

How do I use TRUE or FALSE in VLOOKUP in Excel?

Authored by
stats writer

November 19, 2025

RECOMMENDED CITATION

stats writer (2025). *How do I use TRUE or FALSE in VLOOKUP in Excel?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=96663>

Understanding the VLOOKUP Function

The VLOOKUP function is one of the most powerful and frequently utilized tools within Excel for data retrieval. Its primary purpose is to search for a specific value in the leftmost column of a designated range (the 'table array') and return a corresponding value from a specified column in the same row. While its functionality is straightforward, mastering its final argument--the `range_lookup` parameter--is essential for accurate data analysis and avoiding common lookup errors. Understanding how this parameter dictates the matching behavior is the key difference between reliable data extraction and frustrating inaccuracies.

Many users encounter issues with VLOOKUP because they overlook or misunderstand the implications of the fourth argument. This final parameter, which accepts either **TRUE** or **FALSE**, fundamentally alters how Excel searches through your data. Whether you need to locate an item perfectly by name or categorize a numeric value based on thresholds determines which setting you should choose. By default, Excel assumes the argument is **TRUE**, which can often lead to unexpected and potentially misleading results if your lookup column is not correctly sorted, especially when dealing with non-numeric data.

This detailed guide will clarify the crucial distinction between setting the `range_lookup` argument to **TRUE** (for an approximate match) versus **FALSE** (for an exact match). We will delve into the syntax, provide comprehensive examples, and explain why, in the vast majority of practical business and analytical scenarios, using **FALSE** is the safer and more recommended practice for ensuring data integrity and precision in your spreadsheet operations.

Dissecting the VLOOKUP Formula Arguments

The power of VLOOKUP is encapsulated in its four distinct components. To leverage this function effectively, it is imperative to grasp what each argument requires and how they interact to achieve the desired lookup result. The function's syntax is structured predictably, moving from the value you seek to the specific mechanism used to find it.

The standard syntax for the function is:

VLOOKUP(lookup_value, table_array, col_index_num,)

Here is a detailed breakdown of each argument:

lookup_value: This is the specific item, text string, or numerical value you are trying to find within the first column of your data range. It serves as the search key for the entire operation.

table_array: This defines the entire range of cells that contains both the value you are looking up (which must be in the first column) and the value you wish to retrieve. It is crucial to fix this range using absolute references (e.g., `\$A\$2:\$D\$10`) if you plan to copy the formula down a column.

col_index_num: This mandatory integer specifies the column number within the **table_array** from which the resulting data should be returned. If your **table_array** spans columns A through D, and you want to return a value from column C, your index number would be 3.

range_lookup: This is the optional (but crucial) logical value that determines the search mode. Inputting **TRUE** enables the approximate match mode, while **FALSE** enforces the exact match mode. If omitted, Excel defaults to **TRUE**, often a source of error for novice users.

TRUE: The Approximate Match Setting

When the final argument, `range_lookup`, is set to **TRUE**, or if it is completely omitted (since **TRUE** is the default), VLOOKUP operates in approximate match mode. In this mode, Excel searches for the closest possible value that is less than or equal to the **lookup_value**. It does not require an identical match to return a result. This behavior is primarily useful for working with numerical data organized into tiers or ranges, such as calculating tax rates based on income brackets or assigning grades based on numerical scores.

However, for the approximate mode to function correctly and predictably, a strict requirement must be met: the first column of the **table_array** must be sorted in **ascending order**. If the lookup column is unsorted when using **TRUE**, the results returned by Excel are virtually guaranteed to be incorrect, leading to severe data integrity issues that can be hard to spot immediately. Excel will return the first value it finds that is larger than the lookup value, and then step back to the previous row, assuming the list is sorted. If it hits an unsorted section, the logic breaks entirely.

While **TRUE** provides flexibility for handling interpolated data or ranges, its implementation complexity and the strict sorting requirement mean it is less commonly used than the precise nature of **FALSE**. When using non-numerical data (like names or product codes), **TRUE** should almost always be avoided unless you are absolutely certain that an approximate match is acceptable for your analysis, which is rarely the case for identification lookups. It is crucial to remember that if the VLOOKUP function cannot find a value less than or equal to the **lookup_value** (meaning the **lookup_value** is smaller than the smallest value in the array), it will return the #N/A error, indicating that a match could not be approximated.

FALSE: Ensuring Exact Data Retrieval

Setting the `range_lookup` argument to **FALSE** instructs VLOOKUP to operate in exact match mode. This is the preferred setting for most lookup tasks involving unique identifiers, product names, customer IDs, or any instance where the returned value must correspond precisely to the searched value. When using **FALSE**, Excel searches the first column of the **table_array** exclusively for a value that is identical to the **lookup_value**. If such a match is not found, the function returns the #N/A error.

The primary advantage of using **FALSE** is reliability. It removes the ambiguity associated with approximate searches and eliminates the strict requirement for the lookup column to be sorted. This flexibility is invaluable when working with large or frequently updated datasets where maintaining perfect alphabetical or numerical order might be impractical or inefficient. By using the exact match setting, users can be confident that the data returned accurately reflects the relationship between the lookup value and the corresponding data point.

For all lookups involving text strings (names, categories, codes) or when absolute certainty is required that the lookup value exists in the source data, **FALSE** is the mandatory choice. Although modern alternatives to VLOOKUP exist, such as the more flexible XLOOKUP or the robust INDEX/MATCH combination, understanding the necessity of the **FALSE** argument in VLOOKUP remains a fundamental skill for any advanced Excel user. The examples below demonstrate clearly why defaulting to **FALSE** ensures your results are precise and dependable, particularly when dealing with non-sorted or textual data.

Illustrative Dataset for Comparison

To demonstrate the profound difference between **TRUE** and **FALSE**, we will use a small dataset representing team names and their corresponding point totals. Our goal is to retrieve the points for a list of teams using VLOOKUP, first using the approximate match setting (**TRUE**) and then switching to the accurate exact match setting (**FALSE**).

Observe the structure of the data below. The data in columns A and B represents the master table (our **table_array**), and the values in column D are the specific team names we wish to look up. Note that the team names in column A are not sorted alphabetically, which is a key factor when testing the **TRUE** argument.

	A	B	C	D	E	F
1	Team	Points				
2	Mavs	22				
3	Warriors	29				
4	Cavs	35				
5	Heat	13				
6	Thunder	18				
7	Rockets	29				
8	Spurs	24				
9	Lakers	10				
10	Nuggets	14				
11						
12						
13						
14						
15						
16						
17						
18						

Our primary objective in this exercise is to populate column E with the correct point totals corresponding to the team names listed in column D. The success or failure of the lookup operation hinges entirely on the selection of the final argument.

Case Study 1: Why TRUE Yields Inaccurate Results

In our first scenario, we deliberately choose **TRUE** for the `range_lookup` argument. We are attempting to find the points for the team name listed in cell D2 by searching the master table array `A2:B10`. The resulting formula, when applied starting in cell E2, looks like this:

```
=VLOOKUP(D2, $A$2:$B$10, 2, TRUE)
```

This formula instructs Excel to find an approximate match for the team name. Recall that the approximate match mode is optimized for sorted numerical ranges. Because our lookup column (A) contains text strings and is unsorted, the VLOOKUP function cannot execute the search logic correctly. It searches for a value less than or equal to the lookup value (based on internal character codes) and returns the wrong data.

The following screenshot illustrates the results when this formula is dragged down column E:

	A	B	C	D	E	F
1	Team	Points		Team Lookup	Points	
2	Mavs	22		Spurs	22	
3	Warriors	29		Mavs	22	
4	Cavs	35		Lakers	#N/A	
5	Heat	13				
6	Thunder	18				
7	Rockets	29				
8	Spurs	24				
9	Lakers	10				
10	Nuggets	14				
11						
12						
13						
14						
15						
16						
17						
18						

A careful inspection reveals a high degree of error. For example, the formula attempts to find "Team G" (D4) but returns 88 points (E4), which actually belongs to "Team H" in the source table. Similarly, the points returned for "Team B" (E5, value 75) actually belong to "Team D" (A5). This misalignment proves that when dealing with specific textual lookups, especially in unsorted data, the use of **TRUE** is dangerous and renders the resulting data unreliable for analysis or reporting. The errors are subtle because Excel returns a numerical result instead of an error message, giving a false sense of security that a valid match was found.

Case Study 2: Precision Using the FALSE Argument

In our second and correct scenario, we modify the final argument to **FALSE**. This simple change compels VLOOKUP to search only for an exact match. If the specified team name in column D exists in column A, its corresponding points from column B will be returned. If the team name does not exist, the function will correctly signal the absence of data.

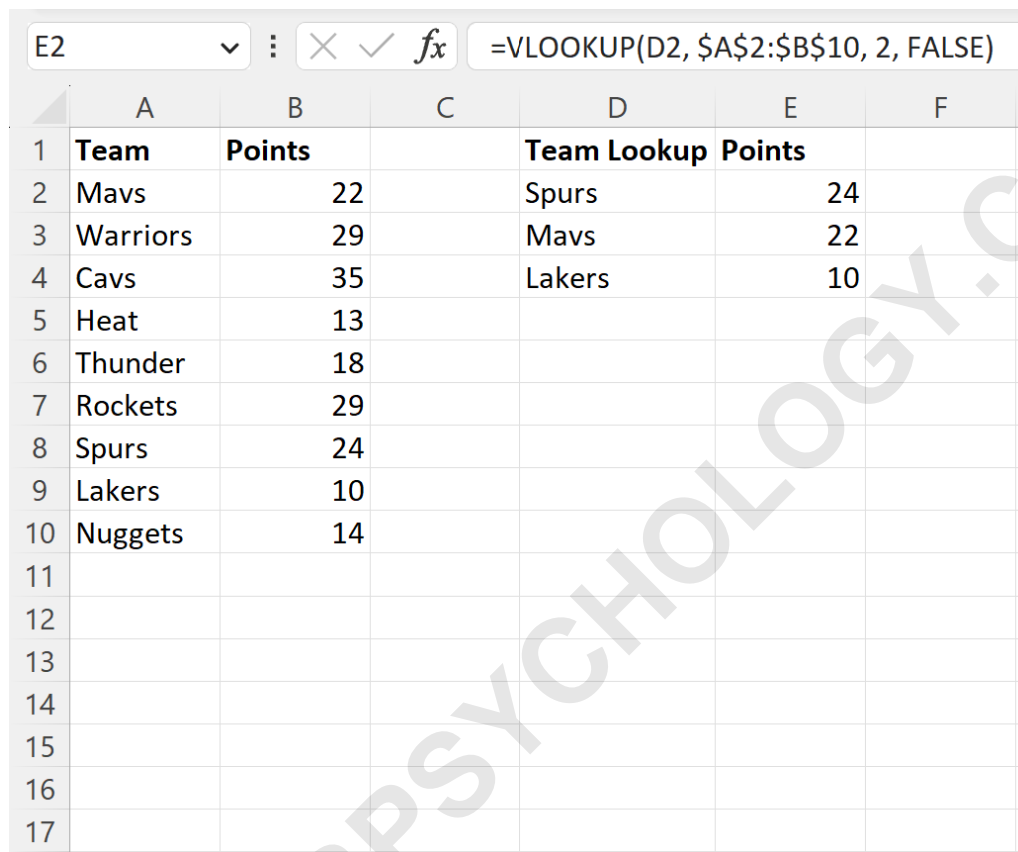
The updated formula, utilizing the **FALSE** argument, is:

=VLOOKUP(D2, \$A\$2:\$B\$10, 2, FALSE)

By specifying **FALSE**, we eliminate the need for sorting and ensure that the lookup operation is

precise. The function now scans the entire array, looking for the exact text string matching D2, D3, and so on. This approach is deterministic and predictable, regardless of whether the source data is organized or chaotic.

The following screenshot demonstrates the highly accurate results achieved by using the **FALSE** argument:



	A	B	C	D	E	F
1	Team	Points		Team Lookup	Points	
2	Mavs	22		Spurs	24	
3	Warriors	29		Mavs	22	
4	Cavs	35		Lakers	10	
5	Heat	13				
6	Thunder	18				
7	Rockets	29				
8	Spurs	24				
9	Lakers	10				
10	Nuggets	14				
11						
12						
13						
14						
15						
16						
17						

As evidenced in the results, every point value in column E now correctly corresponds to the team name in column D. The values are retrieved with precision because the function insisted on an exact textual match before returning the corresponding score. This example definitively proves that for most standard lookups involving text or unique identifiers, setting the `range_lookup` to **FALSE** is the only safe and reliable methodology. Furthermore, if a team name in column D were misspelled or absent from column A, the formula would correctly return the **#N/A** error, alerting the user to a missing data point, which is far preferable to returning an incorrect, misleading numerical value.

VLOOKUP Best Practices and Modern Alternatives

In summary, while the **TRUE** argument has specific niche applications for searching through sorted numerical thresholds, the general best practice for any standard data retrieval task in Excel is to

always explicitly set the final parameter of VLOOKUP to **FALSE**. This guarantees an exact match, eliminating the risk of data contamination due to unsorted source data or subtle variations in text strings. When **FALSE** is used, if the lookup value cannot be found exactly, Excel will return the **#N/A** error, providing clear feedback that the requested data is missing, allowing for immediate correction or handling of the exception.

It is important to remember this final constraint: When using **FALSE**, Excel will return **#N/A** if an exact match is not found. This error is not a failure of the function but an accurate report of the data's absence. Users often wrap VLOOKUP within the IFERROR or IFNA functions to handle these missing values gracefully, perhaps replacing the **#N/A** output with a zero, a blank cell, or a descriptive phrase like "Not Found."

Finally, as data management techniques evolve, many power users are transitioning away from VLOOKUP entirely. Functions like INDEX and MATCH offer greater flexibility, especially as they do not require the lookup column to be the leftmost column of the array. Even more recently, the introduction of the XLOOKUP function provides a simplified, modern alternative that defaults to an exact match, handles horizontal and vertical lookups, and allows for bidirectional searching, effectively replacing most traditional uses of VLOOKUP while offering superior error handling features. Nonetheless, mastering the **TRUE/FALSE** distinction in VLOOKUP remains a foundational skill for interacting with existing legacy spreadsheets and understanding the core mechanics of data retrieval in spreadsheet software.