nfs()

Converts a positive `Number` into a `String` with an extra space in front.

`nfs()` converts positive numbers such as 123.45 into strings formatted with an extra space in front, as in ' 123.45'. Doing so can be helpful for aligning positive and negative numbers.

The first parameter, `num`, is the number to convert to a string. For example, calling `nfs(123.45)` returns the string `' 123.45'`.

The second parameter, `left`, is optional. If a number is passed, as in `nfs(123.45, 4)`, it sets the minimum number of digits to include to the left of the decimal place. If `left` is larger than the number of digits in `num`, then unused digits will be set to 0. For example, calling `nfs(123.45, 4)` returns the string `' 0123.45'`.

The third parameter, `right`, is also optional. If a number is passed, as in `nfs(123.45, 4, 1)`, it sets the minimum number of digits to include to the right of the decimal place. If `right` is smaller than the number of decimal places in `num`, then `num` will be rounded to the given number of decimal places. For example, calling `nfs(123.45, 4, 1)` returns the string `' 0123.5'`. If `right` is larger than the number of decimal places in `num`, then unused decimal places will be set to 0. For example, calling `nfs(123.45, 4, 3)` returns the string `' 0123.450'`.

Examples

```
-123
123
```

```
function setup() {
  createCanvas(100, 100);

  background(200);

  // Create number variables.
  let positive = 123;
  let negative = -123;

  // Convert the positive number to a formatted string.
  let formatted = nfs(positive);

  // Style the text.
  textAlign(CENTER, CENTER);
  textFont('Courier New');
  textSize(16);

  // Display the negative number and the formatted positive
number.
  text(negative, 50, 33);
  text(formatted, 50, 67);

  describe(
    'The numbers -123 and 123 written on separate lines. The
numbers align vertically. The text is in black on a gray
background.'
  );
}
```

```
function setup() {
  createCanvas(100, 100);

  background(200);

  // Create a number variable.
  let number = 123.45;

  // Convert the positive number to a formatted string.
  // Use four digits to the left of the decimal and
  // one digit to the right.
  let formatted = nfs(number, 4, 1);

  // Style the text.
  textAlign(CENTER, CENTER);
  textFont('Courier New');
  textSize(16);

  // Display a negative version of the number and
  // the formatted positive version.
  text('-0123.5', 50, 33);
  text(formatted, 50, 67);

  describe(
    'The numbers "-0123.5" and "0123.5" written on separate
lines. The numbers align vertically. The text is in black on a
gray background.'
```

Syntax

```
nfs(num, [left], [right])
```

```
nfs(nums, [left], [right])
```

Parameters

num	Number: number to format.
left	Integer: number of digits to include to the left of the decimal point.
right	Integer: number of digits to include to the right of the decimal point.
nums	Array: numbers to format.

Returns

String: formatted string.

This page is generated from the comments in src/utilities/string_functions.js . Please feel free to edit it and submit a pull request!

Related References

join
Combines an array of strings into one string.

match
Applies a regular expression to a string and returns an array with the first match.

matchAll
Applies a regular expression to a string and returns an array of matches.

nf
Converts a Number into a String with a given number of digits.