# ibec_FormatFloat

Formats a floating point value.

### Syntax

function ibec_FormatFloat(const Format: string; Value: Extended): string;

### Description

ibec_FormatFloat formats the floating-point value given by Value using the format string given by Format. The following format specifiers are supported in the format string:

| Specifier | Represents |
|---|---|
| 0 | Digit place holder. If the value being formatted has a digit in the position where the 0 appears in the format string, then that digit is copied to the output string. Otherwise, a 0 is stored in that position in the output string. |
| # | Digit placeholder. If the value being formatted has a digit in the position where the #' appears in the format string, then that digit is copied to the output string. Otherwise, nothing is stored in that position in the output string. |
| . | Decimal point. The first '.' character in the format string determines the location of the decimal separator in the formatted value; any additional '.' characters are ignored. The actual character used as a the decimal separator in the output string is determined by the DecimalSeparator global variable. The default value of DecimalSeparator is specified in the *Number Format* of the *International* section in the *Windows Control Panel*. |
| , | Thousand separator. If the format string contains one or more ',' characters, the output will have thousand separators inserted between each group of three digits to the left of the decimal point. The placement and number of ',' characters in the format string does not affect the output, except to indicate that thousand separators are wanted. The actual character used as a the thousand separator in the output is determined by the ThousandSeparator global variable. The default value of ThousandSeparator is specified in the *Number Format* of the *International* section in the *Windows Control Panel*. |
| E+ | Scientific notation. If any of the strings E+, E-, e+, or e- are contained in the format string, the number is formatted using scientific notation. A group of up to four 0 characters can immediately follow the E+, E-, e+, or e- to determine the minimum number of digits in the exponent. The E+ and e+ formats cause a plus sign to be output for positive exponents and a minus sign to be output for negative exponents. The E- and e- formats output a sign character only for negative exponents. |
| 'xx'/"xx" | Characters enclosed in single or double quotes are output as-is, and do not affect formatting. |
| ; | Separates sections for positive, negative, and zero numbers in the format string. |

The locations of the leftmost 0 before the decimal point in the format string and the rightmost 0 after the decimal point in the format string determine the range of digits that are always present in the output string.

The number being formatted is always rounded to as many decimal places as there are digit placeholders (0 or #) to the right of the decimal point. If the format string contains no decimal point,

the value being formatted is rounded to the nearest whole number.

If the number being formatted has more digits to the left of the decimal separator than there are digit placeholders to the left of the '.' character in the format string, the extra digits are output before the first digit placeholder.

To allow different formats for positive, negative, and zero values, the format string can contain between one and three sections separated by semicolons.

- **One section:** The format string applies to all values.
- **Two sections:** The first section applies to positive values and zeros, and the second section applies to negative values.
- **Three sections:** The first section applies to positive values, the second applies to negative values, and the third applies to zeros.

If the section for negative values or the section for zero values is empty, that is if there is nothing between the semicolons that delimit the section, the section for positive values is used instead.

If the section for positive values is empty, or if the entire format string is empty, the value is formatted using general floating-point formatting with 15 significant digits, corresponding to a call to FloatToStrF with the ffGeneral format. General floating-point formatting is also used if the value has more than 18 digits to the left of the decimal point and the format string does not specify scientific notation.

From:
https://www.ibexpert.com/docu/ - **IBExpert**

Permanent link:
**https://www.ibexpert.com/docu/doku.php?id=05-ibexpert-ibeblock-functions:05-27-miscellaneous-functions:ibec_formatfloat**

Last update: **2023/07/05 18:56**