# Contributing Guidelines

So you want to contribute to Push, huh? Well lucky for you, it's really easy to do so, because you're just dealing with like, a few hundred lines of JavaScript. It's not hard.

Alright. Now calm down and take a few deep breaths. Here we go. All you have to remember is two commands... think you can do that?

To BUILD Push, just run:

```bash

$ npm run build

```

To TEST Push on BrowserStack, run:

```bash

$ npm run test

```

See? Not hard at all. Unfortunately the Notifications API doesn't always play nicely with local sites, so don't get discouraged if you try running Push in a local HTML file and it doesn't work.

To TEST Push on a specific, locally-installed browser, you can run one of the following:

```bash

$ npm run test:opera

$ npm run test:firefox

$ npm run test:chrome

$ npm run test:safari

```

### Testing & Travis

Push uses the [Karma](https://karma-runner.github.io/1.0/index.html) JavaScript test runner, so read up on that if you want to make changes to any of the tests that are run. These tests are run post-push by [Travis CI](https://travis-ci.org), so look into that if you want to make any Travis configuration changes. Although, at this point I'd say Travis is all set. The tests might want to be expanded though.

### REAL IMPORTANT STUFF

\*\*THERE IS ONLY ONE RULE TO PUSH CLUB\*\* (and no, it's not that you can't talk about it). \*\*WHENEVER\*\* you make changes to `Push.js`, \*\*RECOMPILE\*\* and commit `push.min.js` as well. Until this build process can be wrapped into a sexy git hook of some sort, this is how changes to the library need to occur. \*\*YOUR PR WILL NOT BE APPROVED UNLESS THIS HAPPENS\*\*. That said, I did let it slide once because I wasn't thinking, but that's why I wrote this file to make sure it will never happen again.

Outside of that, contributing should not be at all scary and should be a fun and positive process. Now go out and write some killer JS! Wait... is there even such a thing?