Learn How to Remove Quotes from Excel Cells: Two Practical Methods

Authored by

Mohammed looti

November 10, 2025

Dealing with inconsistent data formatting is a common challenge when working in Microsoft Excel. One frequent formatting issue involves extraneous quotation marks appearing within cell values. These quotes often result from importing data from external sources, such as CSV files, where text qualifiers are used to manage complex strings. Fortunately, Excel offers powerful tools to efficiently clean this data. We will explore two reliable methods for removing these quotes:

Method 1: Utilize the powerful **Find and Replace** feature for static data cleansing.
Method 2: Employ the dynamic **SUBSTITUTE function** for formula-based manipulation, preserving the original data.

The following examples demonstrate the practical application of each technique using a sample list of basketball team names that contain unwanted quotation marks:

	A	B	C	D	E
1	**Team**				
2	"Mavs"				
3	"Spurs"				
4	"Rockets"				
5	"Kings"				
6	"Warriors"				
7	"Nets"				
8	"Lakers"				
9	"Thunder"				
10	"Blazers"				
11	"Jazz"				
12					
13					
14					
15					
16					

Understanding these techniques is essential for maintaining data integrity and ensuring that your data is ready for analysis and reporting.

Method 1: Precision Removal using the Find and Replace Feature

The **Find and Replace** feature is the quickest and most straightforward way to statically remove specific characters, including quotation marks, from a selected range of cells. This method is highly recommended when you need a permanent, one-time cleanup of your dataset and do not require the original data structure to remain intact.

When quotes are present, they can interfere with various Excel operations, particularly those relying on numerical values or specific text formats. Although the **Find and Replace** operation is simple, it executes a powerful global replacement across the specified range. It functions by locating every instance of the specified character (in this case, the double quote, `"`) and replacing it with the content defined in the "Replace with" field, which we will intentionally leave blank to achieve removal.

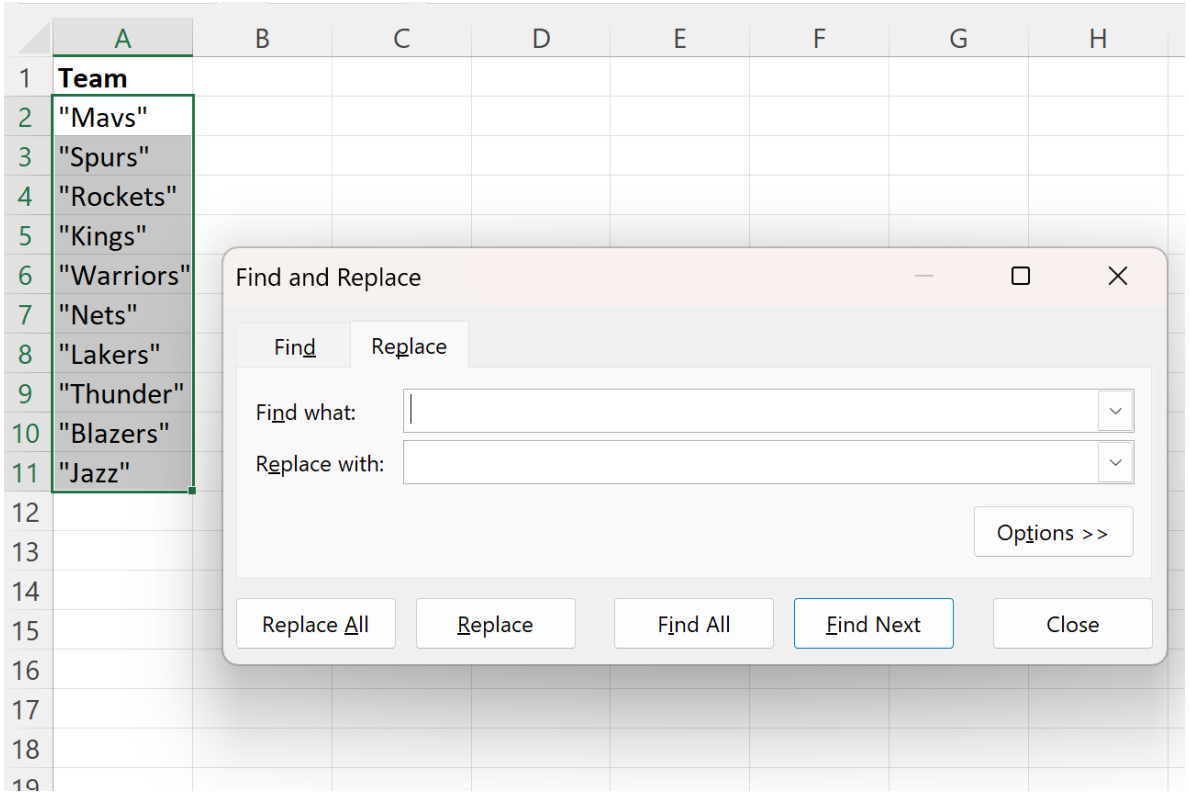
Before proceeding, it is crucial to recognize that this method modifies the data in place. If data provenance or the need for an audit trail is critical, consider duplicating the data column before initiating the replacement process. This ensures that a record of the original, quoted strings is maintained, allowing for backtracking if necessary. However, for sheer speed and efficiency in cleaning large datasets, **Find and Replace** remains the superior choice.

Step-by-Step Implementation of Find and Replace

Executing the quote removal via **Find and Replace** requires careful selection and input to ensure that only the intended characters are affected. The process begins with isolating the data range requiring modification, which minimizes the risk of inadvertently altering data in adjacent, unrelated columns.

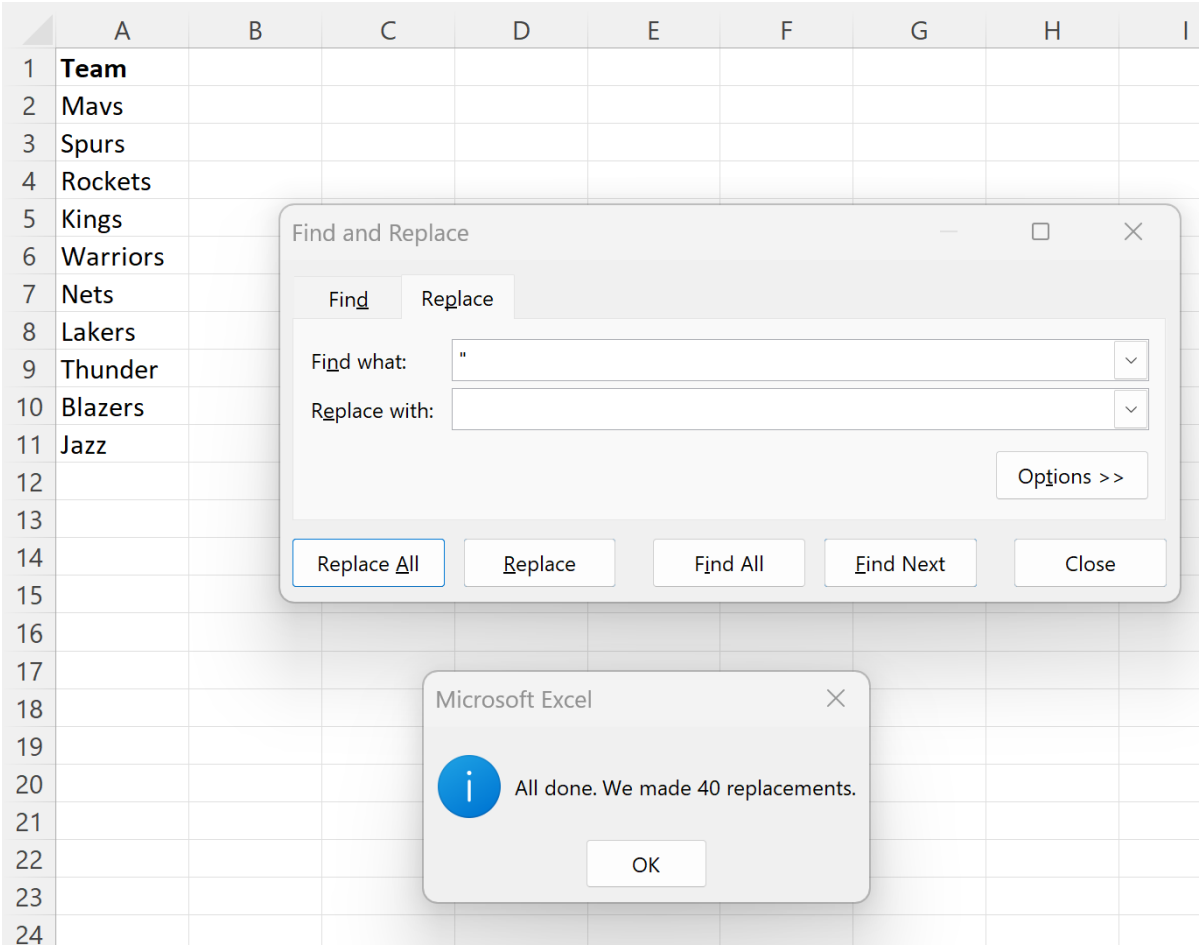
First, highlight the cell range containing the data you wish to clean. In our example, this corresponds to range **A2:A11**. Precise selection ensures the operation's scope is limited solely to the target data.

Next, initiate the **Find and Replace** dialogue box. This is most easily achieved by pressing the keyboard shortcut **Ctrl + H** (or Cmd + Shift + H on macOS). This action immediately opens the necessary interface within Excel:



Within the dialogue box, input the character you intend to remove. Type **"** (the double quotation mark) into the **Find what** box. Crucially, leave the **Replace with** box completely blank. By defining an empty replacement string, you instruct Excel to effectively delete every instance of the quote it finds, rather than substituting it with another character or space.

Once the parameters are set, click the **Replace All** button. Excel will process the selected range, removing all quotes. A confirmation message box will then appear, detailing the exact number of replacements performed, providing immediate verification that the operation was successful across the entire dataset:



Observe the results in the original column: the extraneous quotes surrounding the team names have been instantaneously and permanently removed. This method is highly efficient for large-scale static cleanup tasks, transforming the raw, quoted data into clean, usable text strings.

Method 2: Dynamic Cleaning using the SUBSTITUTE Function

When data integrity is paramount, or when you need a solution that automatically updates if the source data changes, utilizing a formula is the preferred approach. The **SUBSTITUTE function** in Excel is specifically designed for replacing specific text within a text string. Unlike **Find and Replace**, this method creates a new, cleaned output in a separate column, preserving the original quoted data.

The structure of the **SUBSTITUTE function** is `=SUBSTITUTE(text, old_text, new_text, )`. To remove the double quotes, we must instruct the function to look for the double quote character (`old_text`) and replace it with nothing (`new_text = ""`). However, directly typing `""` inside the formula to represent the double quote character can lead to syntax errors, as the double quotes are also used to delineate the formula's text arguments. To bypass this ambiguity, we employ the

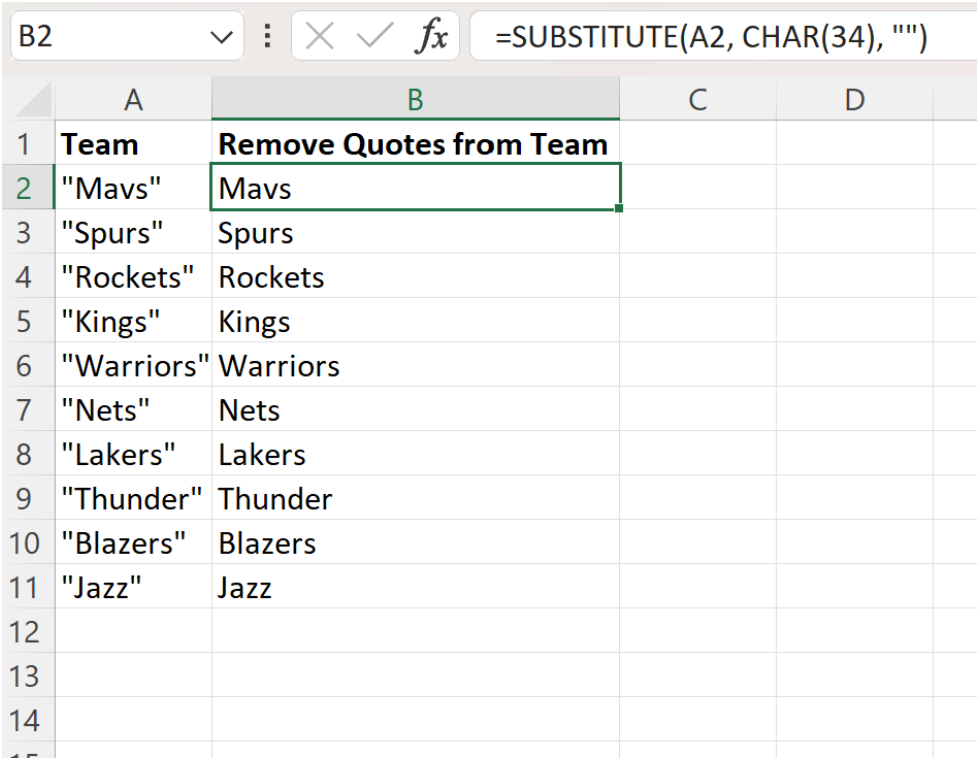
CHAR function.

The **CHAR** function converts a numerical character code (typically based on the ASCII or Unicode system) into its corresponding text character. The standard character code for the double quotation mark (") is 34. Therefore, using **CHAR(34)** reliably represents the double quote character within the formula without causing parsing conflicts. This technical detail is key to making the **SUBSTITUTE function** work correctly for quote removal.

We can implement this formula by typing the following expression into cell **B2** to reference and clean the data from cell **A2**:

=SUBSTITUTE(A2, CHAR(34),)

Note that the `new_text` argument is left blank, achieving the removal effect. After entering the formula in the first cell, the dynamic nature of Excel allows for efficient replication. We can then click and drag the fill handle (the small square at the bottom-right corner of cell B2) down to apply this formula to each corresponding cell in column B:



Column B now displays the cleaned data, reflecting each team name from column A with the quotes successfully removed. This method is preferred when your spreadsheet requires a persistent, formulaic link to the source data, ensuring that any subsequent changes in column A are immediately reflected as clean data in column B.

Advanced Considerations and Best Practices

While both the **Find and Replace** method and the **SUBSTITUTE function** are effective, choosing the right tool depends heavily on the context of your data workflow.

If you opt for the formulaic approach using **SUBSTITUTE**, remember that the resulting column (Column B in our example) contains formulas, not static values. If you intend to use this cleaned data elsewhere--for example, pasting it into another application or using it as criteria for advanced filtering--you must convert the formulas to static values. This is accomplished by copying the cleaned column (B), and then using the **Paste Special > Values** option back onto itself or into a new location. This action permanently replaces the formula with the resulting text string.

Furthermore, it is important to distinguish between single quotes (apostrophes, `'`) and double quotes (`"`). The method demonstrated here specifically targets double quotes (using **CHAR(34)** or typing `"` in **Find what**). If your data is enclosed in single quotes, you would need to adjust the method accordingly. For the **SUBSTITUTE function**, you would use **CHAR(39)**, as 39 is the ASCII character code for the apostrophe.

Finally, always perform a visual check after any bulk data manipulation, especially when dealing with quoted strings. Sometimes, quotes might be embedded within a word rather than just surrounding the whole text. Both methods effectively remove all instances of the specified character, so be mindful if the quote character is intended to be part of the final text (e.g., in names like O'Malley or titles containing quotes).

Additional Resources

Mastering data cleaning techniques is essential for effective data analysis in Excel. We encourage readers to explore additional tutorials covering other common data manipulation tasks:

How to handle leading and trailing spaces in imported data.
Techniques for splitting complex text strings into multiple columns.
Methods for converting text formatted numbers back into numerical values.

These skills collectively ensure your datasets are always standardized and prepared for complex calculations or visualizations.