

> Terminal string styling done right [![Build Status](https://travis-ci.org/chalk/chalk.svg?branch=master)](https://travis-ci.org/chalk/chalk) [![Coverage Status](https://coveralls.io/repos/chalk/chalk/badge.svg?branch=master)](https://coveralls.io/r/chalk/chalk?branch=master) [![](http://img.shields.io/badge/unicorn-approved-ff69b4.svg)](https://www.youtube.com/watch?v=9auOCbH5Ns4) [colors.js](https://github.com/Marak/colors.js) used to be the most popular string styling module, but it has serious deficiencies like extending `String.prototype` which causes all kinds of [problems](https://github.com/yeoman/yo/issues/68). Although there are other ones, they either do too much or not enough. \*\*Chalk is a clean and focused alternative.\*\* ![](https://github.com/chalk/ansi-styles/raw/master/screenshot.png) ## Why - Highly performant - Doesn't extend `String.prototype` - Expressive API - Ability to nest styles - Clean and focused - Auto-detects color support - Actively maintained - [Used by ~4500 modules](https://www.npmjs.com/browse/depended/chalk) as of July 15, 2015 ## Install ``` $ npm install --save chalk ``` ## Usage Chalk comes with an easy to use composable API where you just chain and nest the styles you want. ```js var chalk = require('chalk'); // style a string chalk.blue('Hello world!'); // combine styled and normal strings chalk.blue('Hello') + 'World' + chalk.red('!'); // compose multiple styles using the chainable API chalk.blue.bgRed.bold('Hello world!'); // pass in multiple arguments chalk.blue('Hello', 'World!', 'Foo', 'bar', 'biz', 'baz'); // nest styles chalk.red('Hello', chalk.underline.bgBlue('world') + '!'); // nest styles of the same type even (color, underline, background) chalk.green( 'I am a green line ' + chalk.blue.underline.bold('with a blue substring') + ' that becomes green again!' ); ``` Easily define your own themes. ```js var chalk = require('chalk'); var error = chalk.bold.red; console.log(error('Error!')); ``` Take advantage of console.log [string substitution](http://nodejs.org/docs/latest/api/console.html#console\_console\_log\_data). ```js var name = 'Sindre'; console.log(chalk.green('Hello %s'), name); //=> Hello Sindre ``` ## API ### chalk.`