

Pluralize — *Smooth CoffeeScript*



This literate program is *interactive* in its HTML form. Edit a CoffeeScript segment to try it. You can see the generated JavaScript as you modify a CoffeeScript function by typing 'show name' after its definition.

Pluralize

How do you add 's' to a word when there is more than one of something?

It is often seen in JavaScript with the ternary operator `?:` but that is hard-coding at the expense of future maintenance and internationalization.

First some preliminaries. Let's `show` the results whether we are running in a server or a web client. Something like `show = if exports? then console.log else alert`. And wrap the different methods in a `test` function so we can see if they work.

```
test = (func) ->
  for points in [-3..3]
    func points
```

Readable ways

The simplest readable way

```
test (points) ->
  show "0: You got #{points} point#{if points > 1 then 's' else ''}"
```

Encapsulate in a function, future i18n will be easier

```
test (points) ->
  pluralIf = (stem, cond) -> stem + (if cond then 's' else '')
  show "1: You got #{points} #{pluralIf 'point', points > 1}"
```

Encapsulate in a function, handle negative numbers

```
test (points) ->
  pluralUnless = (stem, cond) -> stem + (unless cond then 's' else '')
  show "2: You got #{points} #{pluralUnless 'point', -2 < points < 2}"
```

Encapsulate in a String method, could collide with another definition

```
test (points) ->
  String::pluralIf ?= (cond) -> this + (if cond then 's' else '')
  show "3: You got #{points} #{'point'.pluralIf points > 1}"
```

Tricks

Use the existential operator to catch the singular undefined

```
test (points) ->
  show "4: You got #{points} point#{('s' if points > 1) ? ''}"
```

Convert the condition to a number and do an array lookup

```
test (points) ->
  show "5: You got #{points} point#{['', 's'][+(points > 1)]}"
```

Use the inverted condition to ask for a letter.

```
true => charAt 1 => '' — false => charAt 0 => 's'
```

```
test (points) ->
  show "6: You got #{points} point#{'s'.charAt points <= 1}"
```

Create an array whose length is determined by the condition. It consists of empty elements so join it with 's'

```
test (points) ->
  show "7: You got #{points} point#{Array(1+(points>1)).join 's'}"
```

Concatenate to an empty array, undefined does nothing

```
test (points) ->
  show "8: You got #{points} point#[].concat ('s' if points > 1)]"
```

Contributed

Create an array with either 's' or undefined — from [satyr](#)

```
test (points) ->
  show "9: You got #{points} point#{['s' if points > 1]}"
```

Output

```
1 0: You got -3 point
2 0: You got -2 point
3 0: You got -1 point
4 0: You got 0 point
5 0: You got 1 point
6 0: You got 2 points
7 0: You got 3 points
8 1: You got -3 point
9 1: You got -2 point
10 1: You got -1 point
11 1: You got 0 point
12 1: You got 1 point
13 1: You got 2 points
14 1: You got 3 points
15 2: You got -3 points
16 2: You got -2 points
17 2: You got -1 point
18 2: You got 0 point
19 2: You got 1 point
20 2: You got 2 points
21 2: You got 3 points
22 3: You got -3 point
23 3: You got -2 point
24 3: You got -1 point
25 3: You got 0 point
26 3: You got 1 point
27 3: You got 2 points
28 3: You got 3 points
29 4: You got -3 point
30 4: You got -2 point
31 4: You got -1 point
32 4: You got 0 point
33 4: You got 1 point
34 4: You got 2 points
35 4: You got 3 points
36 5: You got -3 point
37 5: You got -2 point
38 5: You got -1 point
39 5: You got 0 point
40 5: You got 1 point
41 5: You got 2 points
42 5: You got 3 points
43 6: You got -3 point
```

```

44 6: You got -2 point
45 6: You got -1 point
46 6: You got 0 point
47 6: You got 1 point
48 6: You got 2 points
49 6: You got 3 points
50 7: You got -3 point
51 7: You got -2 point
52 7: You got -1 point
53 7: You got 0 point
54 7: You got 1 point
55 7: You got 2 points
56 7: You got 3 points
57 8: You got -3 point
58 8: You got -2 point
59 8: You got -1 point
60 8: You got 0 point
61 8: You got 1 point
62 8: You got 2 points
63 8: You got 3 points
64 9: You got -3 point
65 9: You got -2 point
66 9: You got -1 point
67 9: You got 0 point
68 9: You got 1 point
69 9: You got 2 points
70 9: You got 3 points

```

JavaScript

```

1 (function() {
2   var show, showDocument, test;
3
4   show = console.log;
5
6   showDocument = function(doc, width, height) {
7     return show(doc);
8   };
9
10  test = function(func) {
11    var points, _results;
12    _results = [];
13    for (points = -3; points <= 3; points++) {
14      _results.push(func(points));
15    }
16    return _results;
17  };
18
19  test(function(points) {
20    return show("0: You got " + points + " point" + (points > 1 ? 's' : ''));
21  });
22
23  test(function(points) {
24    var pluralIf;
25    pluralIf = function(stem, cond) {
26      return stem + (cond ? 's' : '');
27    };
28    return show("1: You got " + points + " " + (pluralIf('point', points > 1)));
29  });
30
31  test(function(points) {
32    var pluralUnless;
33    pluralUnless = function(stem, cond) {
34      return stem + (!cond ? 's' : '');
35    };
36    return show("2: You got " + points + " " + (pluralUnless('point', (-2 < points && points < 2))));
37  });
38
39  test(function(points) {
40    var _base, _ref;

```

```

41     if ((_ref = (_base = String.prototype).pluralIf) == null) {
42         _base.pluralIf = function(cond) {
43             return this + (cond ? 's' : '');
44         };
45     }
46     return show("3: You got " + points + " " + ('point'.pluralIf(points > 1)));
47 });
48
49 test(function(points) {
50     var _ref;
51     return show("4: You got " + points + " point" + ((_ref = (points > 1 ? 's' : void 0)) != null ? _ref : ''));
52 });
53
54 test(function(points) {
55     return show("5: You got " + points + " point" + ['', 's'][+(points > 1)]);
56 });
57
58 test(function(points) {
59     return show("6: You got " + points + " point" + ('s'.charAt(points <= 1)));
60 });
61
62 test(function(points) {
63     return show("7: You got " + points + " point" + (Array(1 + (points > 1)).join('s')));
64 });
65
66 test(function(points) {
67     return show("8: You got " + points + " point" + ([].concat((points > 1 ? 's' : void 0))));
68 });
69
70 test(function(points) {
71     return show("9: You got " + points + " point" + [points > 1 ? 's' : void 0]);
72 });
73
74 }).call(this);

```

Formats [CoffeeScript](#) [Markdown](#) [PDF](#) [HTML](#)

License [Creative Commons Attribution Share Alike](#) by autotelicum © 2554/2011