



Browserify lets you `require('modules')` in the browser by bundling up all of your dependencies.

[Install](#)[Documentation](#)[Help + Articles](#)[Sweet Demos](#)

Install

use Browserify from the Command Line

First install [node](#), which ships with [npm](#). Then do:

```
npm install -g browserify
```

Hello World With Browserify

Bundle up your first module

Browsers don't have the `require` method defined, but Node.js does. With Browserify you can write code that uses `require` in the same way that you would use it in Node.

Here is a tutorial on how to use Browserify on the command line to bundle up a simple file called `main.js` along with all of its dependencies:

`main.js`

```
var unique = require('uniq');
```

```
var data = [1, 2, 2, 3, 4, 5, 5, 5, 6];
```

```
console.log(unique(data));
```

Install the `uniq` module with [npm](#):

```
npm install uniq
```

Now recursively bundle up all the required modules starting at `main.js` into a single file called `bundle.js` with the `browserify` command:

```
browserify main.js -o bundle.js
```

Browserify parses the [AST](#) for `require()` calls to traverse the entire dependency graph of your project. Drop a single `<script>` tag into your html and you're done!

```
<script src="bundle.js"></script>
```

More

- Read the [Browserify Handbook](#)
- Use many of the tens of thousands of modules on NPM in the browser
- Use [watchify](#), a browserify compatible caching bundler, for super-fast bundle rebuilds as you develop.
- Use `--debug` when creating bundles to have Browserify automatically include Source Maps for easy debugging.
- Check out tools like [Beefy](#) or [run-browser](#) which make automating browserify development easier.
- `process.nextTick()`, `__dirname`, and `__filename` node-isms work
- Get browser versions of the node core libraries [events](#), [stream](#), [path](#), [url](#), [assert](#), [buffer](#), [util](#), [querystring](#), [http](#), [vm](#), and [crypto](#) when you `require()` them
- Test your modules in all the browsers with [testling-ci](#). Many modules known to work with browserify will have a [testling-ci](#) badge:



Community

Join [irc.freenode.net/#browserify](irc://freenode.net/#browserify) for help, or tweet [@browserify](#).

Who Uses Browserify

here are some things people have said

At Mapbox we build our website and JavaScript API with Browserify. It makes the structure and modularity of our code rock. - [tmcw](#), developer at [Mapbox](#).

Browserify does exactly what it says it does and it does it well. I'm glad it exists. - [zpao](#), developer on the [React](#) team at Facebook (React uses Browserify).

Browserify radically sped up builds, simplified builds and encouraged modularity. - [terinjokes](#), developer at CloudFlare.

View more at the [Browserify in the wild](#) page.



This website is open source and you can fork it [on GitHub](#). CSS was originally forked from [the-gss](#).