

How to Autofill Data Between Sheets in Google Sheets: A Step-by-Step Guide

Authored by
stats writer

November 30, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Autofill Data Between Sheets in Google Sheets: A Step-by-Step Guide*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=102485>

Google Sheets stands as a cornerstone tool for collaborative data management and analysis in the modern digital workplace. A fundamental necessity when handling complex datasets is the ability to efficiently consolidate information distributed across multiple sheets or tabs within the same workbook. This process, often required for aggregation, reporting, or standardized input generation, is vastly simplified through the powerful built-in functionality known as Autofill. Understanding how to leverage this feature to pull values from a source sheet into a destination sheet is crucial for optimizing workflows and ensuring data integrity across your spreadsheet models. This capability moves beyond simple copy-pasting, offering a dynamic link that updates automatically if the source data changes.

The core benefit of using cross-sheet Autofill lies in its efficiency, especially when dealing with extensive columns or rows that require sequential population. Instead of manually inputting or copying hundreds of values, which introduces significant risk of human error, the Autofill mechanism interprets the required pattern and rapidly applies the appropriate cell reference structure down the range. Whether you are mirroring timestamps, pulling unique identifiers, or transferring computed metrics, mastering this technique ensures that your linked sheets remain synchronized and accurate. This deep dive will explore the precise steps required to establish this connection and execute the autofill successfully, transforming disparate data into a cohesive, interconnected spreadsheet environment.

Understanding Cross-Sheet Referencing

Before executing the Autofill function, it is imperative to grasp the concept of cross-sheet referencing, which forms the bedrock of this operation. Unlike referencing a cell within the current sheet (e.g., **=B2**), referencing a cell located in another sheet requires specifying the sheet name followed by an exclamation mark (!) and the target cell reference (e.g., **=SheetName!B2**). This explicit nomenclature tells Google Sheets exactly where to retrieve the value. It is important to note that if the sheet name contains spaces or special characters, it must be enclosed in single quotes, such as **='Monthly Sales'!B2**, to ensure the formula is correctly parsed by the application.

When preparing to use Autofill, it is critical to set up the initial formula correctly. The first formula you write acts as the template for every subsequent cell in the range you intend to fill. When you drag the fill handle (the small square at the bottom-right corner of the cell), Google Sheets intelligently adjusts the row and column indicators based on the direction of the drag. This behavior, known as relative referencing, is what makes the Autofill feature so effective for sequential data transfer across sheets. For instance, if you start with a reference to **Sheet1!B2** and drag down one row, the next cell will automatically update its formula to **Sheet1!B3**, ensuring a perfect mapping of the original column contents.

Prerequisites: Setting Up Your Source Data

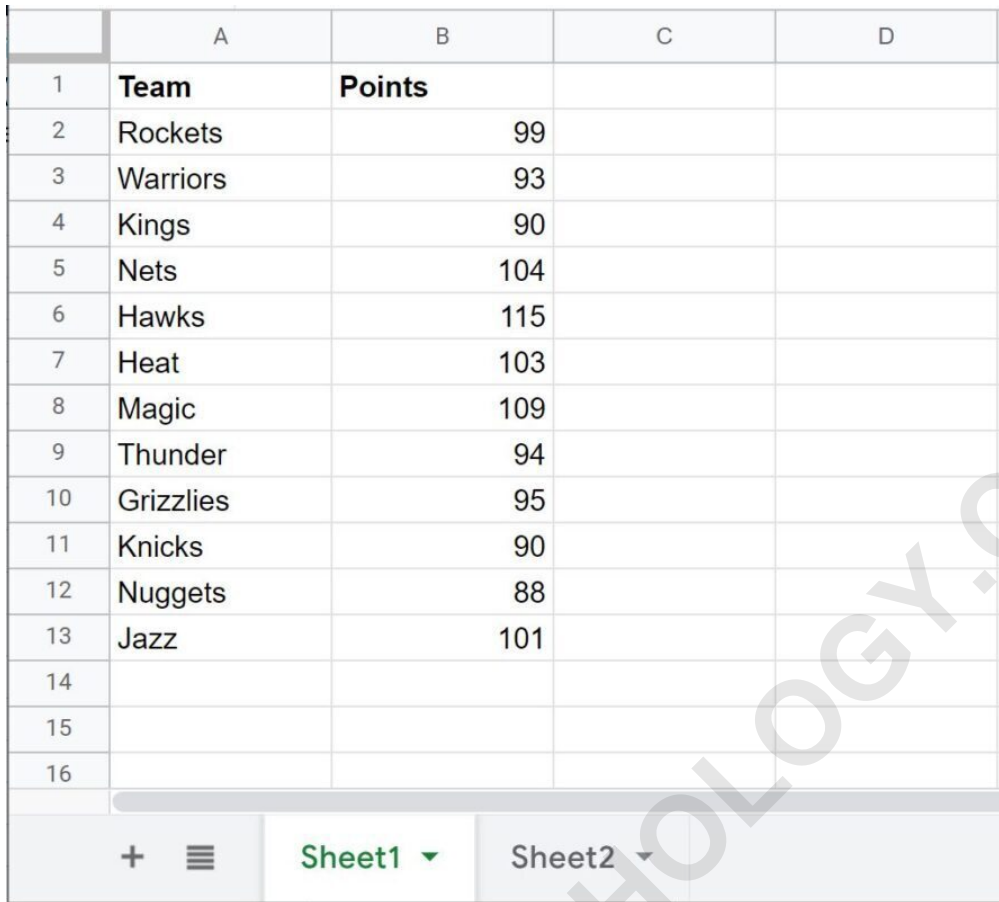
The initial step in this procedure requires setting up and verifying the source data that you wish to transfer dynamically. For the purpose of this practical guide, we will assume this source information resides in a sheet designated as **Sheet1**. Ensuring that your source data is organized into clean, consistent columns and rows is fundamental to a successful autofill operation. Any disorganization in the source data structure, such as inconsistent row alignment or misplaced headers, will inevitably lead to complex or erroneous results upon transfer to the destination sheet.

For our example, let us establish a simple dataset within **Sheet1** that includes categorical labels and corresponding numerical values. This structured approach allows us to clearly observe the cross-sheet transfer functionality in action and verify the integrity of the data linkage. The specific values used here are arbitrary but serve to demonstrate the mechanics of the transfer.

Step 1: Enter Data in First Sheet

First, let us enter the following structured data into **Sheet1** in Google Sheets. We are defining a set of records, where Column A represents an Item ID used for identification, and Column B represents the numerical Points associated with that item. These 'Points' are the values we intend to pull automatically into our second sheet.

	A	B	C	D
1	Team	Points		
2	Rockets	99		
3	Warriors	93		
4	Kings	90		
5	Nets	104		
6	Hawks	115		
7	Heat	103		
8	Magic	109		
9	Thunder	94		
10	Grizzlies	95		
11	Knicks	90		
12	Nuggets	88		
13	Jazz	101		
14				
15				
16				



It is important to verify that the data range is contiguous and complete before moving to the destination sheet. In this illustration, the valuable information we aim to transfer automatically resides specifically in column **B**, starting from row **2**. This column of 'Points' will serve as our primary source array for the Autofill operation into the subsequent sheet, **Sheet2**.

Initiating the Autofill Process: Linking Sheets

Once the source sheet is properly populated and structured, the next phase involves preparing the destination environment. For this guide, we will use **Sheet2** as the receiving location for the mirrored data. It is generally considered best practice to establish corresponding headers in the destination sheet first, ensuring that the context of the transferred values is immediately clear to any user viewing the spreadsheet and maintaining readability.

Step 2: Autofill Data in Second Sheet

Now suppose we have another sheet titled **Sheet2** that contains related data, perhaps unique identifiers or timestamps, but currently lacks the 'Points' values from **Sheet1**. The objective is to populate a new column in **Sheet2** with the corresponding 'Points' values from **Sheet1**, ensuring

that the row-by-row alignment between the sheets is perfectly maintained.

	A	B	C	D
1	Team	Conference		
2	Rockets	Western		
3	Warriors	Western		
4	Kings	Western		
5	Nets	Eastern		
6	Hawks	Eastern		
7	Heat	Eastern		
8	Magic	Eastern		
9	Thunder	Western		
10	Grizzlies	Western		
11	Knicks	Eastern		
12	Nuggets	Western		
13	Jazz	Western		
14				
15				
16				

Suppose we would like to Autofill the values from the Points column (Column B) in **Sheet1** into a new Points column (Column C) in **Sheet2**, beginning in cell **C2**. This essential operation allows us to avoid manual data entry and establish a robust, dynamic link between the two data repositories. The first step involves setting up the initial explicit cross-sheet cell reference formula in the starting cell.

The Mechanics of the Direct Cell Reference Formula

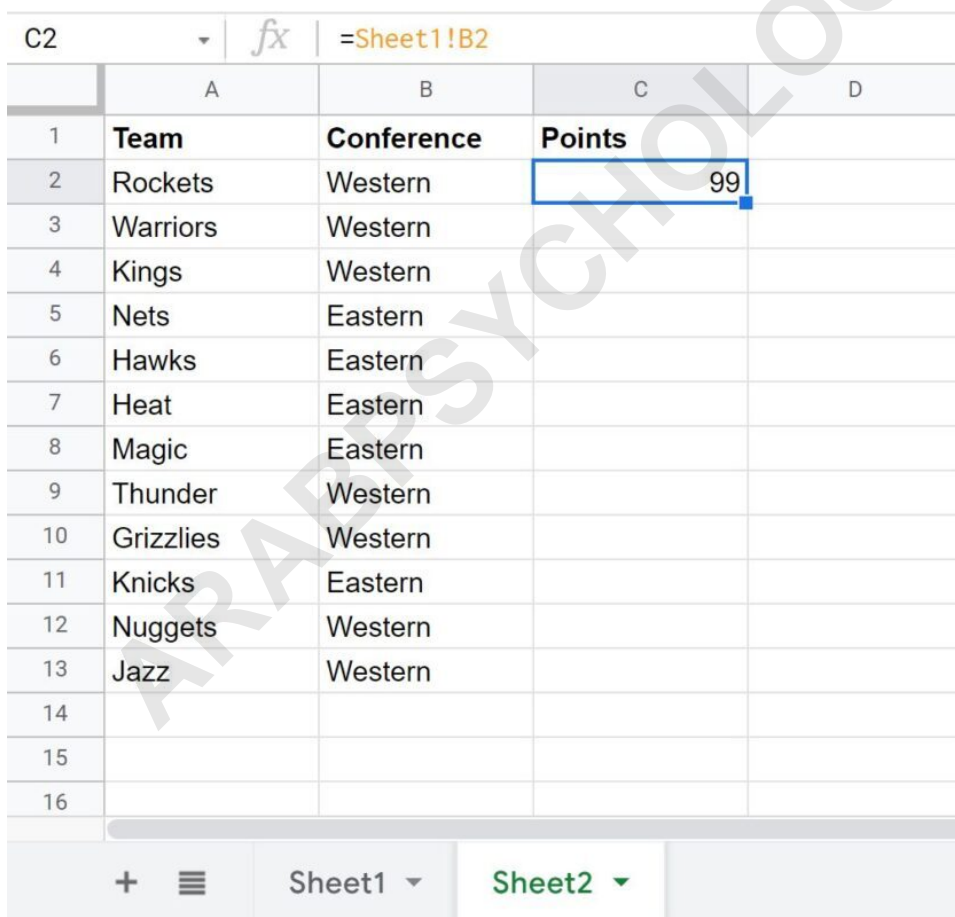
To initiate the linking process, we must enter a simple but powerful formula into the target cell (**C2** of **Sheet2**). This formula does not perform a calculation; rather, it instructs Google Sheets to retrieve and display the exact value currently held in the specified source cell, thereby creating a live link.

To do so, we can type the following formula in cell **C2** of **Sheet2**. Note the specific syntax required for cross-sheet referencing, which includes the sheet name (**Sheet1**), the necessary separator (!), and the target cell reference (**B2**):

=Sheet1!B2

This operation immediately establishes the link. Upon hitting Enter, the value contained in **Sheet1!B2** (which is 400 in our illustrative example) will instantly appear in **Sheet2!C2**. This connection is not merely a static copy; it is a live synchronization. If the value in **Sheet1!B2** is later modified--for example, if the score changes--the value displayed in **Sheet2!C2** will reflect that change instantaneously, ensuring that the destination sheet is always drawing the most current source data. This ability to maintain dynamic linkage is the primary advantage of using a direct reference formula over a simple paste operation.

This initial setup step is critical, as it validates both the formula syntax and the connection path. Once validated, this mechanism will automatically populate cell **C2** in **Sheet2** with the value from cell **B2** in **Sheet1**, setting up the foundation for the subsequent Autofill action. The resulting appearance of the sheet confirms successful formula implementation:



The screenshot shows a Google Sheet interface. At the top, the formula bar displays the formula `=Sheet1!B2` in orange text. Below the formula bar is a table with columns A, B, C, and D, and rows 1 through 16. The table contains the following data:

	A	B	C	D
1	Team	Conference	Points	
2	Rockets	Western	99	
3	Warriors	Western		
4	Kings	Western		
5	Nets	Eastern		
6	Hawks	Eastern		
7	Heat	Eastern		
8	Magic	Eastern		
9	Thunder	Western		
10	Grizzlies	Western		
11	Knicks	Eastern		
12	Nuggets	Western		
13	Jazz	Western		
14				
15				
16				

At the bottom of the sheet, the navigation bar shows a plus sign, a menu icon, and two sheet tabs: 'Sheet1' and 'Sheet2' (which is currently selected and highlighted in green).

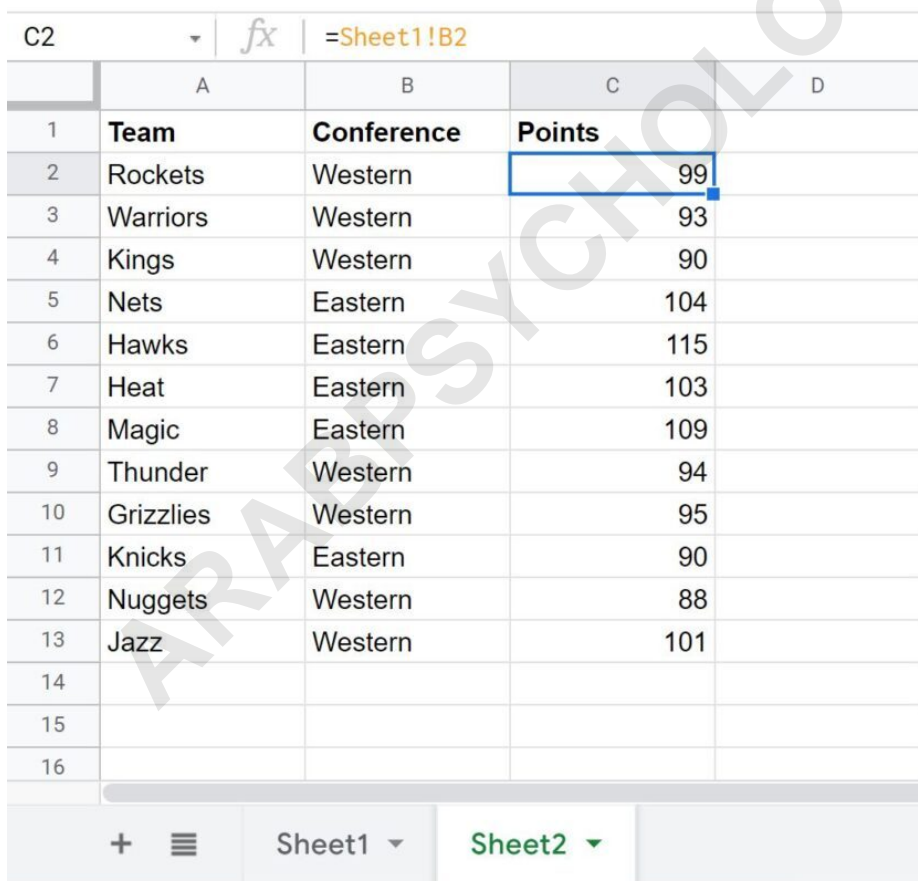
Executing the Autofill Command: Duplicating the Reference

With the template formula correctly placed in the first cell of the destination column (**C2**), we can

now utilize the power of Autofill to replicate this formula down the entire required range. The Autofill process relies heavily on the concept of relative cell reference adjustment. Since we did not use dollar signs (\$) to lock the row or column in our original formula (**=Sheet1!B2**), Google Sheets understands that we want the row number to increment sequentially as we move down the column.

To autofill the rest of the values in column C, we must utilize the fill handle. This handle is accessed by hovering your mouse cursor over the bottom right-hand corner of cell **C2** until a tiny cross or "+" symbol appears. This visual cue is crucial, as it indicates that the Autofill tool is ready for activation. Once the cross appears, click and hold the mouse button down.

Then, click and drag the fill handle down to all of the remaining cells in column C that correspond to the rows populated in **Sheet1**. As you drag the handle, Google Sheets executes the relative adjustment logic: cell C3 receives the formula **=Sheet1!B3**, cell C4 receives **=Sheet1!B4**, and so on. This mechanical operation ensures that the entire column is populated swiftly and accurately with the corresponding data from the source sheet, maintaining row integrity across the workbook.



	A	B	C	D
1	Team	Conference	Points	
2	Rockets	Western	99	
3	Warriors	Western	93	
4	Kings	Western	90	
5	Nets	Eastern	104	
6	Hawks	Eastern	115	
7	Heat	Eastern	103	
8	Magic	Eastern	109	
9	Thunder	Western	94	
10	Grizzlies	Western	95	
11	Knicks	Eastern	90	
12	Nuggets	Western	88	
13	Jazz	Western	101	
14				
15				
16				

Validation and Benefits of Dynamic Data Transfer

Upon releasing the mouse button after the drag operation, the destination column (Column C in

Sheet2) will immediately display all the values from the source column (Column B in **Sheet1**). This confirms that the entire range has been successfully transferred via the Autofill mechanism. Notice that all of the values from the Points column in **Sheet1** have been autofilled into **Sheet2**, establishing a complete, linked dataset ready for further analysis or reporting.

The primary advantage of this method over manual entry or static copying is the creation of a dynamic pipeline. If the underlying data in **Sheet1** changes--for instance, if the points for Item 5 are updated from 500 to 550--that change is instantly reflected in **Sheet2** without any user intervention. This dynamic linking capability is indispensable for dashboards, summary reports, and complex models where maintaining real-time consistency between various components of a spreadsheet workbook is paramount to decision-making accuracy.

Advanced Applications and Considerations

While the direct cell reference is straightforward for sequential mirroring, advanced users might need to consider absolute referencing techniques. If, instead of pulling a sequence of values that change row by row, you needed every single cell in your destination column to reference the exact same single cell (e.g., a fixed tax rate defined globally in **Sheet1!A1**), you would need to employ an absolute cell reference by adding dollar signs: **=Sheet1!\$A\$1**. When Autofill is applied to an absolute reference, the reference remains locked, explicitly preventing the row or column index from changing during the drag operation.

Furthermore, for very large data transfers or when combining multiple columns into a single operation, users might opt for array formulas or functions like **ARRAYFORMULA** or **IMPORTRANGE**. While the simple direct reference followed by Autofill is perfect for mirroring a single, contiguous column within the same workbook, **ARRAYFORMULA(Sheet1!B2:B)** can achieve the same result with a single entry in C2, automatically spilling the results down the column without requiring the manual drag operation. However, this approach is suitable only when no other data exists in the destination cells C3, C4, etc., as the array will reserve and potentially overwrite any content in that range.

Troubleshooting Common Autofill Issues

When attempting to use Autofill for cross-sheet referencing, several common issues may arise that interrupt the transfer process. One frequent problem is the appearance of **#REF!** errors across the destination column. This usually indicates that the sheet name in the formula is misspelled, or the targeted sheet does not exist in the workbook. Always double-check the exact spelling, especially paying attention to any spaces or special characters in the sheet title, which require single quotes around the name.

Another potential issue is inconsistent data transfer, where the Autofill skips cells or pulls incorrect

values relative to the adjacent data. If this occurs, revisit the initial formula in cell C2. Verify that you are using relative referencing (no dollar signs) if you intended for the row number to increment. If the source sheet has blank rows or merged cells within the referenced range, the transfer will mirror these inconsistencies, often leading to unexpected gaps or misalignment in the destination column. Maintaining clean, standardized source data prevents the vast majority of these errors and ensures a smooth, predictable autofill outcome.

ARABPSYCHOLOGY.COM