

# The Front End Stack

As of 2019

# A quick history lesson

HTML, CSS, and JavaScript throughout the years

# HTML (Hypertext Markup Language)

Plain ol' HTML files



Preprocessors (PHP, Jinja2, Handlebars)



HTML-in-JS (JSX, Angular, Vue.js)

# CSS (Cascading Style Sheets)

CSS-in-HTML (Inline styles)



External CSS files



Preprocessors (Sass, Less, Stylus)



CSS-in-JS (styled-components, glamor, emotion)

# JavaScript (the old parts)

One global scope



Namespacing



IIFE (Immediately Invoked Function Expression)



CommonJS modules (Node.js)

# JavaScript (the new parts)

ES6 Modules



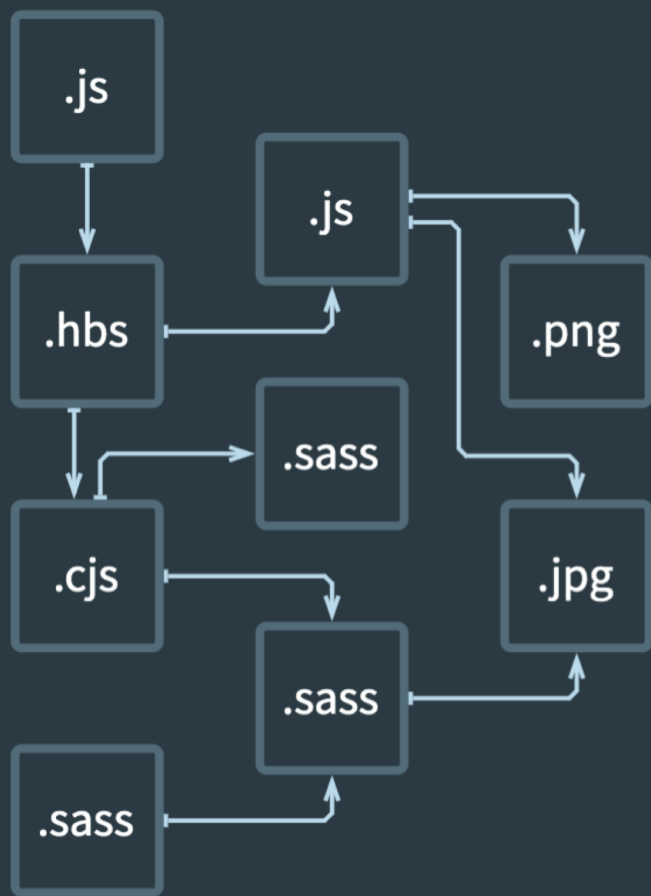
```
graph TD; A[ES6 Modules] --> B[Dynamic imports]; B --> C[Entering the mighty webpack];
```

Dynamic imports

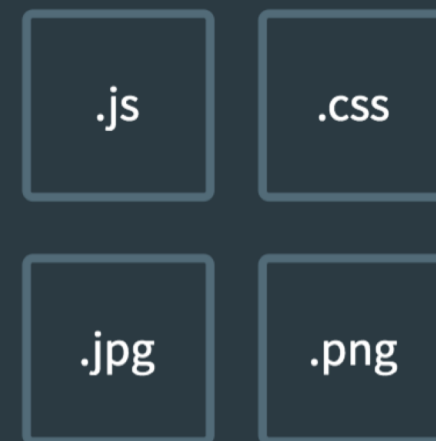
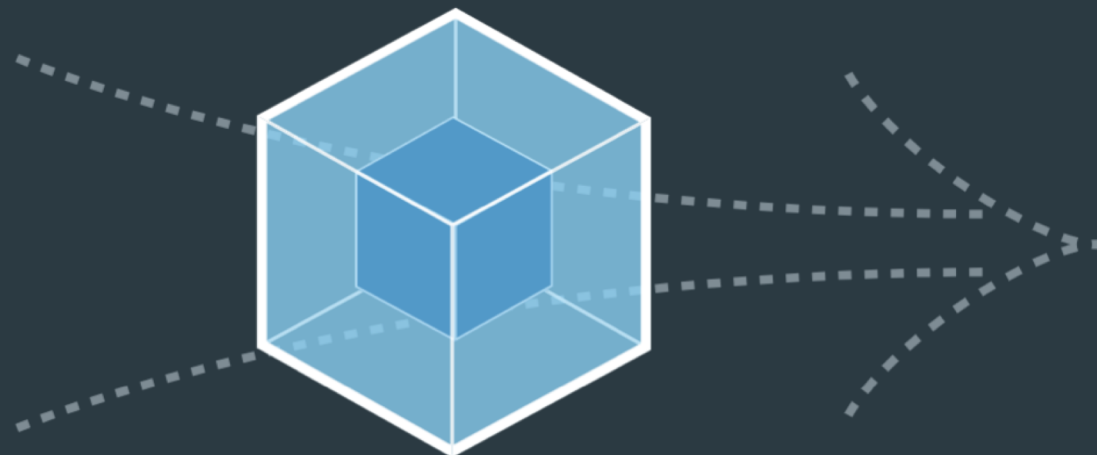
Entering the mighty *webpack*

# webpack

Putting stuff together (aka module bundler)



MODULES WITH DEPENDENCIES



STATIC ASSETS



# Babel

Transform something into usable JavaScript (aka compiler)

```
1 function foo () {
2     return 'foo'
3 }
4
5 class Bar {
6     constructor () {
7         this.value = foo()
8     }
9 }
10
11 new Bar()
12
```

```
1 "use strict";
2
3 function _classCallCheck(instance, Constructor) {
4     if (!(instance instanceof Constructor)) {
5         throw new TypeError("Cannot call a class as a function");
6     }
7 }
8
9 function foo() {
10     return "foo";
11 }
12
13 var Bar = function Bar() {
14     _classCallCheck(this, Bar);
15
16     this.value = foo();
17 };
18
19 new Bar();
20
```