

# ES6 MODULES

*What? More?*

# ES6 MODULES

- **ECMAScript 6 (2015) approved a native module system for JavaScript using a new set of keywords:**
  - `import <thing> from '<file>'`
  - `export <thing>`
  - `export default <thing>`

# RATIONALE

- **Import/export was designed:**
  - **to enable developers to split their JavaScript code into multiple files or “modules”**
  - **to support both sync and async module loading**
  - **are not dynamic, which allows static analysis (i.e. tools can examine your code without running it)**

# DEFAULT EXPORT

a.js

```
const foo = () => { /* etc */ }  
export default foo
```

b.js

```
import foo from './a'
```

# DEFAULT EXPORT

a.js

```
const foo = () => { /* etc */ }
```

```
export default foo
```

b.js

```
import foo from './a'
```

**There can only be one default export!**

# NAMED EXPORTS

a.js

```
export const foo = () => { /* etc */ }  
export const bar = 'bar'
```

b.js

```
import {foo, bar} from './a'
```

# NAMED EXPORTS

a.js

```
export const foo = () => { /* etc */ }  
export const bar = 'bar'
```

b.js

```
import {foo, bar} from './a'
```

There can be *multiple* named exports!

# COMBINING NAMED & DEFAULT EXPORTS

a.js

```
const foo = () => { /* etc */ }  
const bar = 'bar'  
const baz = 42
```

```
export foo  
export bar  
export default baz
```

b.js

```
import {foo, bar}, baz from './a'
```



# USING MODULES ON THE FRONT-END

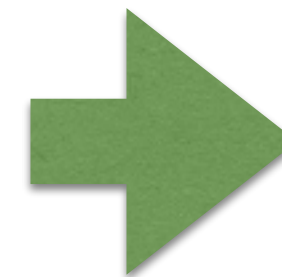
# WHERE CAN I USE IMPORT/EXPORT?

- **Import/Export has support in the latest versions of some browsers**
  - **Not quite safe yet to depend on it without a build tool like Webpack**

# WEBPACK

- Webpack is a “JavaScript Module Bundler”
  - Takes in modules with dependencies, and generates static assets representing those modules
  - Compiles your code into something the browser understands

```
<script defer src="helpers.js"></script>
<script defer src="scripts.js"></script>
<script defer src="funcs.js"></script>
<script defer src="tools.js"></script>
<script defer src="things.js"></script>
<script defer src="main.js"></script>
```



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

# WEBPACK CONFIG

- **Webpack will take a special config file where you can specify:**
  - **Entry: what's the “source” module file? The file that imports all the others? The “starting point” of your code?**
  - **Output: after Webpack bundles it all up, where should the result go?**
  - **...among other things**