



```
_.defaults({ 'a': 1 }, { 'a': 3, 'b': 2 });  
// → { 'a': 1, 'b': 2 }  
_.partition([1, 2, 3, 4], n => n % 2);  
// → [[1, 3], [2, 4]]
```

Download

📄 [Core build](#) (~4kB gzipped)

📄 [Full build](#) (~24kB gzipped)

🔗 [CDN copies](#)

Lodash is released under the [MIT license](#) & supports modern environments. Review the [build differences](#) & pick one that's right for you.

Installation

In a browser:

```
<script src="lodash.js"></script>
```

Using npm:

```
$ npm i -g npm  
$ npm i --save lodash
```

In Node.js:

```
// Load the full build.  
var _ = require('lodash');  
// Load the core build.  
var _ = require('lodash/core');  
// Load the FP build for immutable auto-curried iteratee-first data-last methods.  
var fp = require('lodash/fp');  
  
// Load method categories.  
var array = require('lodash/array');  
var object = require('lodash/fp/object');  
  
// Cherry-pick methods for smaller browserify/rollup/webpack bundles.  
var at = require('lodash/at');  
var curryN = require('lodash/fp/curryN');
```

Note:

Install [n_](#) for Lodash use in the Node.js < 6 REPL.

Why Lodash?

Lodash makes JavaScript easier by taking the hassle out of working with arrays, numbers, objects, strings, etc. Lodash's modular methods are great for:

- › Iterating arrays, objects, & strings
- › Manipulating & testing values
- › Creating composite functions

Module Formats

Lodash is available in a [variety of builds](#) & module formats.

- › [lodash](#) & [per method packages](#)
- › [lodash-es](#), [babel-plugin-lodash](#), & [lodash-webpack-plugin](#)
- › [lodash/fp](#)
- › [lodash-amd](#)

Complementary Tools

- › [futil-js](#) is a set of functional utilities designed to complement lodash

Further Reading

- › [Contributing](#)
- › [Release Notes](#)
- › [Wiki \(Changelog, Roadmap, etc.\)](#)

Support

Tested in Chrome 65-66, Firefox 58-59, IE 11, Edge 16, Safari 10-11, Node.js 6-10, & PhantomJS 2.1.1. Automated [browser](#) & [CI](#) test runs are available.