# supports-color [Build Status](https://travis-ci.org/chalk/supports-color)

Detect whether a terminal supports color

## Install

$ npm install supports-color

## Usage

const supportsColor = require('supports-color');

if (supportsColor.stdout) {

console.log('Terminal stdout supports color');

}

if (supportsColor.stdout.has256) {

console.log('Terminal stdout supports 256 colors');

}

if (supportsColor.stderr.has16m) {

console.log('Terminal stderr supports 16 million colors (truecolor)');

}

## API

Returns an Object with a stdout and stderr property for testing either streams. Each property is an Object, or false if color is not supported.

The stdout/stderr objects specifies a level of support for color through a .level property and a corresponding flag:

* .level = 1 and .hasBasic = true: Basic color support (16 colors)
* .level = 2 and .has256 = true: 256 color support
* .level = 3 and .has16m = true: Truecolor support (16 million colors)

## Info

It obeys the --color and --no-color CLI flags.

Can be overridden by the user with the flags --color and --no-color. For situations where using --color is not possible, add the environment variable FORCE\_COLOR=1 to forcefully enable color or FORCE\_COLOR=0 to forcefully disable. The use of FORCE\_COLOR overrides all other color support checks.

Explicit 256/Truecolor mode can be enabled using the --color=256 and --color=16m flags, respectively.

## Related

* [supports-color-cli](https://github.com/chalk/supports-color-cli) - CLI for this module
* [chalk](https://github.com/chalk/chalk) - Terminal string styling done right

## Maintainers

* [Sindre Sorhus](https://github.com/sindresorhus)
* [Josh Junon](https://github.com/qix-)

## License

MIT