

How to Split Text in Excel and Extract the First Value

Authored by
stats writer

February 12, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Split Text in Excel and Extract the First Value*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=130240>

An Introduction to Modern Text Manipulation in Excel

In the contemporary landscape of **data analysis**, the ability to efficiently reorganize and clean information is a fundamental skill. Within the environment of **Microsoft Excel**, users frequently encounter scenarios where a single cell contains a composite string of data, such as a full name, an address, or a product code, that necessitates separation into distinct components. This process, commonly referred to as **text splitting**, allows for more granular analysis and better database management. Traditionally, users relied on static tools to achieve this, but the introduction of dynamic **spreadsheet** functions has revolutionized the workflow, allowing for real-time updates and more sophisticated data handling.

Mastering the art of splitting text is not merely about aesthetic organization; it is about transforming raw data into actionable insights. When data is properly parsed, it becomes much easier to perform **sorting**, filtering, and complex calculations. For instance, extracting a first name from a list of full names can facilitate personalized communication in marketing or simplify the creation of unique identifiers in a human resources database. By leveraging the power of modern functions, **Excel** users can automate these tasks, significantly reducing the margin for human error and increasing overall productivity in professional settings.

Historically, the primary method for this task was the **Text to Columns** wizard, a robust but manual feature that requires several steps to execute. While effective, it lacks the flexibility of formula-based solutions, which automatically recalculate when the source data changes. In the following sections, we will explore the synergy between the **TEXTSPLIT** and **CHOOSECOLS** functions. This combination provides a dynamic and elegant way to retrieve specific items from a string, such as the very first element, ensuring that your **data management** remains both precise and scalable.

The Evolution of Excel Functions: From Static to Dynamic

The journey of **data cleansing** in **Excel** has seen significant milestones, moving from manual string manipulation to the implementation of **dynamic array formulas**. In older versions of the software, users often had to combine the **LEFT**, **FIND**, and **LEN** functions in complex, nested formulas to extract a first name from a string. These formulas were frequently difficult to debug and prone to breaking if the structure of the text changed even slightly. The introduction of **dynamic arrays** has simplified this logic, allowing a single formula to return multiple values across several cells automatically.

Among these innovations, the **TEXTSPLIT** function stands out as a versatile tool designed to replace the cumbersome **regular expression**-like logic previously required for parsing strings. By allowing users to define a **delimiter**, such as a space, comma, or semicolon, **Excel** can now instantly recognize the boundaries between data points and distribute them accordingly. This level

of automation is essential for professionals who handle large volumes of data and require consistency across their **workbooks** without repetitive manual intervention.

Furthermore, the integration of the **CHOOSECOLS** function adds a layer of precision to the splitting process. While **TEXTSPLIT** handles the separation of data, **CHOOSECOLS** allows the user to target and extract specific columns from the resulting **array**. This is particularly useful when only a specific piece of information, like a first name or a specific category code, is required for a report. This modular approach to formula construction reflects the modern philosophy of **Excel** development: providing powerful, specialized tools that can be combined to solve complex problems with minimal effort.

Deep Dive into the TEXTSPLIT Function

The **TEXTSPLIT function** is the engine behind modern string parsing in **Excel**. Its primary purpose is to take a single string of text and break it into multiple parts based on a specified character or set of characters. The basic syntax of the function requires the text to be split and the **delimiter** that indicates where the split should occur. This function is part of the newer suite of **Excel** features, meaning it is optimized for performance and works seamlessly with other modern functions to create sophisticated data pipelines.

One of the most powerful aspects of **TEXTSPLIT** is its ability to handle multiple **delimiters** simultaneously. For example, if a dataset contains names separated by both spaces and commas, **TEXTSPLIT** can be configured to recognize both, ensuring that the output remains clean and consistent. This flexibility is a significant upgrade over the older **INDEX function** and **MATCH** combinations which often required much more verbose logic to achieve similar results. By understanding the parameters of **TEXTSPLIT**, users can tailor their data extraction to the specific nuances of their unique datasets.

In the context of our goal--retrieving the first item--**TEXTSPLIT** performs the initial work of identifying every individual element within the source cell. If a cell contains "Andy Bernard," the function identifies "Andy" as the first part and "Bernard" as the second part based on the space **delimiter**. While the function by itself would normally "spill" both names into adjacent cells, we can wrap it in other functions to refine the output. This ability to generate an intermediate **array** of values is what makes **TEXTSPLIT** an indispensable component of the modern **Excel** toolkit for **information technology** professionals and data analysts alike.

Excel: Split Text and Get First Item

The **TEXTSPLIT** function in **Excel** can be used to split text based on a specific **delimiter**.

To get the first item that results from this function, you can use the following syntax:

=CHOOSECOLS(TEXTSPLIT(A2, " "), 1)

This particular example will split the text in cell **A2** using a space as a **delimiter** and then it will return only the first item that results from the split.

For example, if the string in cell **A2** is **Andy Bernard** then this formula will return **Andy**.

The following example shows how to use this formula in practice.

Example: How to Split Text and Get First Item in Excel

Suppose we have the following column of names in **Excel**:

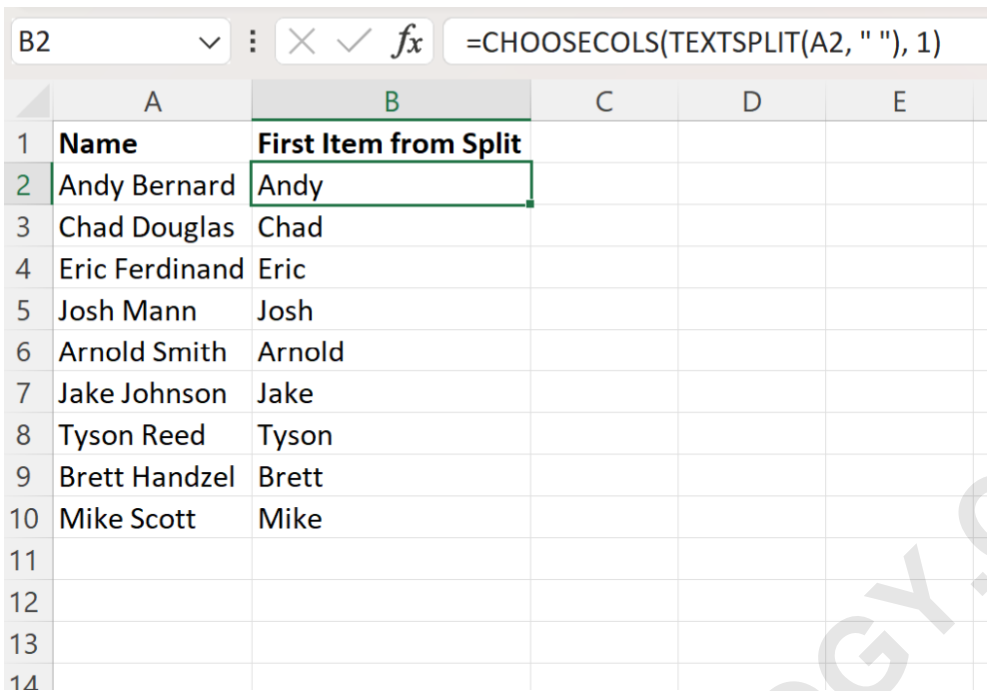
	A	B	C	D	E
1	Name				
2	Andy Bernard				
3	Chad Douglas				
4	Eric Ferdinand				
5	Josh Mann				
6	Arnold Smith				
7	Jake Johnson				
8	Tyson Reed				
9	Brett Handzel				
10	Mike Scott				
11					
12					
13					
14					

Suppose we would like to split the names based on where a space occurs and then get only the first item that results from the split.

We can type the following formula into cell **B2** to do so:

=CHOOSECOLS(TEXTSPLIT(A2, " "), 1)

We can then click and drag this formula down to each remaining cell in column B:



	A	B	C	D	E
1	Name	First Item from Split			
2	Andy Bernard	Andy			
3	Chad Douglas	Chad			
4	Eric Ferdinand	Eric			
5	Josh Mann	Josh			
6	Arnold Smith	Arnold			
7	Jake Johnson	Jake			
8	Tyson Reed	Tyson			
9	Brett Handzel	Brett			
10	Mike Scott	Mike			
11					
12					
13					
14					

The formula splits the names in column A based on where the space occurs and then only returns the first item from the split.

For this particular example, the first name of each person in column A is returned.

Note that if you'd like to return the second item instead, you can replace **1** with **2** in the formula:

=CHOOSECOLS(TEXTSPLIT(A2, " "), 2)

The following screenshot shows how to use this formula in practice:

	A	B	C	D	E
1	Name	Second Item from Split			
2	Andy Bernard	Bernard			
3	Chad Douglas	Douglas			
4	Eric Ferdinand	Ferdinand			
5	Josh Mann	Mann			
6	Arnold Smith	Smith			
7	Jake Johnson	Johnson			
8	Tyson Reed	Reed			
9	Brett Handzel	Handzel			
10	Mike Scott	Scott			
11					
12					
13					
14					
15					

This time the formula returns the second item from the split.

Isolating Data with the CHOOSECOLS Utility

The **CHOOSECOLS** function is an essential companion to **TEXTSPLIT** when the goal is data isolation rather than data expansion. In a standard operation, **TEXTSPLIT** produces a horizontal **array** that fills multiple cells. However, in many reporting scenarios, you only need one specific piece of that **array**. **CHOOSECOLS** acts as a filter, allowing you to specify exactly which column of the generated **array** should be displayed in the cell, effectively suppressing the rest of the split data.

This function works by taking an **array** as its first argument and one or more integers as subsequent arguments. Each integer corresponds to the index of the column you wish to retrieve. By passing the entire **TEXTSPLIT** result into **CHOOSECOLS** and specifying the number 1, you are instructing **Excel** to only show the very first part of the split text. This method is far more intuitive than the older **selection** techniques and allows for much cleaner **spreadsheet** design, as it prevents unwanted data from spilling into neighboring cells.

Furthermore, **CHOOSECOLS** is highly adaptable. If your data structure changes--for example, if you decide you need the last name instead of the first name--you simply change the index number. This makes your **Excel** models robust and easy to update. The function is also capable of returning multiple columns in a custom order, though for the purpose of extracting the first item, its

simplest application is the most common. Using this function ensures that your **data processing** is both targeted and efficient, fulfilling the specific requirements of your analysis without unnecessary clutter.

Practical Walkthrough: Extracting First Names

To implement this formula in a real-world **Excel** environment, start by identifying the source data you wish to parse. As shown in our examples, a list of full names in column A serves as a perfect candidate. By entering the formula into cell B2, you initiate a sequence where **Excel** first looks at the text, identifies the space character, and mentally breaks the string into parts. The **CHOOSECOLS** wrapper then immediately grabs the first part and displays it, while the rest of the string is disregarded by the cell.

The beauty of this method lies in its **scalability**. Once the formula is correctly entered in the first row, you can use the fill handle to drag it down through hundreds or thousands of rows. **Excel** will automatically adjust the cell references and perform the split and extraction for every entry. This is significantly faster than using manual **copy-paste** operations or the **Text to Columns** feature, which would need to be re-run every time new names are added to the list. This dynamic nature is a hallmark of professional **business intelligence** practices.

It is also important to note the role of the **delimiter** in this process. While our example uses a space (" "), you can easily modify this to any character that separates your data. For instance, if you are working with **comma-separated values (CSV)**, you would replace the space with a comma (","). The logic remains identical, demonstrating the versatility of the **TEXTSPLIT** and **CHOOSECOLS** combination. This consistency across different types of data sources is why these functions are highly recommended for any **Excel** user looking to upgrade their skillset.

How This Formula Works

Recall the formula that we used to split the text and get the first item from the resulting split:

```
=CHOOSECOLS(TEXTSPLIT(A2, " "), 1)
```

Here is how this formula works:

First, the **TEXTSPLIT** function splits the text in cell **A2** based on where the space occurs. For example, the name **Andy Bernard** is split into one column that contains **Andy** and a second column that contains **Bernard**. This creates an internal **array** that **Excel** keeps in its memory during the calculation process. Understanding this internal step is key to troubleshooting more complex formulas involving **string manipulation**.

Next, the **CHOOSECOLS** function selects only the first column from the output. By specifying the number 1, you are telling **Excel** to ignore all other columns generated by the split and only return the item at the first index. The end result is that we're able to get only the first item from the split, which is **Andy**. This combination of "split then select" is a powerful pattern that can be applied to many different data extraction tasks.

By using this nested approach, you maintain a single, clean formula that is easy to read and manage. It avoids the need for **helper columns**, which can often clutter a **spreadsheet** and make it more difficult for others to understand. This streamlined workflow is preferred in professional environments where **document automation** and clarity are prioritized. Mastering these functions will undoubtedly make your data analysis tasks more efficient and your **Excel** workbooks more professional.

Advanced Scenarios and Delimiter Management

While splitting by a single space is common, real-world data is often much messier. You might encounter strings with multiple different **delimiters**, such as "Andy;Bernard,Sales". In such cases, **TEXTSPLIT** can accept an **array** of **delimiters**, such as {" ; " , " , " }. This allows the function to split the text whenever it encounters either a semicolon or a comma. By combining this with **CHOOSECOLS**, you can still easily extract the first item regardless of how complex the separators are. This adaptability is vital when dealing with data exported from various **database** systems.

Another advanced scenario involves handling varying numbers of items in a string. For example, some rows might have three names while others have only two. The **CHOOSECOLS** and **TEXTSPLIT** combination remains stable in these situations because it only looks for the first item. However, if you were trying to find the last item, you might use the **COLUMNS function** within **CHOOSECOLS** to dynamically identify the final index. This shows the depth of logic available to users who understand these foundational **Excel** tools.

Finally, consider the case of extra whitespace. Often, data entered by humans contains leading or trailing spaces that can interfere with **string matching**. Wrapping your cell reference in the **TRIM** function--like `TEXTSPLIT(TRIM(A2), " ")`--is a best practice. This ensures that any accidental spaces are removed before the split occurs, resulting in much cleaner data. These small refinements are what separate basic **Excel** users from advanced **data analysts** who produce reliable, high-quality work.

Comparative Analysis: Formulas vs. Traditional Tools

It is worth comparing the **TEXTSPLIT/CHOOSECOLS** method with the traditional **Text to Columns** wizard. The wizard is excellent for one-time transformations where the source data will not change. It is intuitive and doesn't require knowledge of **functional programming** or formula

syntax. However, its static nature is a significant drawback in dynamic projects. If the source data is updated, the user must manually re-run the wizard, which is inefficient and creates a risk that the report will show outdated information.

In contrast, the formula-based approach we have discussed is completely dynamic. As soon as you update cell A2, cell B2 updates instantly. This is crucial for **dashboards** and automated reports. Furthermore, formulas can be easily audited and copied across different files. While there is a slight learning curve to understanding the syntax of **dynamic arrays**, the long-term benefits in terms of time saved and accuracy gained are immense. This shift towards formula-based automation is a key trend in **data science** and office productivity.

Lastly, the formula method is more "portable." If you need to explain your process to a colleague, you can simply share the formula. With the **Text to Columns** wizard, you would have to explain a multi-step UI process. Using functions like **CHOOSECOLS** also makes your intentions clear to anyone auditing the **spreadsheet**: it is obvious that you are intentionally selecting the first column of a split. This transparency is a core component of **data integrity** and collaborative success in any organization.

Conclusion and Best Practices

In summary, splitting text and retrieving the first item in **Excel** has never been easier thanks to the introduction of **TEXTSPLIT** and **CHOOSECOLS**. These functions provide a powerful, flexible, and dynamic alternative to older, manual methods. By understanding the logic of **delimiters** and **array** indexing, you can handle a wide variety of **data cleaning** tasks with ease. Whether you are extracting names, codes, or categories, this formulaic approach ensures that your work is accurate and updates automatically as your data changes.

To get the most out of these tools, always remember to verify your **delimiters** and consider using **TRIM** to handle unexpected spaces. As you become more comfortable with these functions, you will find that they serve as building blocks for even more complex **Excel** solutions. The ability to manipulate strings efficiently is a hallmark of an expert user and is highly valued in any data-driven role. By incorporating these modern functions into your daily routine, you are future-proofing your **workflow** and enhancing your analytical capabilities.

The following tutorials explain how to perform other common operations in **Excel**: