How to Easily Average Filtered Data in Excel

Authored by

stats writer

November 30, 2025

Analyzing specific subsets of data is a fundamental task in Microsoft Excel. While filtering capabilities allow users to quickly isolate relevant records, calculating statistics like the average only for the visible rows presents a unique challenge. Simply applying the standard AVERAGE function after applying a Filter often yields incorrect results because it includes hidden rows in its calculation. This comprehensive guide details the precise technique required to calculate the average exclusively for data visible after filtering, ensuring accurate data analysis and reporting.

The common initial approach involves filtering your desired data set, selecting the relevant cell range, and then attempting to use the quick average calculation available in the Home tab or status bar. While this method provides a quick overview, it is unreliable for creating persistent, formula-driven averages of filtered data. Consider an example where you need to calculate the average sales for a particular product across various regions. You would first filter your comprehensive table to display only that specific product. If you then apply the standard **AVERAGE()** function to the sales column, Excel will erroneously include the sales figures from the rows that were hidden by the filter, skewing your results significantly.

To overcome this limitation and accurately average filtered rows in Excel, we must utilize a specialized function designed specifically to respect the visibility status of cells: the SUBTOTAL function. This powerful function allows you to perform various calculations--including averaging, summing, counting, and more--only on the cells that are currently visible within a range. Understanding its structure and corresponding function codes is crucial for mastering dynamic data analysis in Excel.

Understanding the Power of the SUBTOTAL Function

The most reliable and easiest method to calculate the average of a filtered range in Microsoft Excel involves deploying the **SUBTOTAL** function. This function differs fundamentally from standard aggregation formulas because it accepts a function number argument, which dictates the type of calculation to be performed, and determines whether or not to include manually hidden rows and rows hidden by a filter.

For the specific purpose of averaging data that has been filtered, the syntax is straightforward yet powerful:

The easiest way to take the average of a filtered range in Excel is to use the following syntax:

SUBTOTAL(101, A1:A10)

Note that the value 101 is the function code for taking the average of a filtered range of rows, specifically ignoring values in rows that have been hidden by a filter.

Deciphering SUBTOTAL Function Codes

The key to correctly implementing the SUBTOTAL function lies in selecting the appropriate Function Number (the first argument). Excel assigns specific numerical codes for various aggregation types. Importantly, these codes exist in two distinct ranges: codes 1 through 11, and codes 101 through 111.

When using codes 1 through 11 (e.g., 1 for AVERAGE, 9 for SUM), Excel will calculate results based on rows hidden by a filter, but it will exclude rows that have been manually hidden by the user. While this is helpful, it fails for automatically filtered tables created using the Data tab's filter command.

The advanced range, codes 101 through 111, is designed precisely for filtered data analysis. These codes ensure that Excel **ignores all rows hidden by a filter** (using the Data > Filter command) **and** manually hidden rows.

For calculating the average of only visible cells, we must use the code associated with the AVERAGE function within the 100-series range. Since the standard code for AVERAGE is 1, the corresponding code for visible-only averaging is 101.

The most relevant codes for visible-only calculations are:

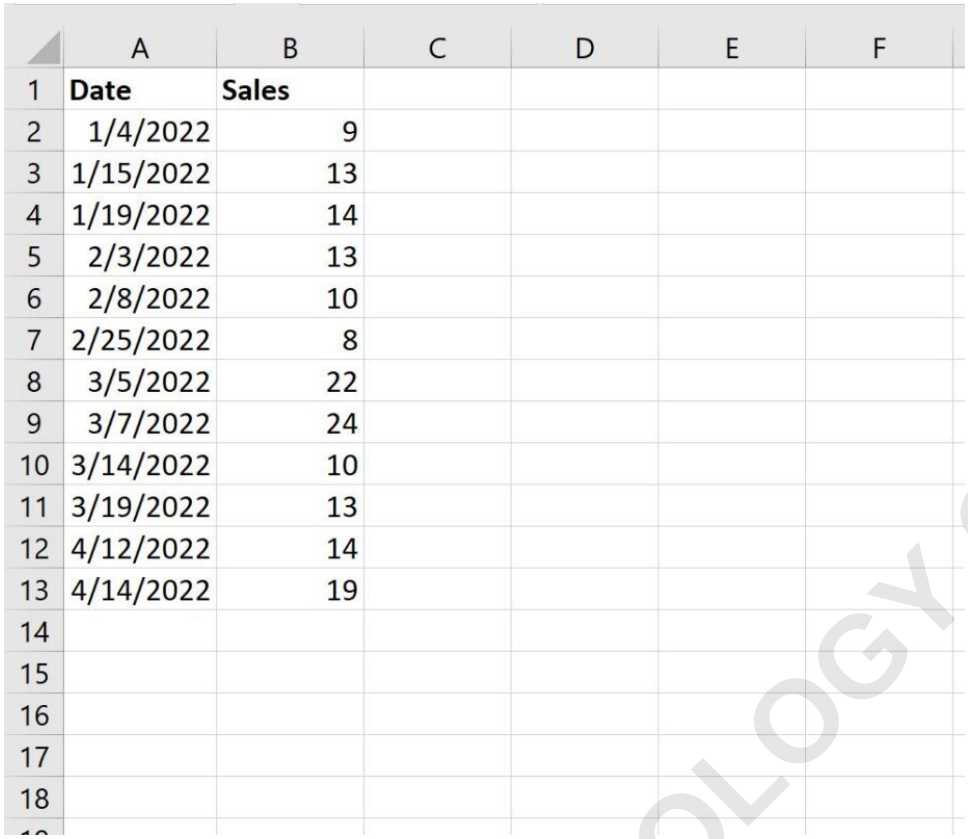
101: Calculates the average of the visible cells.
109: Calculates the sum of the visible cells.
103: Counts the number of non-blank values in the visible cells (COUNTA).
102: Counts the number of numerical values in the visible cells (COUNT).

Setting Up the Practical Example

The following detailed example demonstrates how to implement the **SUBTOTAL** function effectively when working with dynamically filtered data.

Imagine we are tasked with analyzing the sales performance of a company over several months. We have a simple dataset tracking the transaction date and the number of sales made on that day. Before we can calculate the average performance for a specific time frame, we must first establish the structure of our data.

Suppose we have the following dataset that shows the number of sales made during various days by a company. This dataset is organized into two columns: **Date** and **Sales**. For this tutorial, we will assume this data occupies the range A1:B13.



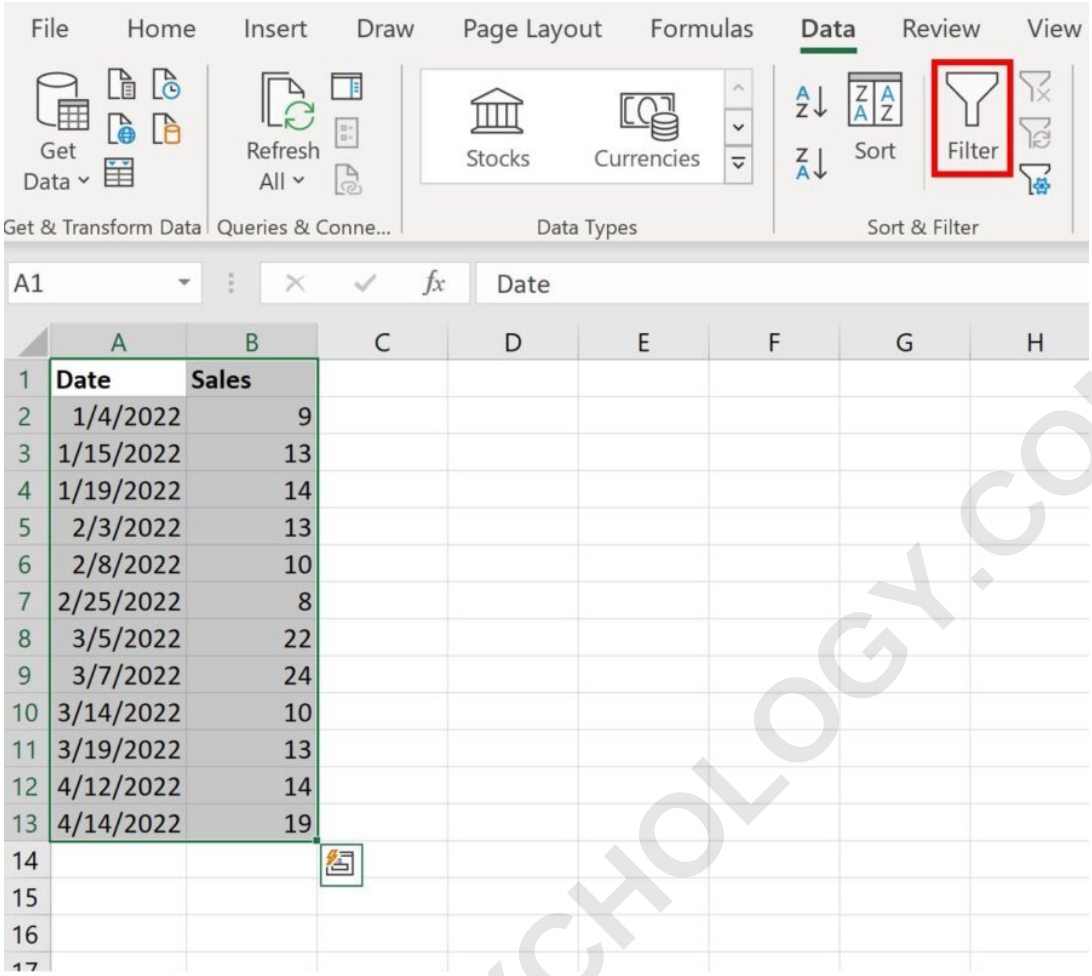
	A	B	C	D	E	F
1	**Date**	**Sales**				
2	1/4/2022	9				
3	1/15/2022	13				
4	1/19/2022	14				
5	2/3/2022	13				
6	2/8/2022	10				
7	2/25/2022	8				
8	3/5/2022	22				
9	3/7/2022	24				
10	3/14/2022	10				
11	3/19/2022	13				
12	4/12/2022	14				
13	4/14/2022	19				
14						
15						
16						
17						
18						

The initial step for any dynamic analysis is ensuring your data is ready for filtering. While Excel can apply filters to simple ranges, converting your data into an official Excel Table (using Ctrl+T) is often recommended for better integration with calculated fields and dynamic range referencing, although a standard range filter works perfectly for this specific example using the **SUBTOTAL** function.

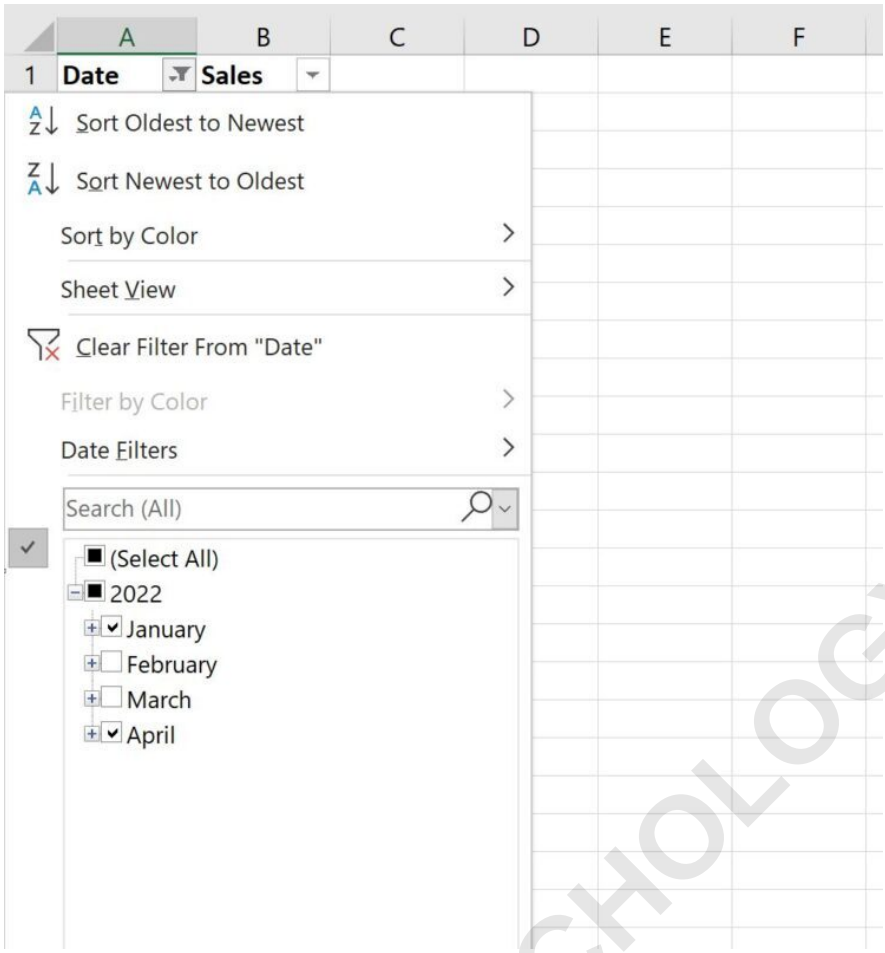
Applying Filters to Isolate Specific Data

Our goal is to determine the average daily sales exclusively during the months of January and April. To achieve this, we must first apply the appropriate criteria to the dataset using Excel's built-in filtering tools.

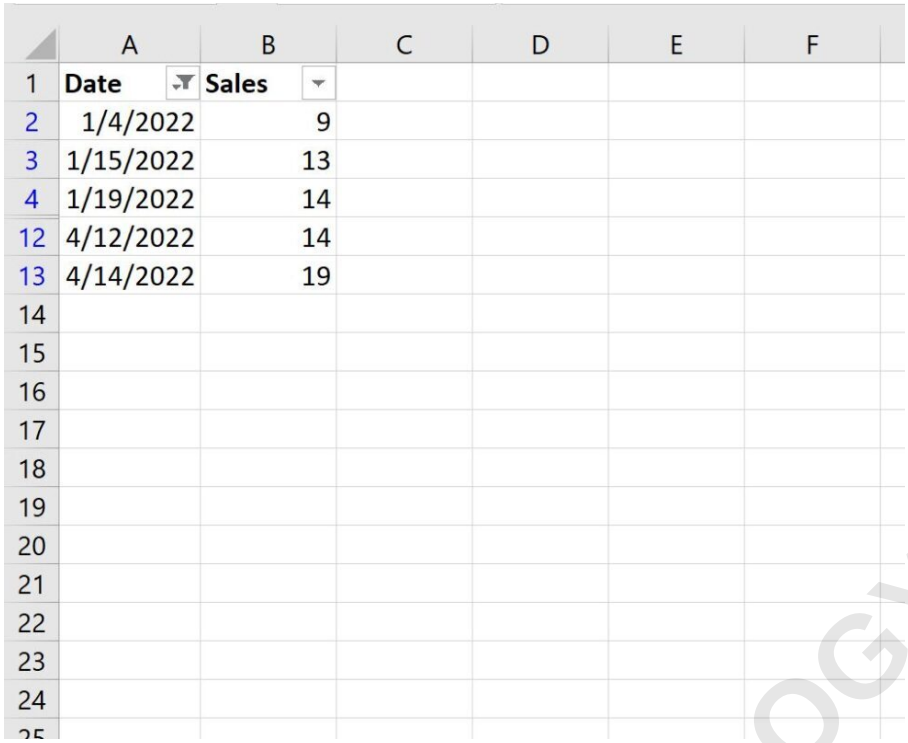
To begin the filtering process, highlight the entire cell range containing your data, specifically the range **A1:B13** in this demonstration. Once the range is selected, navigate to the **Data** tab located along the top ribbon interface. Within the Data tab, locate and click the **Filter** button. This action will instantly place dropdown arrows next to the header cells (A1 and B1), indicating that filtering capabilities are now active for the table.



After activating the filters, the next step is defining the criteria. Since we are interested only in sales figures from January and April, click the dropdown arrow next to the **Date** header. This opens the filter menu. In the date filtering options, expand the list of months, and ensure that only the checkboxes next to **January** and **April** are checked. All other months must be unchecked. Once these selections are confirmed, click the **OK** button to apply the filtering rule.



Upon successful execution of the Filter command, the dataset will automatically refresh. The visual output will now be constrained to only show the rows where the transaction dates fall within January or April. Notice that the row numbering on the left side of the sheet will skip the hidden rows, confirming the data is correctly filtered:

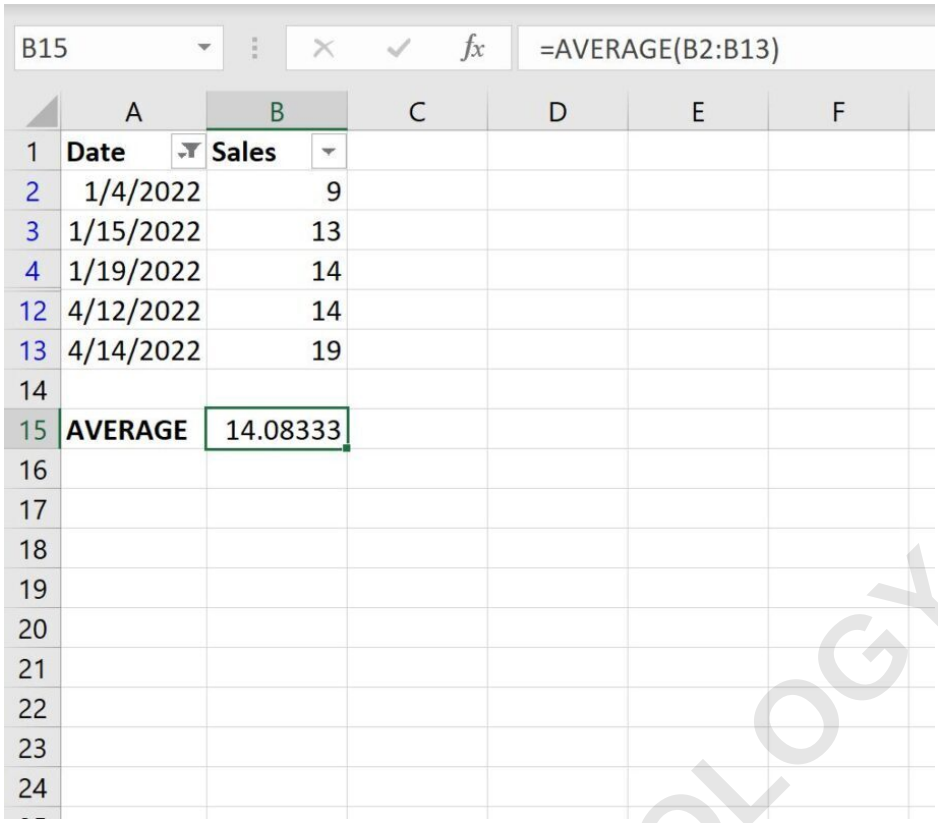


	A	B	C	D	E	F
1	Date	Sales				
2	1/4/2022	9				
3	1/15/2022	13				
4	1/19/2022	14				
12	4/12/2022	14				
13	4/14/2022	19				
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						

The Pitfall of Using the Standard AVERAGE Function

With the data successfully filtered, it is crucial to understand why standard aggregation functions fail in this scenario. If we attempt to use the conventional **AVERAGE()** function, such as `=AVERAGE(B2:B13)`, to find the average value in the Sales column, the result returned will incorrectly reflect the average of all of the original values in that range, including those in the rows hidden by the filter.

This behavior is inherent to most standard statistical functions in Excel. They operate on the underlying range reference regardless of the applied visual filter. In our example, even though rows corresponding to February, March, and other months are hidden, the **AVERAGE()** function still processes the sales figures within those hidden cells. This leads to a flawed analytical result, as the calculated average does not represent the subset of data we intended to analyze (January and April sales).



As shown in the image above, the standard **AVERAGE** function calculates a value that incorporates every data point from the original twelve entries, providing an average that is not specific to the January and April filter criteria. This demonstrates the necessity of using a specialized function like SUBTOTAL for accurate filtered reporting.

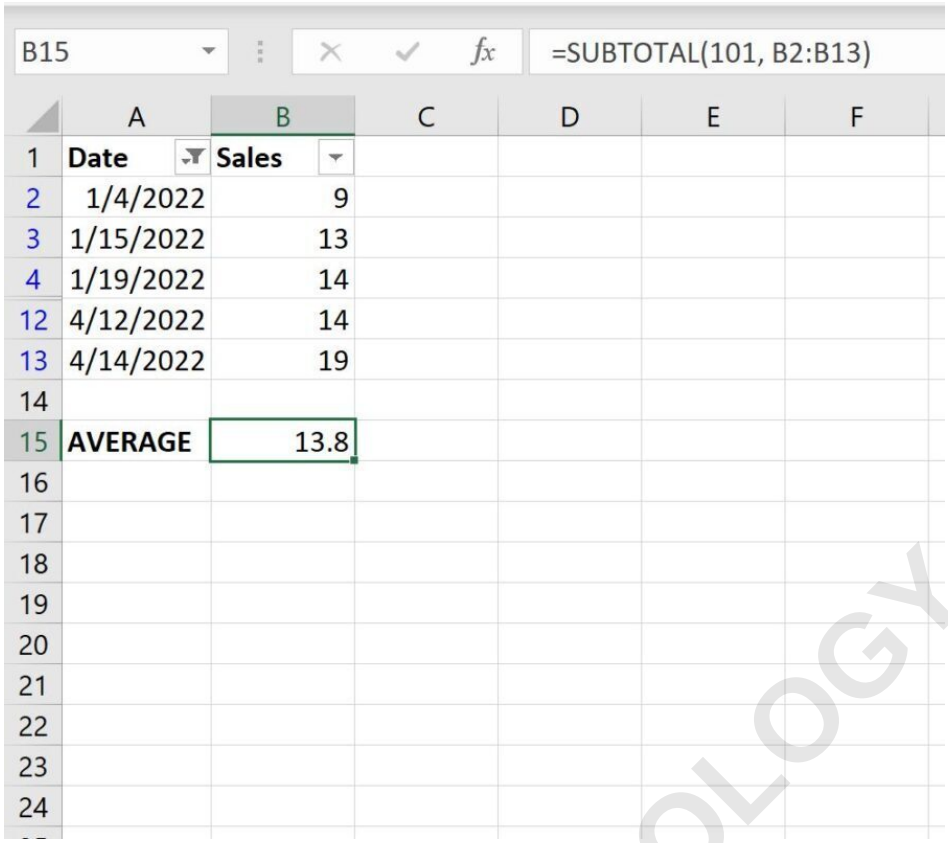
Implementing SUBTOTAL(101) for Accurate Averaging

To achieve the correct average--that is, the average calculated solely on the sales figures visible after filtering--we must replace the **AVERAGE()** function with the appropriate SUBTOTAL function syntax.

In an empty cell adjacent to or below your dataset, input the following formula, referencing the range containing the Sales figures (B2 through B13, excluding the header):

```
=SUBTOTAL(101, B2:B13)
```

Remember, the **101** code is the defining feature here. It instructs Excel to perform an Average calculation (1) while explicitly ignoring any rows that have been hidden by a Filter (the 100 series). This ensures that only the sales data for January and April are included in the calculation.



As observed in the result generated by the **SUBTOTAL(101)** function, the calculated average is now significantly different from the result obtained using the standard **AVERAGE()** function. This calculated value, 13.8, accurately represents the average daily sales strictly for the filtered months of January and April, demonstrating the function's ability to handle dynamic data visibility.

Verification and Conclusion

To ensure complete confidence in the result, it is beneficial to manually verify the calculation based only on the visible rows. By summing the visible sales figures and dividing by the count of visible sales figures, we can confirm the accuracy of the **SUBTOTAL(101)** output.

Based on the filtered view, the visible sales figures are: 9, 13, 14, 14, and 19.

Sum of Visible Sales: 9 + 13 + 14 + 14 + 19 = 69
Count of Visible Sales Rows: 5
Average of Visible Sales: 69 / 5 = **13.8**

We can manually verify this by taking the average of the visible rows:

Average of Sales in visible rows: (9+13+14+14+19) / 5 = **13.8**.

This manual verification confirms that the **SUBTOTAL(101)** function correctly isolates and averages only the data displayed after the filter is applied. It is essential to integrate this function into your workflow whenever statistical analysis is required on dynamic, filtered subsets of data within Excel. Relying on **SUBTOTAL(101)** ensures your reports are accurate, irrespective of the filtering criteria used.

Advanced Considerations: Using SUBTOTAL with Tables

While this example used a standard range, applying filters to a defined Excel Table offers additional benefits. When using Tables, the SUBTOTAL function is often automatically employed when using the Table's 'Total Row' feature. If you enable the Total Row at the bottom of an Excel Table, the default calculation (usually SUM) can be instantly changed to AVERAGE. Excel automatically uses the 100-series function codes (like 101 for Average) in this Total Row, guaranteeing that the result updates dynamically and correctly as you modify your filters. This further highlights the utility of the **SUBTOTAL** framework for handling visible data subsets efficiently.