#### SEARCH

[Syntax](Syntax.docx):

SEARCH ( string-1 , string-2 [ , start-pos ] )

Description: Performs a case-insensitive search for the first occurrence of string-1 in string-2, starting at character position start-pos within string-2. (SEARCH is intended for use with languages that use the single-byte character set (SBCS), whereas [SEARCHB](SEARCHB.docx) (§) is intended for use with languages that use the double-byte character set (DBCS).)

Arguments:

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| string-1 | [text](text.docx) | Designate the string to be searched for within the string designated by string-2. string-1 can contain the following wildcard characters: question mark (?) and asterisk (\*). A question mark matches any single character; an asterisk matches any sequence of characters. To search for an actual question mark or asterisk, that character shall be preceded by a tilde (~). |
| string-2 | [text](text.docx) |
| start-pos | number | The number of the start position within string-2 for which string-1 is to be searched. The start position of the first character is 1. If omitted, a position of 1 shall be assumed. start-pos shall be at least 0. |

Return Type and Value: number – The start position of the first occurrence of string-1 in string-2, starting at character position start-pos within string-2. If string-1 is an empty string, it shall always be found in any string-2 at position start-pos, or at position 1 if start-pos is omitted.

However, if

* string-1 is not found within string-2, #VALUE! is returned.
* start-pos designates a position outside string-2, #VALUE! is returned.

[Example:

SEARCH("de","abcdEF") results in 4
SEARCH("?c\*e","abcdEF") results in 2

end example]