# emoji-regex [Build status](https://travis-ci.org/mathiasbynens/emoji-regex)

*emoji-regex* offers a regular expression to match all emoji symbols and sequences (including textual representations of emoji) as per the Unicode Standard.

This repository contains a script that generates this regular expression based on [Unicode data](https://github.com/node-unicode/node-unicode-data). Because of this, the regular expression can easily be updated whenever new emoji are added to the Unicode standard.

## Installation

Via [npm](https://www.npmjs.com/):

npm install emoji-regex

In [Node.js](https://nodejs.org/):

const emojiRegex = require('emoji-regex/RGI\_Emoji.js');

// Note: because the regular expression has the global flag set, this module

// exports a function that returns the regex rather than exporting the regular

// expression itself, to make it impossible to (accidentally) mutate the

// original regular expression.

const text = `

\u{231A}: ⌚ default emoji presentation character (Emoji\_Presentation)

\u{2194}\u{FE0F}: ↔️ default text presentation character rendered as emoji

\u{1F469}: 👩 emoji modifier base (Emoji\_Modifier\_Base)

\u{1F469}\u{1F3FF}: 👩🏿 emoji modifier base followed by a modifier

`;

const regex = emojiRegex();

let match;

while (match = regex.exec(text)) {

const emoji = match[0];

console.log(`Matched sequence ${ emoji } — code points: ${ [...emoji].length }`);

}

Console output:

Matched sequence ⌚ — code points: 1

Matched sequence ⌚ — code points: 1

Matched sequence ↔️ — code points: 2

Matched sequence ↔️ — code points: 2

Matched sequence 👩 — code points: 1

Matched sequence 👩 — code points: 1

Matched sequence 👩🏿 — code points: 2

Matched sequence 👩🏿 — code points: 2

## Regular expression flavors

The package comes with three distinct regular expressions:

// This is the recommended regular expression to use. It matches all

// emoji recommended for general interchange, as defined via the

// `RGI\_Emoji` property in the Unicode Standard.

// https://unicode.org/reports/tr51/#def\_rgi\_set

// When in doubt, use this!

const emojiRegexRGI = require('emoji-regex/RGI\_Emoji.js');

// This is the old regular expression, prior to `RGI\_Emoji` being

// standardized. In addition to all `RGI\_Emoji` sequences, it matches

// some emoji you probably don’t want to match (such as emoji component

// symbols that are not meant to be used separately).

const emojiRegex = require('emoji-regex/index.js');

// This regular expression matches even more emoji than the previous

// one, including emoji that render as text instead of icons (i.e.

// emoji that are not `Emoji\_Presentation` symbols and that aren’t

// forced to render as emoji by a variation selector).

const emojiRegexText = require('emoji-regex/text.js');

Additionally, in environments which support ES2015 Unicode escapes, you may require ES2015-style versions of the regexes:

const emojiRegexRGI = require('emoji-regex/es2015/RGI\_Emoji.js');

const emojiRegex = require('emoji-regex/es2015/index.js');

const emojiRegexText = require('emoji-regex/es2015/text.js');

## For maintainers

### How to update emoji-regex after new Unicode Standard releases

1. Update the Unicode data dependency in package.json by running the following commands:

# Example: updating from Unicode v12 to Unicode v13.

npm uninstall @unicode/unicode-12.0.0

npm install @unicode/unicode-13.0.0 --save-dev

1. Generate the new output:

npm run build

1. Verify that tests still pass:

npm test

1. Send a pull request with the changes, and get it reviewed & merged.
2. On the main branch, bump the emoji-regex version number in package.json:

npm version patch -m 'Release v%s'

Instead of patch, use minor or major [as needed](https://semver.org/).

Note that this produces a Git commit + tag.

1. Push the release commit and tag:

git push

Our CI then automatically publishes the new release to npm.

## Author

| [**twitter/mathias**](https://twitter.com/mathias) |
| --- |
| [Mathias Bynens](https://mathiasbynens.be/) |

## License

*emoji-regex* is available under the [MIT](https://mths.be/mit) license.