

Inflector Helper

The Inflector Helper file contains functions that permits you to change **English** words to plural, singular, camel case, etc.

- [Loading this Helper](#)
- [Available Functions](#)

Loading this Helper

This helper is loaded using the following code:

```
helper('inflector');
```

Available Functions

The following functions are available:

singular(*\$string*)

Parameters: • *\$string* (*string*) – Input string

Returns: A singular word

Return type: string

Changes a plural word to singular. Example:

```
echo singular('dogs'); // Prints 'dog'
```

plural(\$string)

Parameters: • \$string (string) – Input string

Returns: A plural word

Return type: string

Changes a singular word to plural. Example:

```
echo plural('dog'); // Prints 'dogs'
```

camelize(\$string)

Parameters: • \$string (string) – Input string

Returns: Camelized string

Return type: string

Changes a string of words separated by spaces or underscores to camel case. Example:

```
echo camelize('my_dog_spot'); // Prints 'myDogSpot'
```

underscore(\$string)

Parameters: • **\$string** (*string*) – Input string

Returns: String containing underscores instead of spaces

Return type: string

Takes multiple words separated by spaces and underscores them. Example:

```
echo underscore('my dog spot'); // Prints 'my_dog_spot'
```

humanize(\$string[, \$separator = '_'])

Parameters: • **\$string** (*string*) – Input string
• **\$separator** (*string*) – Input separator

Returns: Humanized string

Return type: string

Takes multiple words separated by underscores and adds spaces between them. Each word is capitalized.

Example:

```
echo humanize('my_dog_spot'); // Prints 'My Dog Spot'
```

To use dashes instead of underscores:

```
echo humanize('my-dog-spot', '-'); // Prints 'My Dog Spot'
```

is_countable(\$word)

Parameters: • **\$word** (*string*) – Input string

Returns: TRUE if the word is countable or FALSE if not

Return type: bool

Checks if the given word has a plural version. Example:

```
is_countable('equipment'); // Returns FALSE
```

dasherize(\$string)

Parameters: • **\$string** (*string*) – Input string

Returns: Dasherized string

Return type: string

Replaces underscores with dashes in the string. Example:

```
dasherize('hello_world'); // Returns 'hello-world'
```

ordinal(\$integer)

Parameters: • **\$integer** (*int*) – The integer to determine the suffix

Returns: Ordinal suffix

Return type: string

Returns the suffix that should be added to a number to denote the position such as 1st, 2nd, 3rd, 4th. Example:

```
ordinal(1); // Returns 'st'
```

ordinalize(\$integer)

Parameters: • **\$integer** (*int*) – The integer to ordinalize

Returns: Ordinalized integer

Return type: string

Turns a number into an ordinal string used to denote the position such as 1st, 2nd, 3rd, 4th. Example:

```
ordinalize(1); // Returns '1st'
```

[← Previous](#)

[Next →](#)

© Copyright 2014-2018 British Columbia Institute of Technology. Last updated on Jan 09, 2018.

Built with [Sphinx](#) using a [theme](#) provided by [Read the Docs](#).