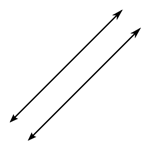
# run-parallel [travis](https://travis-ci.org/feross/run-parallel) [npm](https://npmjs.org/package/run-parallel) [downloads](https://npmjs.org/package/run-parallel) [javascript style guide](https://standardjs.com)

### Run an array of functions in parallel

 [Sauce Test Status](https://saucelabs.com/u/run-parallel)

### install

npm install run-parallel

### usage

#### parallel(tasks, [callback])

Run the tasks array of functions in parallel, without waiting until the previous function has completed. If any of the functions pass an error to its callback, the main callback is immediately called with the value of the error. Once the tasks have completed, the results are passed to the final callback as an array.

It is also possible to use an object instead of an array. Each property will be run as a function and the results will be passed to the final callback as an object instead of an array. This can be a more readable way of handling the results.

##### arguments

* tasks - An array or object containing functions to run. Each function is passed a callback(err, result) which it must call on completion with an error err (which can be null) and an optional result value.
* callback(err, results) - An optional callback to run once all the functions have completed. This function gets a results array (or object) containing all the result arguments passed to the task callbacks.

##### example

var parallel = require('run-parallel')

parallel([

function (callback) {

setTimeout(function () {

callback(null, 'one')

}, 200)

},

function (callback) {

setTimeout(function () {

callback(null, 'two')

}, 100)

}

],

// optional callback

function (err, results) {

// the results array will equal ['one','two'] even though

// the second function had a shorter timeout.

})

This module is basically equavalent to [async.parallel](https://github.com/caolan/async#paralleltasks-callback), but it's handy to just have the one function you need instead of the kitchen sink. Modularity! Especially handy if you're serving to the browser and need to reduce your javascript bundle size.

Works great in the browser with [browserify](http://browserify.org/)!

### see also

* [run-auto](https://github.com/feross/run-auto)
* [run-parallel-limit](https://github.com/feross/run-parallel-limit)
* [run-series](https://github.com/feross/run-series)
* [run-waterfall](https://github.com/feross/run-waterfall)

### license

MIT. Copyright (c) [Feross Aboukhadijeh](http://feross.org).