

How to Find the First Match in Excel Using Multiple Criteria

Authored by
stats writer

February 20, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Find the First Match in Excel Using Multiple Criteria*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=131785>

Excel: Find First Occurrence Based on Multiple Criteria

Introduction to Advanced Data Retrieval in Microsoft Excel

In the modern landscape of **data analysis**, the ability to efficiently query a complex **dataset** is a fundamental skill for any professional working with **Microsoft Excel**. Often, users find themselves needing to extract a specific piece of information that satisfies not just one, but several distinct conditions. While basic lookup functions are suitable for simple tasks, more sophisticated scenarios require a robust combination of tools to ensure accuracy and speed. Understanding how to navigate these challenges is essential for maintaining a high level of productivity and ensuring that the insights derived from your **spreadsheet** are reliable.

The primary challenge when searching for the first occurrence based on multiple criteria lies in the inherent limitations of traditional functions like **VLOOKUP**, which are typically designed for single-column searches. To overcome this, **Excel** power users leverage the synergistic relationship between the **INDEX function** and the **MATCH function**. This combination offers unparalleled flexibility, allowing for horizontal, vertical, and multi-dimensional searches. By mastering these functions, you can move beyond simple data entry and into the realm of advanced **data management**, providing a more efficient and accurate way of locating specific information within a large **dataset**.

This tutorial provides a comprehensive deep dive into the mechanics of multi-criteria lookups. We will explore the underlying **Boolean logic** that makes these formulas work and provide a practical, step-by-step example involving a basketball performance **database**. By the end of this guide, you will be equipped with the knowledge to implement these advanced **formulas** in your own professional projects, streamlining your workflows and enhancing your analytical capabilities. This feature is particularly useful for organizing large amounts of data in **Excel** and performing high-level **statistical analysis**.

The Core Mechanics of the INDEX and MATCH Combination

To find the first occurrence of a value based on multiple criteria, one must utilize a specific **formula** structure that bypasses the standard limitations of **Excel** lookups. The **INDEX function** is used to return a value from a specific location within a range, while the **MATCH function** identifies the relative position of a value that meets your defined requirements. When these two are nested, they create a dynamic search engine capable of parsing through rows and columns with surgical precision. This method is widely considered superior to other lookup methods due to its non-volatile nature and its ability to handle array-based calculations without slowing down the **software** application.

You can use the following formula to find the first occurrence of a value in a column in Excel based on multiple criteria:

```
=INDEX(C2:C13,MATCH(1,INDEX((A2:A13=F1)*(B2:B13=F2)),),FALSE))
```

This particular **formula** returns the first value in the range **C2:C13** where the corresponding value in **A2:A13** is equal to the value in **cell reference F1** and the corresponding value in **B2:B13** is equal to the value in **cell reference F2**. The use of the number 1 within the **MATCH function** is a clever trick; it searches for the "True" state generated by the multiplication of the two criteria arrays. When **Excel** evaluates these conditions, it treats "True" as 1 and "False" as 0, meaning only the rows where both conditions are met will result in a 1, which the **MATCH function** then identifies as the target row.

The following example shows how to use this **formula** in practice, demonstrating its utility in a real-world scenario where data points must be isolated from a cluttered **spreadsheet** environment. By utilizing this structure, you ensure that your **data analysis** remains both scalable and transparent, as the logic is clearly defined within the nested arguments of the function. This approach is highly recommended for anyone looking to optimize their **data processing** tasks and minimize the risk of manual lookup errors.

Practical Demonstration: Basketball Performance Metrics

Example: Find First Occurrence Based on Multiple Criteria in Excel

Suppose we have the following **dataset** that contains information about points scored by various basketball players across different teams and positions. In professional sports **analytics**, such data is common, and being able to quickly filter through it to find the first instance of a specific performance metric is invaluable for coaches and analysts alike. In this scenario, we have columns representing the "Team," the "Position" of the player, and the "Points" they scored in a given game.

	A	B	C	D	E	F
1	Team	Position	Points			
2	Mavs	Guard	22			
3	Mavs	Forward	14			
4	Mavs	Forward	17			
5	Mavs	Guard	19			
6	Spurs	Guard	30			
7	Spurs	Forward	31			
8	Spurs	Forward	37			
9	Spurs	Guard	20			
10	Rockets	Forward	28			
11	Rockets	Guard	12			
12	Rockets	Guard	16			
13	Rockets	Guard	19			
14						
15						
16						
17						
18						
19						

Suppose we would like to return the points value for the first occurrence of a player who is on the **Spurs** team and has a position of **Forward**. This requires the **Excel** engine to scan the "Team" column for "Spurs" and the "Position" column for "Forward" simultaneously. The first row that satisfies both of these conditions will then yield the value from the "Points" column. This type of multi-factor filtering is essential when a single criterion, such as the team name, is not unique enough to identify the specific record you are looking for within the **spreadsheet**.

We can specify this criteria in cells **F1** and **F2**, then type the following **formula** into cell **F3** to execute the search. By referencing external cells for our criteria, we make the **spreadsheet** dynamic and interactive, allowing users to change the search parameters without having to rewrite the underlying **Excel syntax**. This is a best practice in **user interface** design for financial models and data dashboards.

=INDEX(C2:C13,MATCH(1,INDEX((A2:A13=F1)*(B2:B13=F2)),),FALSE))

The following screenshot shows how to use this **formula** in practice, illustrating the seamless integration of the criteria and the resulting output. This visual representation highlights how the

software identifies the intersection of the specified conditions to deliver the correct data point. This visual confirmation is a critical step in verifying that the **Boolean logic** within your **array formula** is functioning as intended.

	A	B	C	D	E	F	G	H
1	Team	Position	Points		Team	Spurs		
2	Mavs	Guard	22		Position	Forward		
3	Mavs	Forward	14		Points of First Occurrence	31		
4	Mavs	Forward	17					
5	Mavs	Guard	19					
6	Spurs	Guard	30					
7	Spurs	Forward	31					
8	Spurs	Forward	37					
9	Spurs	Guard	20					
10	Rockets	Forward	28					
11	Rockets	Guard	12					
12	Rockets	Guard	16					
13	Rockets	Guard	19					
14								
15								
16								
17								

The **formula** returns a points value of **31**, since this represents the points value for the first player to be on the **Spurs** team and have a position of **Forward**. It is important to note that if there were other "Spurs Forwards" lower down in the **dataset**, they would be ignored by this specific **formula**, as the **MATCH function** is configured to stop at the very first match it encounters. This behavior is ideal for finding the earliest instance in a chronologically ordered **list** or the primary entry in a prioritized **database**.

Dynamic Data Updates and Interactivity

One of the most powerful aspects of using the **INDEX function** and **MATCH function** combination is its dynamic nature. Unlike static filters or manual searches, this **formula** updates in real-time. Note that if we change the criteria in cells **F1** and **F2**, the **formula** will automatically return a new player who matches the new criteria. This interactivity is what transforms a simple table into a functional **data analysis** tool capable of providing instant answers to complex queries.

For example, suppose we change the team to **Rockets** and the position to **Guard**. As soon as the values in the **cell references** are updated, **Excel** recalculates the **array formula**, scanning the ranges once more to find the new first occurrence. This eliminates the need for repetitive manual

tasks and ensures that the information displayed is always current based on the inputs provided by the user.

	A	B	C	D	E	F	G	H
1	Team	Position	Points		Team	Rockets		
2	Mavs	Guard	22		Position	Guard		
3	Mavs	Forward	14		Points of First Occurrence	12		
4	Mavs	Forward	17					
5	Mavs	Guard	19					
6	Spurs	Guard	30					
7	Spurs	Forward	31					
8	Spurs	Forward	37					
9	Spurs	Guard	20					
10	Rockets	Forward	28					
11	Rockets	Guard	12					
12	Rockets	Guard	16					
13	Rockets	Guard	19					
14								
15								

The **formula** correctly returns a value of **12**, which is the points value that corresponds to the first player to be on the **Rockets** team and have a position of **Guard**. This demonstrates the consistency of the **logic** across different data points. Whether you are dealing with a dozen rows or thousands of records, the structural integrity of the **lookup** remains the same, providing a scalable solution for various **data management** needs.

Understanding the Logic: How Array Multiplication Works

To truly master this technique, one must understand the **Boolean logic** occurring behind the scenes. When we write $(A2:A13=F1) * (B2:B13=F2)$, **Excel** creates two separate arrays of "True" and "False" values. For instance, if the first row matches the team but not the position, the result is `True * False`. In mathematical terms within a **spreadsheet**, this is calculated as $1 * 0$, which equals 0. Only when both conditions are `True` (i.e., $1 * 1$) does the result equal 1.

This multiplication process effectively creates a single array consisting of 1s and 0s. The **MATCH function** then searches for the number 1 within this new array. By specifying `FALSE` (or 0) as the last argument in the **MATCH function**, we are telling **Excel** to find an exact match. This ensures that the function doesn't settle for the nearest value but instead identifies the precise row where all criteria are satisfied simultaneously. This use of **binary logic** is a hallmark of advanced **computer science** principles applied to everyday **business intelligence** tasks.

Furthermore, the inner **INDEX function** used within the **MATCH function** is a sophisticated way to handle the array without requiring the user to press Ctrl+Shift+Enter in older versions of **Excel**. It forces the **software** to evaluate the array calculation internally, making the **formula** more robust and less prone to user error. This technical nuance is part of why the **INDEX/MATCH** combo has remained a staple of professional **data analysis** for decades, even with the introduction of newer functions like **XLOOKUP**.

Comparing INDEX/MATCH with Modern Alternatives

While the **INDEX/MATCH** method is a classic and highly reliable approach, it is worth noting the existence of the **XLOOKUP function**, which was introduced in more recent versions of **Office 365**. **XLOOKUP** simplifies the syntax for multi-criteria searches by allowing users to concatenate criteria using the ampersand (&) symbol. However, **INDEX/MATCH** remains the preferred choice for many analysts who require backward compatibility with older versions of **Excel** or who prefer the explicit control that the nested structure provides.

The advantages of **INDEX/MATCH** over other methods include:

Non-volatile performance: It does not trigger a recalculation every time a cell is changed elsewhere in the **spreadsheet**, unlike functions like **OFFSET** or **INDIRECT**.

Leftward lookups: Unlike **VLOOKUP**, it can easily return values from columns to the left of the criteria columns.

Granular Control: By breaking the search into two functions, users can more easily **debug** complex **formulas** by evaluating the **MATCH** and **INDEX** components separately.

Memory Efficiency: For extremely large **datasets**, **INDEX/MATCH** is often faster because it only processes the specific columns required rather than the entire table array.

Ultimately, choosing between these methods depends on your specific environment and the requirements of your **data analysis** project. However, having a deep understanding of the **INDEX/MATCH** logic is essential for any power user, as it forms the basis for many other advanced **Excel** techniques and **programming** logic found in more advanced **data science** tools.

Best Practices for Structuring Excel Data

To ensure that your **lookup formulas** work correctly every time, it is vital to maintain a clean and well-structured **dataset**. This involves ensuring that there are no leading or trailing spaces in your text fields, as these can cause the **MATCH function** to fail. Utilizing the **TRIM function** or **Excel Tables** (List Objects) can help in maintaining data integrity. When your data is formatted as a **table**, you can use structured references, which make your **formulas** much easier to read and maintain.

Additionally, consider the following organizational tips:

Standardize Data Entry: Use **data validation** drop-down menus for criteria cells like **F1** and **F2** to prevent typos that would result in **#N/A errors**.

Avoid Merged Cells: Merged cells are the enemy of **array formulas** and can cause unpredictable results when calculating ranges.

Use Named Ranges: Define names for your columns (e.g., "TeamRange," "PositionRange") to make your **formula syntax** more intuitive and less reliant on abstract **cell references**.

Keep Data Contiguous: Ensure there are no entirely blank rows or columns within your **dataset**, as this can interrupt the search logic of certain functions.

Document Your Logic: Use **comments** or a dedicated documentation sheet to explain complex **formulas** to other users who may interact with your **spreadsheet**.

By following these guidelines, you create a professional environment where **data analysis** can flourish. Clean data is the foundation of any successful **business intelligence** initiative, and the **formulas** you build will only be as good as the information they are processing. Investing time in data preparation saves significant time in the **troubleshooting** and reporting phases of your project.

Conclusion: Elevating Your Excel Proficiency

Mastering the ability to find the first occurrence based on multiple criteria is a significant milestone in your journey toward **Excel** mastery. This technique not only solves a common practical problem but also introduces you to the concept of **array processing** and **Boolean logic**, which are critical for more advanced **automation** and **data modeling**. By combining the **INDEX function** and **MATCH function**, you have developed a tool that is both flexible and powerful, capable of handling the demands of modern **data analysis**.

As you continue to explore the capabilities of **Microsoft Excel**, remember that the most effective solutions are often those that are clear, dynamic, and easy for others to understand. The multi-criteria lookup is a perfect example of this balance. It provides a sophisticated result while relying on a logical framework that can be **deconstructed** and explained. This transparency is vital when presenting your findings to stakeholders or collaborating with teammates on a shared **spreadsheet**.

The following tutorials explain how to perform other common operations in **Excel**, helping you to further expand your toolkit and become a more effective **data analyst**. Whether you are working in finance, marketing, or **data science**, these skills will serve as a valuable asset in your professional development, allowing you to turn raw data into actionable insights with confidence and precision.