to automate transforming – cleaning up – your data as part of the query process.

To transform the source data so that the query results are acceptable, click the **Transform Data** button near the window's lower right corner to open the **Query Editor** shown in **Figure 33**. The Query Editor provides tools to transform the data before executing the query to align the results with expectations. Examples of some of the transformations you can make with the Query Editor include adding and deleting columns of data, changing data types, merging columns, splitting columns, renaming columns, transposing data, and adding user-defined calculations.

Power Query stores these transformations as you add them using the **Query Editor**. The next time you use or refresh your query, the stored changes apply automatically to your data, relieving you of the task of manually re-performing your modifications. In the example shown, the user is transforming the **Zipcode** field to reformat it as text so that US postal codes with leading zeros display five characters. Without such a transformation, these codes would show only four characters. Upon completing this and any additional modifications, click the Query Editor's **Home**

tab and **Close & Load** to load the transformed data to Excel. **Figure 34** illustrates the completed data query, including the transformation of the Zipcode field.

Notably, with the transformations saved as part of your query, Excel applies them automatically each time you refresh your data. Additionally, recognize that using changes is not limited to text files. Instead, you can transform any data you link into Excel with Power Query.

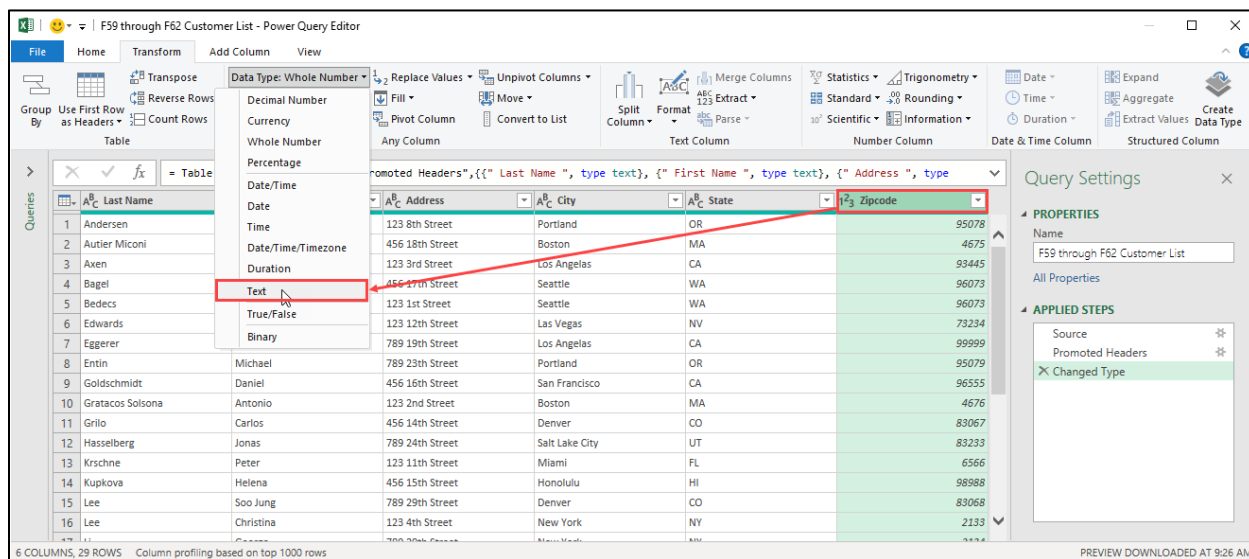


Figure 33 - Using Power Query to Transform Data Before Completing the Query

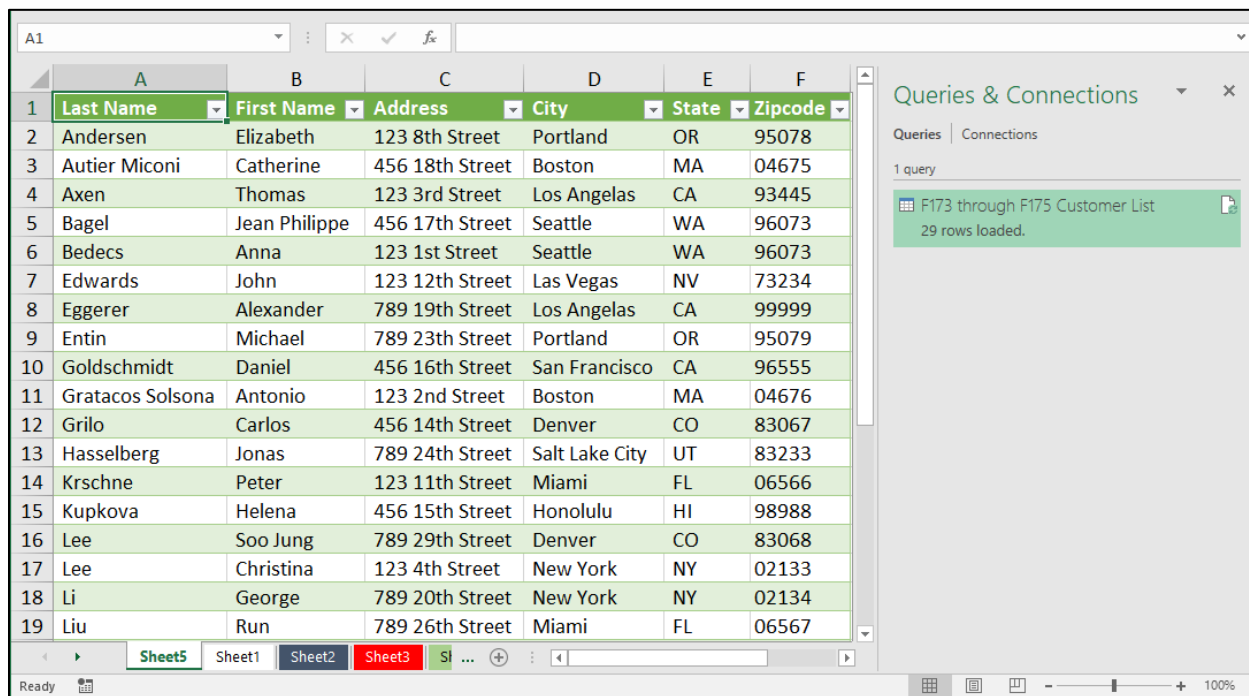


Figure 34 - Completed Text Query, Including Data Transformation

Importing Data From Pictures And PDFs

Business professionals often come across non-Excel documents containing numerical data they need in Excel. Unfortunately, most instinctively turn to the tried-and-true process of re-keying the data into Excel – a tedious, time-consuming, and error-prone task. That need not be the case!

Previously, we discussed Excel's Power Query tool as a way of cleaning up, transforming data, and manipulating data feeding into an Excel workbook. An often overlooked feature associated with Power Query is the tool's ability to import data from PDF documents and pictures, eliminating manual data entry in many cases.

Importing Data From A PDF

Consider the report pictured in **Figure 35**. It is a PDF document that contains operational data from a small chain of restaurants. We have hidden six months of data from the display for presentation purposes.

Month Ending Date	Restaurant Number	Sales	Payroll Expense
1/31/2022	101	1,172,329	215,709
1/31/2022	102	960,554	157,531
1/31/2022	103	1,169,956	180,173
1/31/2022	104	989,238	212,686
2/28/2022	101	1,170,269	227,032
2/28/2022	102	973,438	205,395
2/28/2022	103	1,116,058	179,685
2/28/2022	104	1,051,097	230,190
3/31/2022	101	1,191,158	200,115
3/31/2022	102	950,959	165,467
3/31/2022	103	1,180,638	204,250
3/31/2022	104	1,189,884	188,002
10/31/2022	101	1,071,390	162,851
10/31/2022	102	1,095,797	168,753
10/31/2022	103	938,998	205,641
10/31/2022	104	1,105,036	212,167

Figure 35 - Sample Sales Report for Importing into Excel Using Power Query

Suppose you need the data from the import available to analyze operations and results in Excel. However, you do not want to spend time manually entering 160 cells into a workbook! Fortunately, you don't have to because Power Query can extract the data from the PDF document and place it into the spreadsheet.

In Excel, click **Data** from the Ribbon's Home tab to take advantage of this feature, followed by **Get Data, From File**, and **From PDF**. Then, select the PDF document containing the data you need to extract in the dialog box and click **Import** to open Power Query's **Navigator**, shown in **Figure 36**.

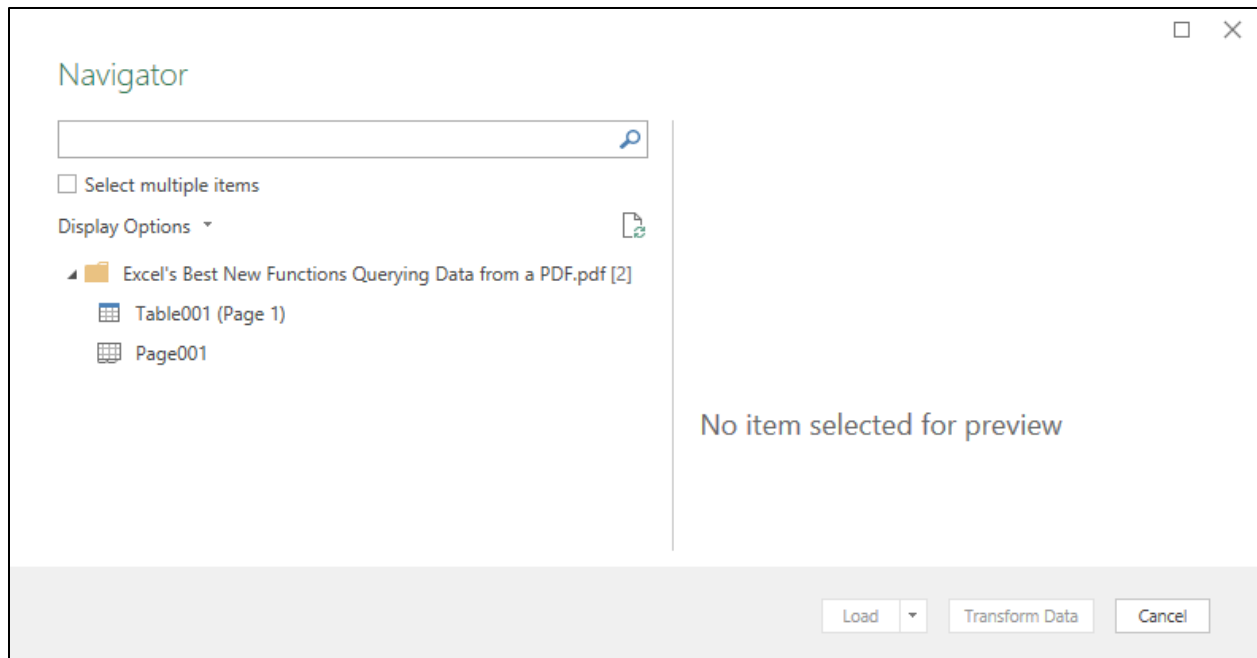


Figure 36 - Opening Power Query's Navigator

Next, choose the table or range you wish to import and select the **Load To...** option, as shown in **Figure 37**.

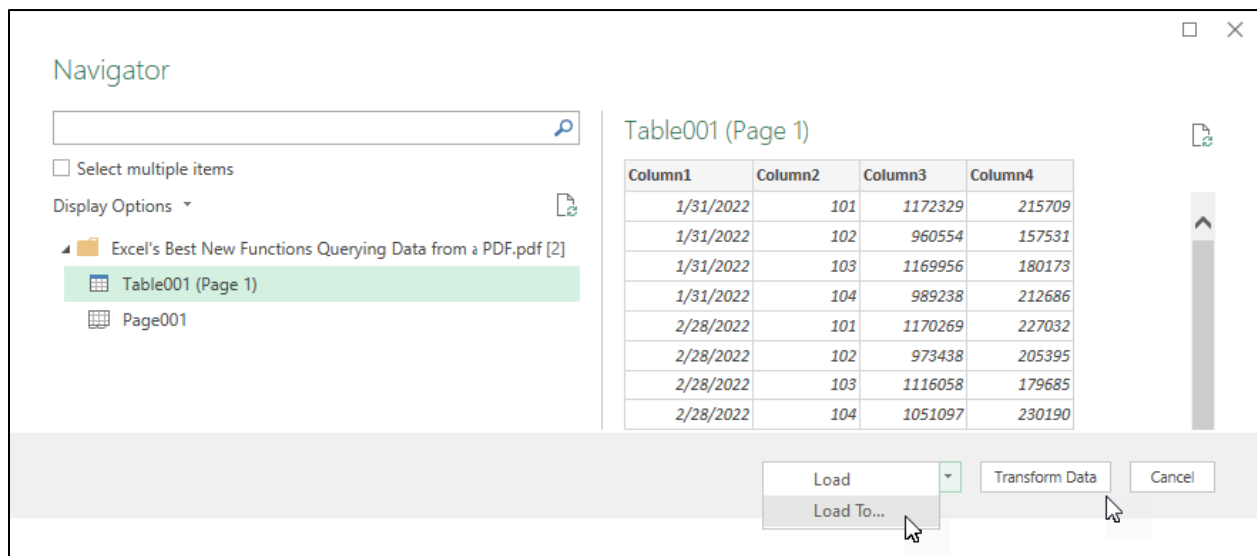


Figure 37 - Previewing the Data to Load into Excel

In the **Import Data** dialog box pictured in **Figure 38**, choose to import the data into a **Table** and place the table in a **New worksheet**.

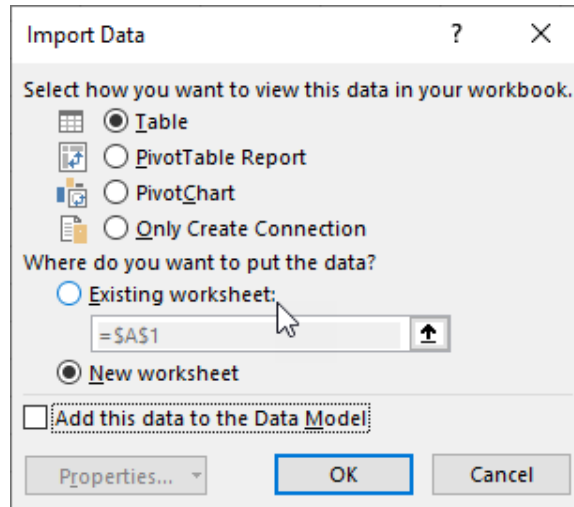


Figure 38 - Excel's Import Data Dialog Box

Figure 39 presents the results of the query. Power Query's ability to extract and import data into Excel from PDF documents should benefit almost all business professionals by reducing time spent on unnecessary data entry.

	A	B	C	D
1	Column1	Column2	Column3	Column4
2	1/31/2022	101	1172329	215709
3	1/31/2022	102	960554	157531
4	1/31/2022	103	1169956	180173
5	1/31/2022	104	989238	212686
6	2/28/2022	101	1170269	227032
7	2/28/2022	102	973438	205395
8	2/28/2022	103	1116058	179685
9	2/28/2022	104	1051097	230190
10	3/31/2022	101	1191158	200115

Figure 39 - Results of Querying a PDF's Data into Excel Using Power Query

Importing Data From A Picture

Continuing with the previous example, suppose someone handed us a sheet of paper with the same numbers as in the last illustration. Further, assume we were away from the office and had no scanner available to use. In this case, we could use a smartphone to snap a picture of the numerical data on the paper and use Power Query's capability to extract data from the image.

Upon taking a picture of your data and saving it to your computer, open the file that contains the image. Then, copy the data you want to import into Excel to the Windows Clipboard. Next, click **From Picture** followed by **From Clipboard** on the Ribbon's **Data** tab. This action will take you to the preview feature shown in **Figure 40**. Verify that the data is correct and, assuming so, click **Insert Data**.

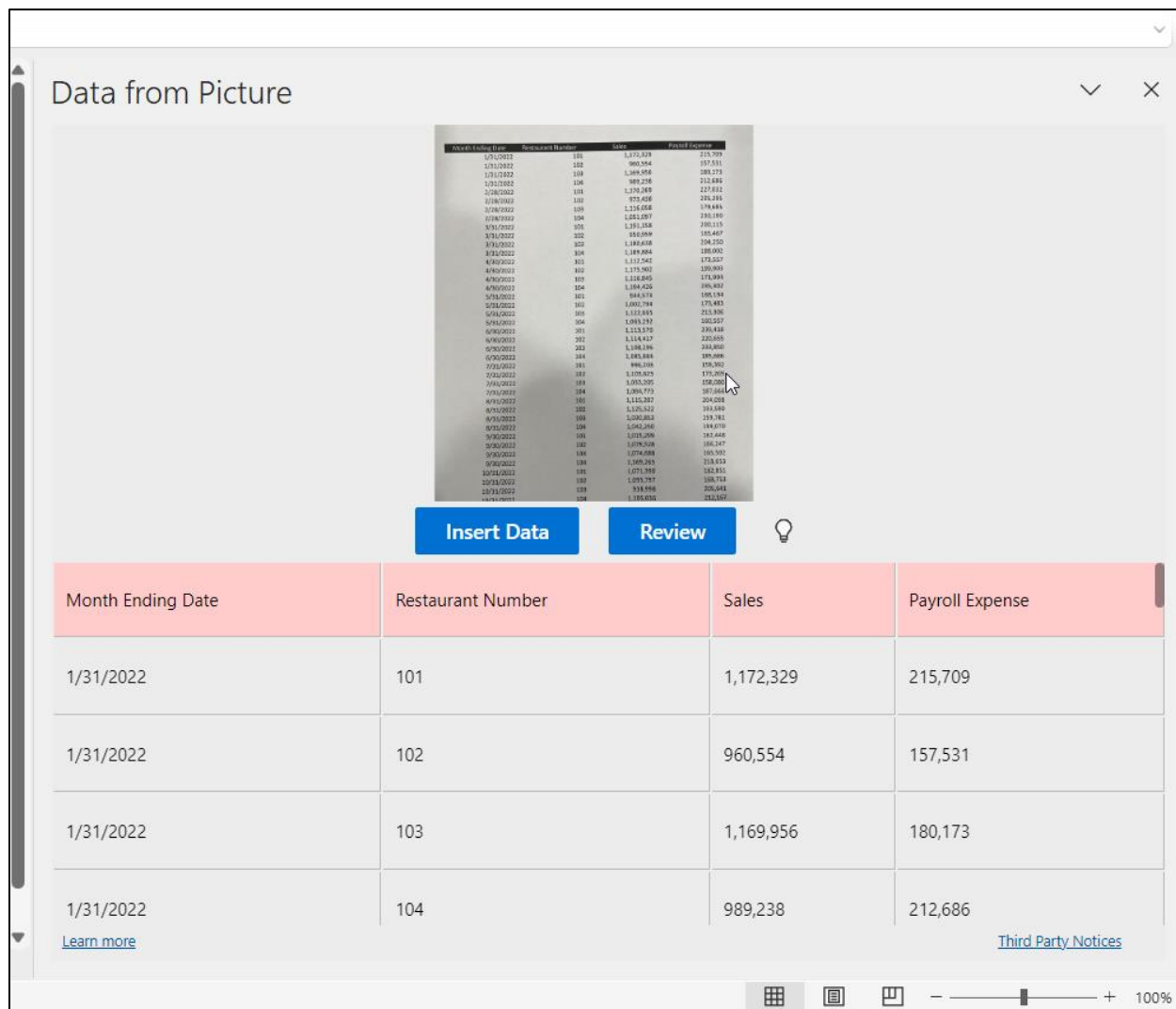


Figure 40 - Importing Data From A Picture

Upon clicking Insert Data, Excel inserts the data into the worksheet, eliminating the tedious and time-consuming task of entering the data manually. Of course, once the data resides in Excel, you can use it in any manner you see fit, including using it in formulas, PivotTables, charts, etc.

Using the Accounting Format

When asked whether they use the accounting format, many accounting professionals will respond that they do not use it. However, accounting is probably the profession's most

frequently used number format. The accounting format applies when selecting the currency or comma styles from the Ribbon's **Home** tab. You can also choose the accounting format from the mini-toolbar. **Figure 41** illustrates both processes.

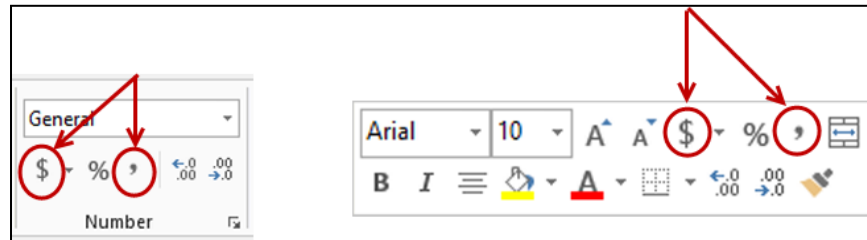


Figure 41 - Currency and Comma Style Buttons on the Ribbon and Mini-Toolbar

The currency style applies the accounting format with the dollar sign and two decimal places, while the comma style applies the accounting format without the dollar sign. In addition, the accounting format displays negative numbers within parentheses and ensures that decimal points align— perfect for financial reports!

Single and Double Underlines

The accounting format affects how single and double underlines apply to numbers and labels. When using the accounting format, single and double underlines for subtotals, totals, and column headings do not span the entire column like cell borders. Instead, the accounting format indents single and double underlines one-character space from the left and right cell margins. These spaces provide the desired break between underlines across adjacent columns without resorting to the frequent practice of inserting narrow *faux* columns to accomplish the same effect.

	A	B	C	D	E
1	DNM Marketing Inc				
2	Income Statement vs. Budget				
3	For the Year Ended December 31, 2021				
4					
5		In Millions of Dollars			
			Adjusted		Variance
6	Region	Budgeted	Budgeted	Actual	from Adjusted Budget
7					
8	Southern	\$ 2.80	\$ 2.70	\$ 2.72	\$ 0.02
9	Eastern	4.60	4.70	4.69	(0.01)
10	Midwestern	3.90	3.50	3.30	(0.20)
11	Mountain	1.70	1.70	1.65	(0.05)
12	Far West	4.70	4.80	4.82	0.02
13					
14	Total Revenues	\$ 17.70	\$ 17.40	\$ 17.18	\$ (0.22)

Figure 42 - Report Showing the Effect of the Accounting Format on Single and Double Underlines

Figure 42 displays single and double underlines applied to labels and numbers in cells formatted with the accounting format. Notice how the single and double underlines do not span the cells' full width, allowing for breaks in the underlines between adjacent columns.

As shown in **Figure 43**, underlines and double underlines are available from the Ribbon's **Home** tab. You may also apply underlines from the **Mini-Toolbar** by pressing **CTRL + U** or pressing **ALT, H, 3, U**. Further, you may create Double underlines from the keyboard by pressing **ALT, H, 3, D**.

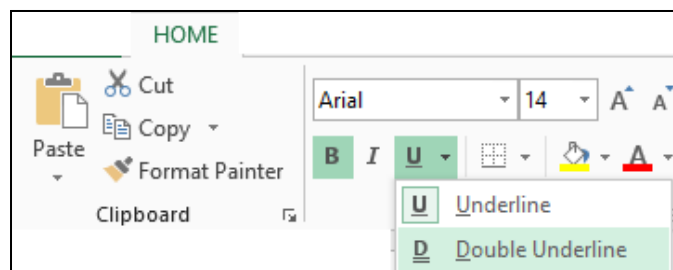


Figure 43 - Applying Underlines and Double Underlines from the Ribbon

Excel Table Fundamentals

Added to the 2007 release of Excel, tables remain one of the application's least utilized features. This situation is unfortunate because tables offer tremendous opportunities for improved accuracy and productivity in all facets of working with Excel, including reporting.

Following is a simple working definition of an Excel table.

Tables are dynamically re-sizing ranges of data in Excel that facilitate end-user interaction at multiple levels. For example, you can interact with data in a table at the cell level, the row level, the column level, the header/footer level, and the table level. Because of the dynamically resizing nature of an Excel table and the ability to interact with a table at multiple levels, tables are the preferred option for storing data in Excel workbooks.

Four Advantages of Working with Tables

While numerous benefits are associated with working with tables, the most significant are the following four.

1. **Tables serve as automatically resizing ranges of data.**
A table's ability to serve as an automatically resizing range of data may be its most important feature. Once you create a table or link data into a table, the table will dynamically and automatically adjust its dimensions to match the volume of data present. By doing so, any formula that references the table will also automatically include any newly added data.
2. **Formulas that you enter in tables copy automatically down the entire column.** When you enter a formula in a table, that formula automatically copies to extend to all rows in the table. Further, if you add additional data to the table, the formulas will continue to self-replicate to all newly added rows.
3. **Formulas that reference information in a table are known as structured reference formulas.** Formulas that reference data in a table refer to the table's structure, not traditional row-and-column addresses. Thus, as the dimensions of the table change, the integrity of the formulas remains intact. You will notice a resemblance between structured reference formulas and formulas that utilize Data Analysis Expressions.
4. **You can relate multiple tables together to create data models.** For example, if you have two or more tables in a workbook, you can quickly and easily relate (*link*) the tables to create a data model. Once you establish the data model, you can build reports that summarize all the data in the model without using VLOOKUP, XLOOKUP, or similar functions to manipulate the data. As a result, the data model approach is more efficient, faster, and less prone to error.

Creating Tables

Knowing the benefits of working with tables, let us focus on creating tables to experience them. Tables are easy to make using any one of three methods:

1. Creating tables from the Ribbon,
2. Creating tables with a keyboard shortcut, and
3. Creating tables by linking data from external data sources.

Additionally, although beyond the scope of this course, you can create tables with macros.

Data Considerations Before Creating a Table

Before you create a table using either of the first two techniques – creating a table from the Ribbon or using a keyboard shortcut –consider the arrangement of the data for the table and whether you should make any modifications. Generally, you should arrange the initial data for the table in contiguous rows and columns. Remember, you can add rows and columns later, so there is no reason to include placeholders for future data when creating the table.

Additionally, while not necessary, you should have column headers at the top of each column that appropriately describe the data in that column; if your data does not have column headers, Excel will insert headers such as *Column1* and *Column2* into your table. All the column headers in a table must be unique. Excel will append a number to the column name to differentiate the labels if there are duplicate names. For example, if there are three columns labeled *Date*, converting the data into a table will cause Excel to rename the second occurrence *Date2* and the third occurrence *Date3*. Of course, you can change a column header anytime by typing over it to replace the existing column header with the desired one.

Creating Tables from the Ribbon

With your data appropriately organized, you are ready to convert it into a table, and you can do so by issuing a command from the Ribbon. First, however, click inside the range of contiguous rows and columns. Then, as shown in **Figure 44**, click **Format as Table** on the Ribbon's **Home** tab and choose from one of the predefined Table Styles that appear in the resulting window. Doing so will transform your data into a table formatted to your desired style. As you drag your cursor over the various style images, Excel's live preview will temporarily apply the formatting to your table. Clicking on a format applies the design automatically. Should you later decide to change the style, click any cell in the table, choose the **Format as Table** icon again from the **Ribbon**, and select a different table style.

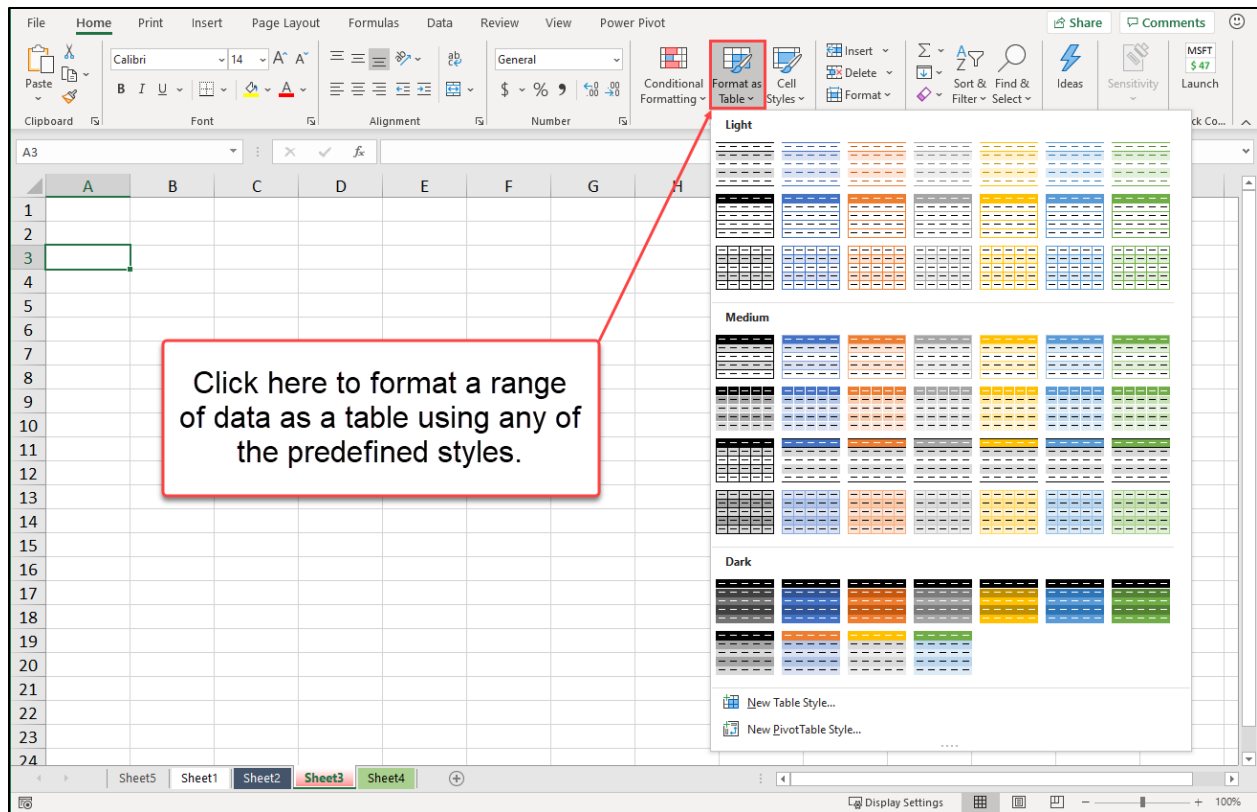


Figure 44 - Creating a Table from the Ribbon

Creating Tables with a Keyboard Shortcut

If you prefer to create your table using a keyboard shortcut, click anywhere inside the data range and press **CTRL + T** or **CTRL + L** to open the **Create Table** dialog box. Confirm the scope of the data and whether your data has headers. Excel then transforms your data into a table using your default table style. Like creating a table from the Ribbon, if you wish to reformat the table, you can choose **Format as Table** from the Ribbon and choose a different table style.

Creating Tables from External Data Sources

For those who use Excel to query external data sources, you can use Power Query to link external data into an Excel table. First, click **Get Data** from the Ribbon's **Data** tab, choose the data source, and click **Import**. Then, click the drop-down arrow next to Load in the resulting Navigator window and select **Load To**. To complete the process, choose the **Table** option in the resulting **Import Data** dialog box and click **OK**. **Figure 45** illustrates this procedure.

Several vital considerations warrant expansion with the data queried into an Excel table.

- From a practical perspective, the data links from the data source into Excel. Thus, as you change the source database's data, refreshing the query will update the Excel table results. To do so, right-click the table and choose **Refresh**.

- If you add formulas to the table, they will self-replicate. Further, upon refreshing the query and new records appearing in the table, the formulas will self-replicate to the newly added rows.
- If you wish to “freeze” the query results in the table, you can choose **Queries & Connections** on the Ribbon’s **Data** tab to open the **Queries & Connections task pane**. There, you can right-click on the query and click **Delete**. Then select **OK** or **Cancel** in the warning dialog box. This procedure de-couples the workbook query so that the data remains in the workbook, but you can no longer refresh the query.
- Further, you can use the abovementioned process to link data from multiple sources. Importantly, those data sources need not be homogeneous. For example, you might link data from an Excel workbook, a text file, and an Access database into three separate tables in Excel. Upon doing so, using techniques described later, you could relate those three tables together to create a data model and then build your report based on the collective data in the data model.

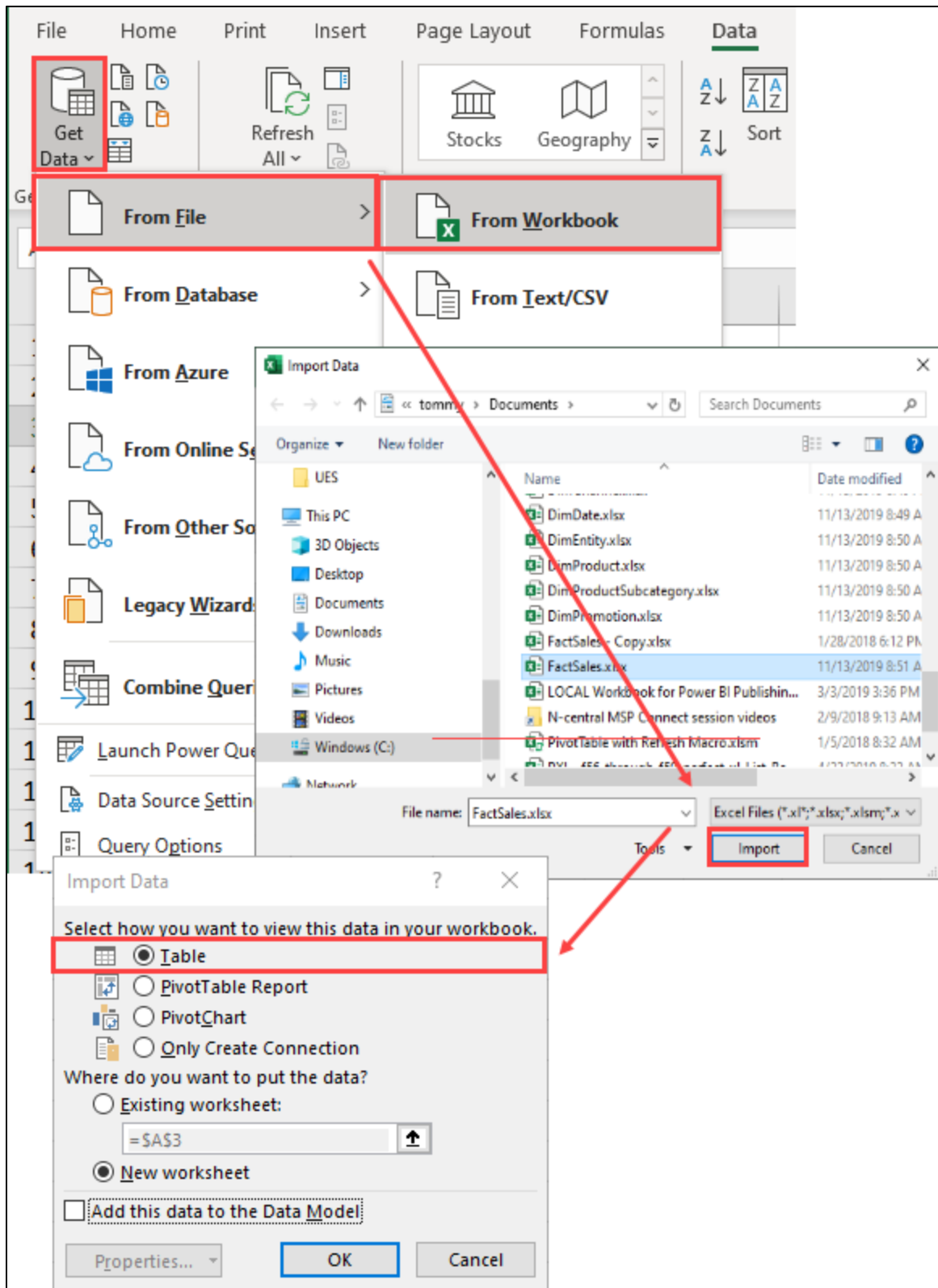


Figure 45 - Selecting a Data Source for a Data Query

It would be difficult, if not impossible, to overstate the importance of linking data from external data sources into a table. With this capability, you are relieved of the tedious and time-consuming tasks of copying and pasting data from external data sources into Excel. In addition, this feature means that your tables – and, by extension, all reports you build based on them – can summarize real-time data. For many, this is a “game-changer” for Excel-based financial and operational reporting!

Naming and Renaming Tables

Regardless of how you create a table, Excel automatically assigns a name to the table. The first table you add to a workbook inherits the name *Table1*, the second table you create in a workbook inherits the name *Table2*, and so on. Many users find these default names to be non-descriptive of the data in the table. Further, these non-descript names can be confusing if you use numerous tables in a workbook. Fortunately, you can easily rename a table and assign it a name that makes it easier for all workbook users to understand the nature of the table's data.

To rename a table, begin by clicking anywhere in the table. Then, from the Ribbon's **Table Tools Design** tab, click the **Table Name** field in the **Properties Group** and type the new name for your table, as shown in **Figure 46**.

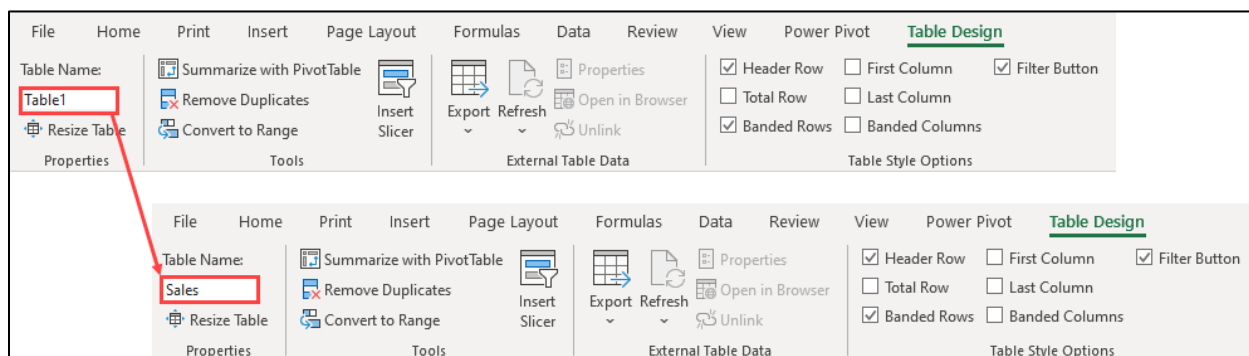


Figure 46 - Renaming a Table

While you have tremendous freedom when creating a new table name, you must follow some guidelines. The following list should look familiar to those familiar with the rules associated with adding names to other objects in Excel.

1. The first character of the table name must be a letter, an underscore, or a backslash. You can use numbers and special characters in a table name, but they cannot be the name's first character. Additionally, spaces are not allowed in table names.
2. The length of the table name must be 255 characters or fewer.
3. The table name cannot be a combination of letters and numbers that could otherwise be a row and column address. For example, *A100* cannot be a table name because it is a cell address in a worksheet.

4. All table names must be unique within a workbook.
5. Table names are not case-sensitive. Excel recognizes *Revenue*, *revenues*, and *REVENUE* as the same table name.

Summary

Excel is anything but a new application. However, that doesn't mean that it is a stale tool. On the contrary, new features continue to emerge, and existing features continue to improve. Those who learn about and utilize key features in Excel will find their efficiency and accuracy soar. In this session, you have learned about "must-know" Excel features that are broadly applicable to accounting and financial professionals. Commit to investing a few minutes learning how to use these tools to be as efficient and accurate when working with Excel as possible.