How do I use the SEARCH and SEARCHB functions in Excel?

Authored by

stats writer

July 1, 2024

The SEARCH and SEARCHB functions in Excel are useful tools for finding specific text within a larger string of text. The SEARCH function is used for searching for text in a case-insensitive manner, while the SEARCHB function is used for searching in a case-sensitive manner. To use these functions, the user must specify the text they are searching for and the cell or range where the text should be searched. The functions will then return the starting position of the first occurrence of the text within the specified range. The user can also include additional parameters, such as the starting position of the search and whether to search from left to right or right to left. These functions are particularly helpful for analyzing large sets of data and quickly finding specific information within them.

This article describes the formula syntax and usage of the **SEARCH** and **SEARCHB** functions in Microsoft Excel.

Description

The **SEARCH** and **SEARCHB** functions locate one text string within a second text string, and return the number of the starting position of the first text string from the first character of the second text string. For example, to find the position of the letter "n" in the word "printer", you can use the following function:

=SEARCH("n","printer")

This function returns **4** because "n" is the fourth character in the word "printer."

You can also search for words within other words. For example, the function

=SEARCH("base","database")

returns **5**, because the word "base" begins at the fifth character of the word "database". You can use the **SEARCH** and **SEARCHB** functions to determine the location of a character or text string within another text string, and then use the **MID** and **MIDB** functions to return the text, or use the **REPLACE** and **REPLACEB** functions to change the text. These functions are demonstrated in Example 1 in this article.

Important:

These functions may not be available in all languages.

SEARCHB counts 2 bytes per character only when a DBCS language is set as the default language. Otherwise SEARCHB behaves the same as SEARCH, counting 1 byte per character.

The languages that support DBCS include Japanese, Chinese (Simplified), Chinese (Traditional),

and Korean.

Syntax

SEARCH(find_text,within_text,)

SEARCHB(find_text,within_text,)

The SEARCH and SEARCHB functions have the following arguments:

find_text Required. The text that you want to find.

within_text Required. The text in which you want to search for the value of the **find_text** argument.

start_num Optional. The character number in the **within_text** argument at which you want to start searching.

Remark

The **SEARCH** and **SEARCHB** functions are not case sensitive. If you want to do a case sensitive search, you can use **FIND** and **FINDB**.

You can use the wildcard characters -- the question mark (**?**) and asterisk (*****) -- in the **find_text** argument. A question mark matches any single character; an asterisk matches any sequence of characters. If you want to find an actual question mark or asterisk, type a tilde (**~**) before the character.

If the value of **find_text** is not found, the #VALUE! error value is returned.

If the **start_num** argument is omitted, it is assumed to be 1.

If **start_num** is not greater than 0 (zero) or is greater than the length of the **within_text** argument, the #VALUE! error value is returned.

Use **start_num** to skip a specified number of characters. Using the **SEARCH** function as an example, suppose you are working with the text string "AYF0093.YoungMensApparel". To find the position of the first "Y" in the descriptive part of the text string, set **start_num** equal to 8 so that the serial number portion of the text (in this case, "AYF0093") is not searched. The **SEARCH** function starts the search operation at the eighth character position, finds the character that is specified in the **find_text** argument at the next position, and returns the number 9. The **SEARCH** function always returns the number of characters from the start of the **within_text** argument, counting the

characters you skip if the **start_num** argument is greater than 1.