

Select Every Nth Row in Excel (With Example)

Authored by
Mohammed looti

October 31, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Select Every Nth Row in Excel (With Example)*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=6630>

When managing and analyzing extensive [datasets](#) in [Excel](#), analysts often face the requirement of extracting or highlighting specific rows based on their precise positional interval--a process commonly referred to as selecting every Nth row. This technique is fundamental for crucial tasks such as systematic data sampling, generating concise summary reports, or isolating specific trends within a large data structure. Fortunately, [Excel](#) offers a highly versatile and dynamic formula solution to achieve this complex selection with remarkable efficiency. This expert guide will walk you through the necessary steps, explaining the core functions involved, and providing detailed, practical examples to ensure mastery of this extraction method.

The underlying mechanism for this powerful operation relies on leveraging [Excel](#)'s navigational functions. Primarily, we utilize the dynamic capabilities of the [OFFSET function](#) in precise concert with the [ROW function](#). This potent combination generates a formula that automatically calculates and adapts its position as it is copied down a column, effectively enabling the selection of values from rows that occur at perfectly regular intervals.

Establishing the Core Formula for Nth Row Selection

To systematically select data from every **Nth** row within your [Excel](#) spreadsheet, the following fundamental formula provides the necessary structure and logic:

=OFFSET(\$A\$1,(ROW()-1)*n,0)

This formula is meticulously engineered to dynamically retrieve data by calculating an exact row offset based on a user-defined interval, denoted by the variable **n**. By simply substituting **n** with any positive integer, you gain complete and immediate control over the specific pattern of row selection. For example, if your analytical requirement is to select data from every **3rd** row in the source data, you would substitute the variable **n** with the integer **3**, resulting in the following targeted formula:

=OFFSET(\$A\$1,(ROW()-1)*3,0)

Before proceeding to the practical examples, it is essential to delve deeper into the construction of this formula, analyzing each component to fully grasp how it achieves such precise, periodic selection across the sheet.

Deconstructing the Formula: Understanding OFFSET, ROW, and References

Successful implementation and future customization of this extraction method hinge upon a clear understanding of each element within the formula. The expression **=OFFSET(\$A\$1, (ROW() - 1) * n, 0)** is a powerful synthesis of multiple core Excel functionalities

designed for dynamic data navigation.

The primary engine of the formula is the [OFFSET function](#). This function is designed to return a reference to a range that is a specified number of rows and columns away from a starting point. Its formal syntax is: `OFFSET(reference, rows, cols, , )`.

reference (\$A\$1): This argument defines the fixed anchor point from which all subsequent offsets are calculated. Utilizing \$A\$1 ensures that it is an [absolute reference](#). The absolute nature of the reference (indicated by the dollar signs) is critical because it prevents the starting point from shifting when the formula is copied to other [cells](#). This guarantees that the "Nth" interval is always counted correctly from the very beginning of the source data.

rows ((ROW()-1)*n): This is the dynamic core of the calculation. This argument specifies the number of rows the [OFFSET function](#) must move down (or up) from the reference. The sub-expression `(ROW()-1)*n` calculates the precise jump required for the Nth selection.

cols (0): This argument dictates the column shift relative to the reference. A value of 0 is used here to ensure the formula stays within the same column as the starting reference (Column A in this example). If your data were in column F and you used \$F\$1 as the reference, 0 would still be correct for the column argument.

and : These are optional parameters that define the size of the range returned by OFFSET. Since our goal is to extract a single [cell](#) value at a time, these arguments are typically omitted from the formula.

The Crucial Role of the ROW Function and Indexing

The [ROW function](#) is the key to creating the sequential sequence necessary for the interval selection. When utilized without any arguments (simply `ROW()`), it returns the exact row number of the [cell](#) where the formula itself resides.

ROW(): If you enter this function into [cell](#) D1, it returns 1. If you copy it to D5, it returns 5. This increasing value is essential for iteration.

(ROW()-1): This offset adjustment is the functional brilliance of the formula. When the formula is first entered into the output column (e.g., C1), `ROW()` returns 1. Subtracting 1 results in 0. This calculation forces the first offset to be $0 * n = 0$, meaning the [OFFSET function](#) returns the value from the reference [cell](#) itself (A1). As the formula is copied down, `ROW()` increases incrementally, generating the required sequence of multipliers: 0, 1, 2, 3, and so on.

The variable **n**, which represents the interval, is simply the periodic jump required. By changing this single numerical value, you redefine the entire selection pattern. If **n = 4**, the formula targets rows 1, 5, 9, 13, etc. This singular point of control allows this formula to be universally adaptable to any systematic data extraction need.

Practical Application: Selecting Every 3rd Row in a Dataset

To solidify your understanding, let us apply this formula using a concrete example. Imagine you possess a list of raw data residing in column A, and your objective is to efficiently extract every **third** row from this existing list into a new, separate column for reporting purposes.

We will use the following sample [dataset](#) displayed in column A as our source data:

	A	B	C	D	E	F
1	4					
2	8					
3	12					
4	14					
5	19					
6	22					
7	24					
8	28					
9	29					
10	31					
11	34					
12	35					
13	49					
14	13					
15	6					
16	7					
17	12					
18	10					
19	11					
20	6					
21						

Since we are targeting every **third** row, the variable **n** is set to 3. The precise formula required is:

=OFFSET(\$A\$1,(ROW()-1)*3,0)

Execute the extraction using the following steps:

Enter the formula =OFFSET(\$A\$1, (ROW() - 1) * 3, 0) into [cell C1](#). This [cell](#) will serve as the starting point for your newly extracted list.

Press **Enter**. The value contained within [cell A1](#) should immediately appear in C1, validating the starting position.

Select [cell C1](#) again. Locate the small green square, known as the fill handle, positioned at the bottom-right corner of the [cell](#) boundary.

Click and drag the fill handle downwards. This action copies the formula iteratively to the remaining cells in column C, covering the entire projected range of your original data.

The dynamic process of entering the formula and subsequently copying it down the output column will appear as demonstrated below:

	A	B	C	D	E	F	G
1	4		4				
2	8		14				
3	12		24				
4	14		31				
5	19		49				
6	22		7				
7	24		11				
8	28						
9	29						
10	31						
11	34						
12	35						
13	49						
14	13						
15	6						
16	7						
17	12						
18	10						
19	11						
20	6						
21							
22							

Upon completion of the drag-and-drop operation, you will successfully observe that only every **third** row has been systematically selected and accurately displayed in column C, showcasing the precision and reliability of this interval selection methodology:

	A	B	C	D	E	F	G
1	4		4				
2	8		14				
3	12		24				
4	14		31				
5	19		49				
6	22		7				
7	24		11				
8	28						
9	29						
10	31						
11	34						
12	35						
13	49						
14	13						
15	6						
16	7						
17	12						
18	10						
19	11						
20	6						
21							
22							

Extending the Logic: Selecting Every 5th Row

One of the greatest benefits of this formula structure is its inherent adaptability. Should your data extraction requirements evolve, for instance, demanding the selection of every **fifth** row instead of the third, the required modification is incredibly minor. You are only required to adjust the numerical value of the variable **n**.

To successfully select every **fifth** row from the source data, the updated formula structure becomes:

=OFFSET(\$A\$1,(ROW()-1)*5,0)

Following the exact same procedure as the previous example, you would input this revised formula into [cell C1](#) and utilize the fill handle to copy it down the column.

The visualization below illustrates the application of this modified formula targeting the fifth interval:

=OFFSET(\$A\$1,(ROW()-1)*5,0)							
	A	B	C	D	E	F	G
1	4		4				
2	8		22				
3	12		34				
4	14		7				
5	19						
6	22						
7	24						
8	28						
9	29						
10	31						
11	34						
12	35						
13	49						
14	13						
15	6						
16	7						
17	12						
18	10						
19	11						
20	6						
21							
22							
23							

The final result clearly confirms that every **fifth** row has been accurately and reliably extracted from the initial list of values, underscoring the versatility and scalable nature of this indexing technique for various data sampling tasks:

	A	B	C	D	E	F	G
1	4		4				
2	8		22				
3	12		34				
4	14		7				
5	19						
6	22						
7	24						
8	28						
9	29						
10	31						
11	34						
12	35						
13	49						
14	13						
15	6						
16	7						
17	12						
18	10						
19	11						
20	6						
21							
22							
23							

Benefits and Performance Considerations

Employing the [OFFSET function](#) to select every Nth row in [Excel](#) provides substantial benefits, especially when dealing with systematic sampling or segmentation of large [datasets](#). This method completely automates what would otherwise be a labor-intensive and error-prone manual process. Crucially, the dynamic link created by the [OFFSET function](#) ensures that the extracted data remains connected to the source; if the original data in column A is updated, the extracted list in column C will automatically recalculate and reflect those changes.

However, it is vital for advanced users to be aware of a specific technical detail: the [OFFSET function](#) is classified as a volatile function in Excel. This means that it is forced to recalculate every single time any change, no matter how minor, occurs anywhere within the entire workbook, not just the range it affects. For standard or moderately sized [datasets](#), this performance impact is negligible. If, however, you are working with extremely massive datasets involving hundreds of thousands of rows and numerous volatile formulas, alternative, non-volatile methods (such as advanced filtering or Power Query) might be necessary to maintain optimal spreadsheet

responsiveness. For most common business and analytical applications, however, the simplicity, clarity, and efficiency of this OFFSET/ROW combination make it the ideal solution for targeted row extraction.

Additional Resources for Excel Proficiency

To further enhance your data manipulation skills, the following resources explain how to perform other essential data operations in [Excel](#):