

Learning to Sort Excel Pivot Tables by Multiple Columns

Authored by
Mohammed loot

November 14, 2025

RECOMMENDED CITATION

Mohammed loot (2025). *Learning to Sort Excel Pivot Tables by Multiple Columns*.
PSYCHOLOGICAL STATISTICS. Retrieved from
<https://statistics.arabpsychology.com/?p=945>

The Critical Need for Multi-Level Sorting in Excel Pivot Tables

When performing rigorous data analysis using [Excel](#), the [Pivot Table](#) is universally recognized as an essential utility. It excels at summarizing, aggregating, and presenting massive volumes of information derived from complex [datasets](#). However, analysts frequently encounter a significant hurdle: the native limitations when attempting a multi-criteria [sort](#) directly on the row labels. While sorting by a single field is straightforward, achieving a precise hierarchical order--where one column dictates the primary ranking and a second column serves as a tie-breaker--is not natively supported by the standard Pivot Table interface. This restriction often necessitates advanced techniques to derive the clearest, most actionable insights from summarized data.

Consider a common analytical requirement, such as ranking sports teams or sales territories based on performance metrics. The goal is often multi-layered: teams must first be ranked by the highest total **Sum of Points** (descending). If two or more teams achieve an identical point total, the secondary requirement is to use the total **Sum of Assists** (also descending) to break the tie and finalize the ranking. While Excel allows you to sort the Pivot Table rows individually by the Points column or the Assists column, there is no simple, built-in mechanism to combine these criteria into a single, composite sort order. Attempts to apply a manual sequence of sorts inevitably fail, as the secondary sort operation overrides the results of the primary one, preventing the establishment of the desired hierarchical structure.

The image below illustrates a typical [Pivot Table](#) setup that aggregates results. Notice how a simple sort based on points alone ignores the crucial tie-breaking information contained in the assists column. Overcoming this fundamental limitation requires a creative solution that extends beyond basic data manipulation. Fortunately, by leveraging one of Excel's most powerful advanced features, we can construct a method that achieves complex, multi-level sorting with absolute precision and elegance.

	A	B	C	D	E	F	G
1	Team	Points	Assists		Row Labels ↓↑	Sum of Points	Sum of Assists
2	Warriors	20	4		Lakers	81	27
3	Lakers	13	8		Thunder	64	17
4	Lakers	15	8		Warriors	64	21
5	Lakers	35	6		Grand Total	209	65
6	Thunder	29	9				
7	Warriors	24	14				
8	Thunder	13	3				
9	Lakers	18	5				
10	Thunder	22	5				
11	Warriors	20	3				
12							
13							
14							
15							
16							

Introducing the Weighted Calculated Field Workaround

The most robust and streamlined solution to the multi-column sorting challenge within Excel Pivot Tables involves the strategic deployment of a [Calculated Field](#). This versatile tool allows the user to define a custom metric directly within the Pivot Table environment, enabling the combination of multiple sorting criteria into a single, comprehensive numerical value. This composite value acts as a synthetic sorting key, precisely engineered to encapsulate the necessary logic for hierarchical ordering.

The core concept behind this technique is the careful construction of a weighted [formula](#). This [formula](#) must assign an overwhelmingly high weight to the primary [sort](#) column (e.g., Points) and a comparatively tiny, tie-breaking weight to the secondary sort column (e.g., Assists). By integrating these metrics into one numerical key, we effectively bypass the inherent software limitations that prevent sorting by multiple value fields simultaneously. Once the Pivot Table is sorted based on this temporary, composite metric, the desired hierarchical order is immediately established and locked into the row labels.

This method establishes an elegant and professional workflow: we create the helper column, utilize it exclusively for sorting the row labels, and then conceal it from view. This leaves the user with a perfectly organized [Pivot Table](#) that displays only the final, relevant summary metrics. The subsequent sections of this guide provide a detailed, step-by-step walkthrough for implementing this advanced technique. We will break down the exact [formula](#) structure, explain the rationale

behind choosing the weighting factors, and demonstrate how to apply and subsequently hide this powerful helper column for a polished final presentation in [Excel](#).

Step 1: Preparing and Structuring Your Source Dataset

The success of any Pivot Table analysis, particularly one involving complex calculations, is contingent upon the integrity and structure of the source data. A well-organized [dataset](#) is the indispensable foundation for accurate summarization and manipulation. For this specific tutorial, we will utilize a sample statistical dataset designed to illustrate the sorting challenge. This data features three crucial columns: the 'Team' name, the individual 'Points' scored, and the individual 'Assists' made.

It is imperative that the raw data adheres to a clean, tabular format. This means every column must have a unique, descriptive header, and the data range should be free of merged cells, blank rows, or subtotals. This structure allows the Pivot Table generator to correctly identify and categorize all fields. To follow the example precisely, please input the following sample data into a new worksheet. This data serves as the source for our Pivot Table and demonstrates the necessity of a multi-column sort, as several teams share identical point totals, requiring the assists metric as a tie-breaker.

	A	B	C	D	E
1	Team	Points	Assists		
2	Warriors	20	4		
3	Lakers	13	8		
4	Lakers	15	8		
5	Lakers	35	6		
6	Thunder	29	9		
7	Warriors	24	14		
8	Thunder	13	3		
9	Lakers	18	5		
10	Thunder	22	5		
11	Warriors	20	3		
12					
13					
14					
15					
16					
17					

Ensuring meticulous accuracy during this data entry phase is vital, as all subsequent steps--including the Pivot Table creation and the definition of the [Calculated Field](#)--rely entirely on correct

references to these source columns. After inputting the data, a recommended best practice is to convert the raw range into an official Excel Table (via the Insert tab). This practice simplifies data management and ensures the Pivot Table dynamically updates if new rows are added to the source data.

Step 2: Initializing and Configuring the Pivot Table Structure

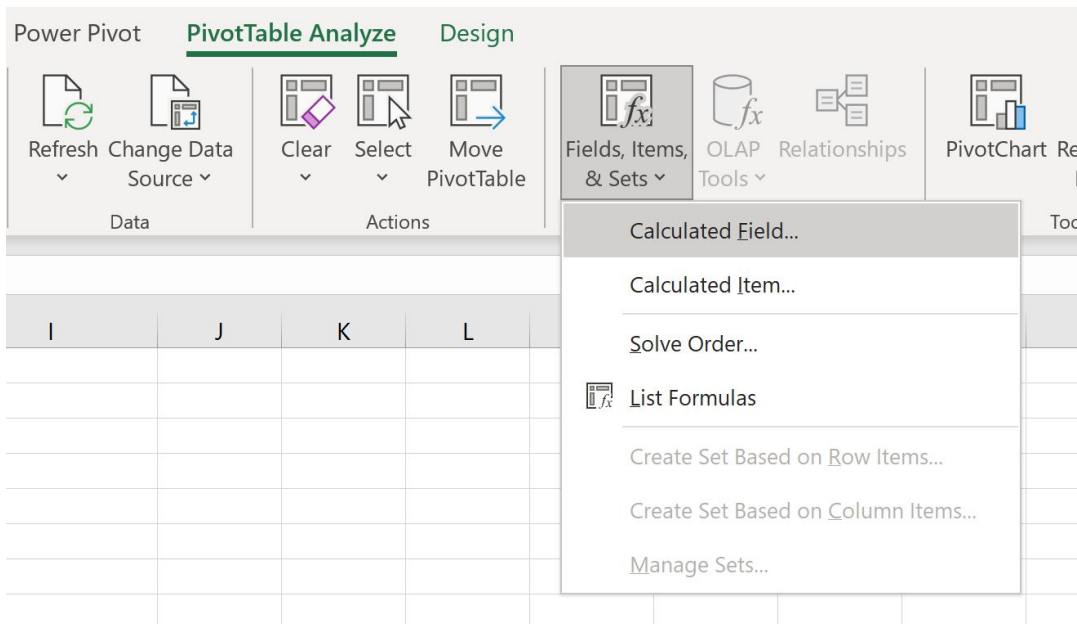
Once the source data is meticulously prepared, the next phase involves creating the initial [Pivot Table](#). This summary table is designed to aggregate the raw, player-level data into team-level totals for both points and assists. To begin, select the entire range of your prepared data, including the column headers. Navigate to the **Insert** tab on the Excel ribbon and click the **PivotTable** button. In the resulting 'Create PivotTable' dialog box, confirm that the correct data range is selected and choose the option to place the Pivot Table on a **New Worksheet** to maintain optimal clarity and separation from the source data.

With the Pivot Table framework established on the new sheet, proceed to configure the fields using the PivotTable Field List pane. Drag the 'Team' field into the **Rows** area, defining the primary grouping structure. Subsequently, drag both the 'Points' field and the 'Assists' field into the **Values** area. By default, Excel will summarize these numerical fields using the **Sum** function, correctly providing us with the total points and total assists accumulated by each respective team.

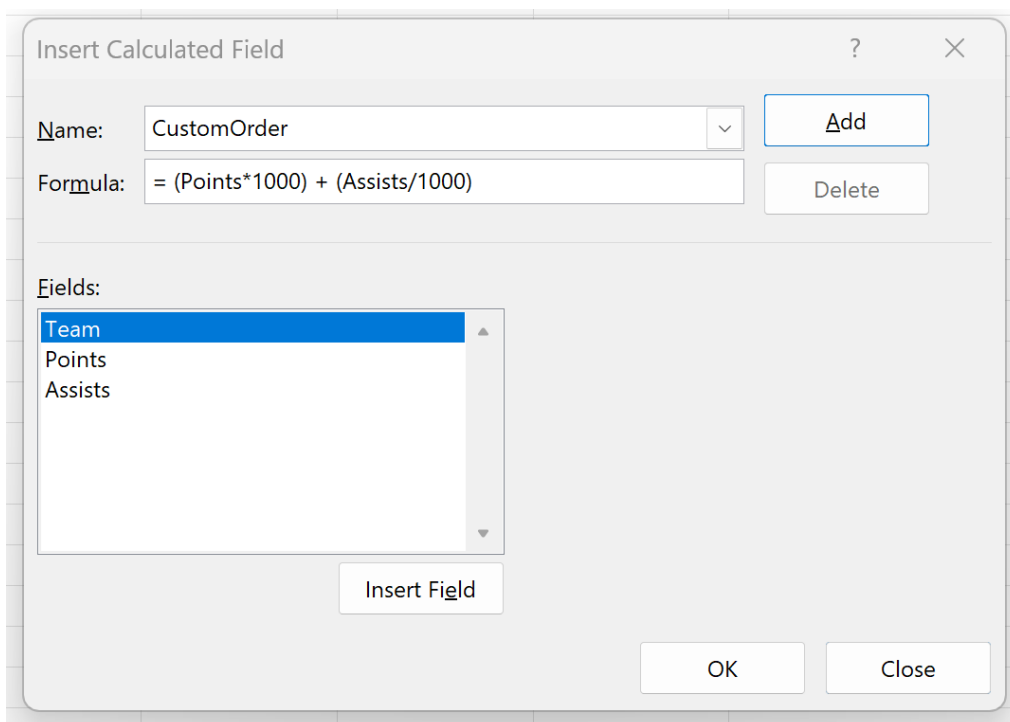
The resulting table, as illustrated in the preceding image, displays the necessary summary data. However, at this stage, the teams are typically sorted alphabetically by their name (the Row Label) or perhaps based on their appearance order in the source data. Our ultimate objective is to overwrite this default arrangement and enforce a functional ranking hierarchy: points as the primary rank, followed by assists as the secondary rank. This preliminary setup is the essential precursor before we introduce our custom sorting logic in the subsequent, critical step.

Step 3: Constructing the Custom Sort Logic with a Calculated Field

This step represents the technical core of the solution. To successfully sort by **Sum of Points** (Primary) and then by **Sum of Assists** (Secondary), we must mathematically combine these two metrics into a single numerical key. We achieve this by using the [Calculated Field](#) feature. First, click anywhere inside your Pivot Table to activate the specialized PivotTable contextual tools. Navigate to the **PivotTable Analyze** tab (or the **Options** tab in legacy versions) on the ribbon. Within the 'Calculations' group, click on **Fields, Items & Sets**, and then select **Calculated Field**.



The 'Insert Calculated Field' dialog box requires two pieces of input: a descriptive field name and the defining mathematical [formula](#). Name the field "CustomOrder" for easy reference. For the core logic, input the following expression into the **Formula** box: **= (Points * 1000) + (Assists / 1000)**. After confirming the syntax is correct, click **Add**, and then click **OK**.



Upon creation, the new field, automatically labeled "Sum of CustomOrder," is inserted into the

Values area of your Pivot Table. This field now holds the aggregated numerical value determined by our custom [formula](#) for each team. The weighting mechanism used in this [formula](#) is critical to its function: by multiplying **Points** by a significant factor (1000 in this example), we ensure that Points contribute the dominant integer part of the combined score, thereby establishing the primary sort hierarchy. Conversely, dividing **Assists** by the same large factor (1000) guarantees that the Assists value only influences the decimal portion of the combined number. This precise design ensures that the Assists metric can only affect the final ranking when the Points totals are identical, functioning perfectly as the secondary tie-breaker.

E	F	G	H	
Row Labels	Sum of Points	Sum of Assists	Sum of CustomOrder	
Lakers	81	27	81000.027	
Thunder	64	17	64000.017	
Warriors	64	21	64000.021	
Grand Total	209	65	209000.065	

Step 4: Applying the Custom Sort to Finalize the Order

With the complex preparatory work complete, the final application of the multi-column sort becomes remarkably straightforward. Since the "Sum of CustomOrder" field contains a single numerical value that mathematically reflects both sorting criteria, we only need to sort the Pivot Table based on this one column. The inherent structure of the calculated value will automatically enforce the hierarchical ordering we designed.

To execute the sort, locate any cell within the data area of the newly created "Sum of CustomOrder" column. Right-click on that cell to access the contextual menu. Hover your cursor over the **Sort** option, and then select **Sort Largest to Smallest**. This command instructs the Pivot Table to [sort](#) the row labels (Teams) according to the descending magnitude of the custom field values.

	E	F	G	H	I
Row Labels ▼▼	Sum of Points	Sum of Assists	Sum of CustomOrder		
Lakers	81	27	81000.027		
Warriors	64	21	64000.021		
Thunder	64	17	64000.017		
Grand Total	209	65	209000.065		

Immediately following this step, the rows of your Pivot Table will reorder themselves. You will observe that the teams are ranked descendingly by their Points totals. Most importantly, for teams sharing the same point totals (like Team A and Team B in our example), the ranking is now correctly resolved by comparing their total Assists, successfully achieving the precise, multi-level sort order required for the analysis.

Step 5: Finalizing the Pivot Table by Concealing the Helper Column

The "Sum of CustomOrder" column has successfully performed its critical function--facilitating the advanced sort operation. In nearly all final reporting scenarios, this numerical helper column is extraneous to the presentation and can clutter the visual analysis. Crucially, hiding this field does not disrupt or alter the established sort order, which remains permanently locked to the row labels.

To finalize your professional report, the helper column must be concealed. Locate the column header for the "Sum of CustomOrder" field and right-click directly on it. A context menu will appear. From the available options, select **Hide**. Executing this action removes the column from view, leaving your Pivot Table immaculately clean, displaying only the essential summary metrics (Team, Sum of Points, Sum of Assists) in the highly accurate, hierarchically sorted order you meticulously designed.

helper column ensures a clean, professional output, transforming raw summary data into meaningful, ordered information optimized for stakeholder review.

Additional Resources for Advanced Excel Proficiency

To further refine your proficiency in Excel and explore related advanced data manipulation techniques, consider exploring the following tutorials. These resources cover a range of common tasks and provide supplementary insights into leveraging Excel's full capabilities for effective data analysis and reporting.

Exploring complex data filtering and conditional formatting rules.

Techniques for managing and consolidating data from multiple worksheets.

In-depth guide to using the GETPIVOTDATA function for dynamic reporting.