

How do I extract text between second and third space in Excel?

Authored by
stats writer

November 18, 2025

RECOMMENDED CITATION

stats writer (2025). *How do I extract text between second and third space in Excel?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=95723>

Working with text data in Excel often requires advanced manipulation techniques, especially when dealing with data that is separated by delimiters like spaces, commas, or semicolons. A common challenge faced by data analysts is isolating specific substrings--for example, extracting only the third word from a lengthy sentence or a file path. This task, historically complex, has been simplified significantly with the introduction of newer text manipulation functions in modern versions of Excel.

The core problem we address here is pinpointing and extracting the exact text segment situated between the second and third occurrence of a space character. Achieving this requires a precise method of identifying the starting point (after the second space) and the ending point (before the third space). While older formulas relied heavily on nested combinations of **MID**, **FIND**, and **SUBSTITUTE**, modern Excel versions offer a vastly cleaner and more intuitive solution using the TEXTBEFORE and TEXTAFTER functions.

Leveraging Modern Excel Functions for Text Extraction

For users running recent versions of Microsoft 365 or Excel 2021, the task of extracting delimited text has been streamlined through specialized functions designed for this exact purpose. To efficiently isolate the text located specifically between the second and third spaces within a cell, we utilize a powerful nested structure combining the **TEXTBEFORE** and **TEXTAFTER** functions. This approach is superior to legacy methods because it handles the index counting of delimiters inherently, removing the need for complex calculations involving character positions.

The strategy involves two distinct stages executed in a nested manner. First, we must identify the starting boundary of our target text, which means discarding everything up to and including the second space. This is the role of the **TEXTAFTER** function. Once the starting boundary is established, the resultant string contains our desired segment followed by the remainder of the cell's content. The second stage uses the **TEXTBEFORE** function to discard everything following the desired segment, using the next space (which is now the first space in the truncated string) as the stopping delimiter. This elegant combination ensures precise extraction of the target word or phrase.

The standard syntax employed for this operation is concise and highly readable, significantly enhancing the maintainability of your spreadsheets. If, for instance, your data resides in cell **A2**, the formula structure looks like this. The original content mandated preserving the following code block exactly:

```
=TEXTBEFORE(TEXTAFTER(A2, " ", 2), " ")
```

This specific example is designed to extract the textual content found precisely between the

second space and the third space marker in cell **A2**, offering a straightforward solution to a common data processing requirement.

Deconstructing the Core Formula Syntax

To fully appreciate the power of this solution, it is essential to analyze how the nested components interact. The formula `=TEXTBEFORE(TEXTAFTER(A2, " ", 2), " ")` operates from the inside out, following Excel's standard rule for nested functions. Understanding the parameters of both the inner and outer functions is key to adapting this formula for different delimiting requirements, such as extracting the text between the fifth and sixth commas.

The inner function, `TEXTAFTER(A2, " ", 2)`, is executed first. It takes three arguments: the text source (**A2**), the delimiter (" "), and the instance number (**2**). By setting the instance number to **2**, the `TEXTAFTER` function instructs Excel to return everything that appears after the second occurrence of the space character. If the content of A2 is "Hello how are you today," this inner function returns the substring "are you today."

The result of the inner function--the truncated string--then becomes the primary text input for the outer `TEXTBEFORE` function. The outer function is structured as `TEXTBEFORE(Result of TEXTAFTER, " ")`. This function asks Excel to return all text that occurs before the first instance of the specified delimiter (" ") within the newly processed string. Since the target word (e.g., "are") is now the first segment in the truncated string, applying `TEXTBEFORE` with a delimiter instance of 1 (the default when omitted) successfully isolates that word, achieving our goal of extracting the text between the original second and third spaces.

Practical Demonstration: Isolating Text Segments

We will now walk through a concrete example demonstrating the application of this formula to a list of sample data. Suppose we are dealing with a column of data that contains structured phrases or sentences where the information needed is always positioned as the third word. This kind of requirement is common in parsing logs, addresses, or specific product codes that use spaces as inherent separators.

Consider the following list of sample strings provided in column A of our worksheet. Our objective is to populate column B with the text segment that exists between the second and third space for each corresponding row. This scenario confirms the utility of the combined `TEXTBEFORE` and `TEXTAFTER` approach for reliable data segmentation.

The initial dataset arrangement is visually represented below, showing various strings that contain multiple spaces:

	A	B	C	D
1	String			
2	Hey how are you today			
3	This is great weather			
4	We should have fun			
5	This is a good team			
6	They play really well I think			
7	The dog ran so fast			
8	The cat walked quite slow			
9	This is one horse			
10				
11				
12				
13				
14				
15				
16				
17				

To begin the extraction process, we input the established formula into cell **B2**, targeting the content of **A2**. The formula remains consistent with our prior discussion, utilizing the delimiter count to precisely locate the third segment of text within the entire string:

=TEXTBEFORE(TEXTAFTER(A2, " ", 2), " ")

Once the formula is entered and confirmed in cell B2, it yields the extracted text for the first row. The critical next step in applying this solution to the entire dataset is to efficiently replicate the formula down the column. We achieve this by clicking and dragging the fill handle located in the bottom-right corner of cell B2 down to encompass all remaining cells corresponding to the data in Column A. This action automatically adjusts the cell references (e.g., from A2 to A3, A4, etc.) due to relative referencing.

The result of applying this drag-and-fill operation is shown below, confirming that Column B successfully contains the target text extracted between the second and third space from every cell in Column A:

	A	B	C
1	String	Text Between 2nd and 3rd Space	
2	Hey how are you today	are	
3	This is great weather	great	
4	We should have fun	have	
5	This is a good team	a	
6	They play really well I think	really	
7	The dog ran so fast	ran	
8	The cat walked quite slow	walked	
9	This is one horse	one	
10			
11			
12			
13			
14			
15			
16			

Column B now provides the clean, segmented text, achieving the desired extraction efficiently across the entire range without needing complex array formulas or manual parsing.

Detailed Step-by-Step Formula Evaluation

To solidify the understanding of this extraction method, let us meticulously trace the execution of the formula using a single example string: "Hello brilliant people of the world." We are aiming to extract the word "people."

The formula applied to this cell (say, A2) is: `=TEXTBEFORE(TEXTAFTER(A2, " ", 2), " ")`.

Inner Function Execution (TEXTAFTER): Excel first evaluates `TEXTAFTER(A2, " ", 2)`. The `TEXTAFTER` function looks for the delimiter (" ") and counts its instances. The first space is after "Hello," and the second space is after "brilliant." Since the instance number is 2, the function returns everything after the second space.

Result of inner function: **"people of the world."**

Outer Function Execution (TEXTBEFORE): This result, "people of the world," is then passed to the outer `TEXTBEFORE` function. The formula now conceptually reads: `=TEXTBEFORE("people of the world", " ")`.

The **TEXTBEFORE** function, with no specified instance number, defaults to searching for the first occurrence of the delimiter (" "). It extracts all text preceding that first space.

Final Output: The text before the first space in the string "people of the world" is "people."

Final Result: "people"

This sequential execution confirms that the nesting structure successfully isolates the text segment bordered by the second and third spaces of the original input string, demonstrating high precision and reliability for data parsing tasks within Excel.

Addressing Edge Cases and Error Handling

While the combination of **TEXTBEFORE** and **TEXTAFTER** is highly efficient, users must consider scenarios where the input data does not conform perfectly to the expected structure. The primary edge case occurs when the source cell contains fewer than three spaces, meaning the third segment of text does not exist or the delimiters required for the extraction boundaries are absent.

If the input cell, say **A2**, contains only "One Two," there are only two spaces. When **TEXTAFTER(A2, " ", 2)** runs, it returns "Two" (the text after the second space). However, when the outer **TEXTBEFORE** tries to execute on "Two," it searches for a space delimiter that is not present. In such cases, the TEXTBEFORE function returns the entire remaining string ("Two"), or it may return an error depending on the exact version and handling of non-existent delimiters, although typically it returns the full text if the delimiter is not found.

If the input contains only "One Two Three," which has two spaces, the result is "Three." If the input is "One Two Three Four," having three spaces, the target "Three" is successfully extracted. To robustly handle situations where the required number of delimiters might be missing, it is advisable to wrap the core formula using the **IFERROR** function. This allows the user to return a blank cell or a descriptive message if the extraction fails, preventing misleading results or formula errors.

A robust formula structure incorporating error handling would look like this: `=IFERROR(TEXTBEFORE(TEXTAFTER(A2, " ", 2), " "), "")`. This ensures that if the extraction fails due to insufficient spaces, the cell simply displays a blank value, maintaining the cleanliness of the data output.

Alternative Method for Legacy Excel Versions

For users operating older versions of Excel that predate the release of **TEXTBEFORE** and **TEXTAFTER** (i.e., versions before Microsoft 365 or Excel 2021), achieving this text extraction requires a significantly more complex formula utilizing **MID**, **FIND**, and sometimes **SUBSTITUTE** to

accurately locate and segment the string based on delimiter position.

The legacy approach relies on calculating the exact start position and the exact length of the desired substring. The formula needs to determine the position of the second space to find the start point, and the position of the third space to calculate the length. The generalized legacy formula structure is: `=MID(Text, Start_Num, Num_Chars)`.

To calculate the **Start_Num** (position after the second space), we use: `FIND(" ", A2, FIND(" ", A2) + 1) + 1`. This complex nesting finds the first space, then finds the second space relative to the first, and adds 1 to start reading immediately after the second space.

To calculate the **Num_Chars** (length between the second and third spaces), we subtract the position of the second space from the position of the third space, and then subtract 1. Finding the third space requires another level of nesting or the use of **SUBSTITUTE** to isolate the position. This demonstrates why the modern **TEXTBEFORE** and **TEXTAFTER** functions represent such a dramatic improvement in readability and simplicity for text manipulation tasks.

Conclusion and Best Practices

Extracting delimited text segments, such as the text between the second and third space, is a fundamental requirement in data cleansing and preparation within Excel. The introduction of the **TEXTBEFORE** and **TEXTAFTER** functions provides the cleanest, most efficient method for accomplishing this task in modern spreadsheet applications. By nesting **TEXTBEFORE** around **TEXTAFTER**, we first truncate the string to start precisely after the required delimiter count, and then we truncate the resulting string to end exactly before the next required delimiter.

The recommended formula structure, `=TEXTBEFORE(TEXTAFTER(A2, " ", 2), " ")`, is powerful because it externalizes the complex counting logic that previously required cumbersome manual calculation of character indices using **FIND** and **MID**. For best practices, always ensure that your formula includes robust error handling, such as the use of **IFERROR**, especially when dealing with inconsistent data inputs that may not contain the expected minimum number of delimiters.

Mastering these modern text manipulation tools significantly enhances productivity and allows analysts to focus on interpreting results rather than debugging lengthy, nested formulas.