**Frontend Developer Resources 2022 — Best Free pack to start with -** [Anshu Mishra](https://medium.com/@anshumishra1168?source=post_page-----42f5ea1bcc7--------------------------------) Jul 23, 2022

If you have an educational stipend at work, use it. Also, many public libraries give you access for free to paid resources like LinkedIn Learning. Check out your local library before purchasing certain content. Also, just a massive shoutout to public libraries.

I work with Preact daily, and I also work with React. There are other libraries and frameworks out there, but I will veer away from all that in this post. The browser extensions section is the only place where I’ll mention frameworks and libraries.

This post is a work in progress, so expect this list to update throughout 2022.

**JavaScript**

Regardless of whether or not you start with a library or framework, at some point, as a frontend developer, you will need to understand JavaScript in more depth. Here are some excellent resources to get you there.

* [JavaScript 30](https://javascript30.com/) — A classic care of [Wes Bos](https://wesbos.com/). These are fun projects to help solidify some JS fundamentals.
* [JavaScript Katas](https://jskatas.org/) — I found this resource a long time ago, but it’s gotten a refresh in recent years. I even did a live stream of doing some of the Katas!
* [patterns.dev](https://www.patterns.dev/)

**TypeScript**

TypeScript is getting more and more popular every year. Even if you’re not a fan of it, it’s good to be at least aware of it.

* Anything by [Marius Schulz](https://mariusschulz.com/) — Marius has a tonne of great blog posts about TypeScript and some great courses on Egghea.
* [Practical Advanced TypeScript](https://egghead.io/courses/practical-advanced-typescript) — I loved this course from Rares Matei
* [The TypeScript team’s 2021 type | treat series on DEV](https://dev.to/orta/series/15226) — Orta from the TypeScript team put some amazing challenges together this year
* [TypeScript Playground](https://www.typescriptlang.org/play) — The TypeScript Playground keeps improving and is a great place to test stuff out.
* [Learn TypeScript from the Ground Up with James Henry](https://typescriptcourses.com/typescript-fundamentals) — A great free course from James Henry. If you enjoy it, consider buying his advanced course.

**CSS**

Let’s skip the centering jokes in CSS. We’re in the 2020s now. This is relatively easy to do nowadays. Dig in and level up on your CSS.

* [Andy Bell’s Modern CSS Reset](https://piccalil.li/blog/a-modern-css-reset/)
* [Josh W. Comeau’s CSS Reset](https://www.joshwcomeau.com/css/custom-css-reset/)
* [Designing Beautiful Shadows in CSS](https://www.joshwcomeau.com/css/designing-shadows/)
* [CSS for JavaScript Developers](https://css-for-js.dev/)
* [Grid by Example](https://gridbyexample.com/)
* [cssgrid.io](https://cssgrid.io/)
* [Modern CSS Solutions](https://moderncss.dev/)
* [SmolCSS](https://smolcss.dev/)
* [Every Layout](https://every-layout.dev/)
* [A Primer On CSS Container Queries](https://www.smashingmagazine.com/2021/05/complete-guide-css-container-queries)

**HTML**

If you’re doing web development long-term, you must become familiar with semantic markup. Pass on the order for <div /> soup.

* [MDN](https://developer.mozilla.org/en-US/) as always
* [HTMHell](https://www.htmhell.dev/) — A great site with horrible examples of HTML and how to fix them
* [Periodic table of HTML elements](https://madebymike.github.io/html5-periodic-table) — A fun spin on the classic periodic table
* This one isn’t just HTML, so I should probably create another section. Check out [buildexcellentwebsit.es](https://buildexcellentwebsit.es/) for key foundations & principles of building great sites.

**Accessibility**

Accessibility is super important and honestly, if you want to stand out as a frontend developer, levelling up here is a good move.

* [WebAIM’s contrast checker](https://webaim.org/resources/contrastchecker) — An easy way to check colour contrast
* [ButtonBuddy](https://buttonbuddy.dev/) — A fun interactive building tool to create accessible buttons
* [Inclusive Components](https://inclusive-components.design/) — A great site on how to build inclusive components
* [Forem Developer documentation on accessibility](https://developers.forem.com/frontend/accessibility) — I’m biased, but I love our a11y documentation
* Marcy Sutton has a [great list of a11y tools](https://workshop-resources.testingaccessibility.com/) on her course site [testingaccessibility.com](https://testingaccessibility.com/)
* 5 things I’m thinking about when I check a Pull Request for accessibility — A great checklist of things to look for when reviewing frontend related pull requests

**Animations**

Animations are a great way of enhancing a user experience, but remember, don’t add animations to your site for the sake of adding animations. Here’s some great people’s work to check out in this space.

* [Anything by Jhey Thompkins](https://codepen.io/jh3y)
* [Anything by Cassie Evans](https://www.cassie.codes/)
* I’m biased, but my co-worker [@coffeecraftcode](https://dev.to/coffeecraftcode) is a whiz when it comes to animations. Check out her [codepen profile](https://codepen.io/cgorton/" \t "_blank) and [LinkedIn courses](https://www.linkedin.com/learning/instructors/christina-gorton). I’ve paired with her a bunch of times to learn about animations, so I can 100% guarantee that she knows her stuff.
* [prefers-reduced-motion: Taking a no-motion-first approach to animations](https://www.tatianamac.com/posts/prefers-reduced-motion) by [Tatiana Mac](https://twitter.com/TatianaTMac) — This falls under accessibility concerning animations, but I’m putting it here to make it stand out a bit more

**Jamstack**

* [Jamstack Explorers](https://explorers.netlify.com/)
* [jamstack.org](https://jamstack.org/)
* [Incremental Static Regeneration](https://vercel.com/docs/concepts/next.js/incremental-static-regeneration)
* [Distributed Persistent Rendering](https://www.netlify.com/blog/2021/04/14/distributed-persistent-rendering-a-new-jamstack-approach-for-faster-builds)
* [Understanding Rendering in the Jamstack](https://bejamas.io/blog/understanding-rendering-in-the-jamstack)

**Browser Extensions**

* [WAVE](https://wave.webaim.org/extension) — “evaluate web content for accessibility issues directly within your browser”
* [axe](https://www.deque.com/axe/browser-extensions) — “test your web applications to help identify and resolve common accessibility issues”
* [Accessibility Insights for Web](https://accessibilityinsights.io/docs/en/web/overview/) — “helps developers find and fix accessibility issues in web apps and sites”
* If you’re working with React: React DevTools ([Chromium based browsers](https://chrome.google.com/webstore/detail/react-developer-tools/fmkadmapgofadopljbjfkapdkoienihi) | [Firefox](https://addons.mozilla.org/en-CA/firefox/addon/react-devtools))
* If you’re working with Preact: [Preact DevTools](https://preactjs.github.io/preact-devtools/" \t "_blank)
* [VisBug](https://github.com/GoogleChromeLabs/ProjectVisBug#installation) — Open source web design debug tools

**Testing**

Testing is a big topic, and I’ll scratch the surface here, but knowing what to test is super important. At the end of the day, when you ship something, ask yourself, “Do I feel confident with shipping this?”.

* I’m a fan of Cypress, and we use it at Forem (the software that powers dev.to). Over time, we’ve been building out our documentation, and I think it’s a great resource if you dive into end-to-end testing. Check out [Writing Cypress Tests](https://developers.forem.com/tests/e2e-tests#writing-cypress-tests)
* Testing library’s suite of tools. Whether it’s component tests in React, Svelte, plain old HTML/CSS/JS or end to end testing, [Testing Library](https://testing-library.com/) has you covered.

**Cloud IDEs/full-blown development environments**

* [Gitpod](https://gitpod.io/)
* [vscode.dev](https://vscode.dev/)
* [CodeSpaces](https://github.com/features/codespaces) (if you have access)
* [Stackblitz](https://stackblitz.com/)
* [CodeSandbox](https://codesandbox.io/)
* [vscodetips.com](https://vscodetips.com/) — a community discussing all things VS Code including cloud IDEs. Full disclosure that this is a shameless plug. It’s a new community I started this year.

**Living on the Edge**

A lot of cool stuff is happening on The Edge. I haven’t done much here, but looking to level up in 2022.

* [Cloudflare Workers](https://workers.cloudflare.com/) — Includes stuff like Durable Objects, key value stores etc.
* [Cloudflare Pages](https://pages.cloudflare.com/)
* [Living on the edge: A deep dive into Netlify Edge Functions](https://www.netlify.com/blog/deep-dive-into-netlify-edge-functions/)
* [Vercel Edge Functions](https://vercel.com/docs/concepts/functions/edge-functions)

**Tooling**

We all love tools. There’s plenty to go around, but I’ll touch on a few.

* Storybook — [learnstorybook.com](https://learnstorybook.com/)
* If you have Node.js installed and need to configure a JavaScript-based project’s gitignore file from the root of the project, run npx gitgnore node (It works for other languages as well, or if you omit the language, it will give you a list of all the supported gitignores)
* [Parcel](https://parceljs.org/)
* [Vite](https://vitejs.dev/)
* [Turborepo](https://turborepo.org/)

**Books**

If you have some of your student savings please go for books as books are the ultimate think to learn anything.

These are some great reads.

* [Road to Learn React](https://amzn.to/3vbfmHh)
* [JavaScript the Definitive Guide](https://amzn.to/3vazo4O)
* [Beginning React](https://amzn.to/3OtZvdK)

**Twitch Streams**

I’m a big fan of Twitch. I stream on my own at [livecoding.ca](https://livecoding.ca/) as well as on the [DEV Twitch stream](https://www.twitch.tv/thepracticaldev). Here are some great folks you should be watching for all things frontend.

* [TrostCodes](https://www.twitch.tv/trostcodes)
* [Stephanie Eckles](https://www.twitch.tv/5t3phdev)
* [Jhey Tompkins](https://www.twitch.tv/jh3yy)
* [Michael Chan](https://www.twitch.tv/lunchdev)
* [Ben Myers](https://www.twitch.tv/SomeAnticsDev)
* [Learn with Jason](https://www.learnwithjason.dev/)

I’m sure I’ve missed some stuff, but this is the initial brain dump. As mentioned initially, I will update this over the coming year.