Webpack cheatsheet

This is a very basic "getting started with Webpack" guide for use with Webpack v3. This doesn't cover all features, but it should get you started in understanding the config file format.

Basic config

webpack.config.js

```
module.exports = { context: __dirname, entry:
'src/app.js', output: { path: __dirname +
'/public', filename: 'app.js' } }
```

Terminal

```
npm install --save-dev webpack
```

`webpack`	build
`webpack -- -p`	build production
`webpack -- --watch`	compile continuously

This compiles `src/app.js` into `public/app.js`. (Note: you may need to use `./node_modules/.bin/webpack` as a command if you're not invoking Webpack via npm scripts.)

Multiple files

webpack.config.js

```
module.exports = { entr
vendor: 'src/vendor.js'
__dirname + '/public',
```

This creates app.js and ve

⧣ Loaders

Babel

Terminal

CSS

Terminal

```
npm install --save-dev \ babel-loader \ babel-
preset-env \ babel-preset-react
```

webpack.config.js

```
module.exports = { ··· module: { rules: [ { test:
/\.js$/, exclude: /node_modules/, use: [ { loader:
'babel-loader' } ] } ] } }
```

.babelrc

```
{ "presets": [ "env", "react" ] }
```

Adds support for Babel.

```
npm install --save-dev
```

webpack.config.js

```
module.exports = { ···
/\.css$/, exclude: /nod
'style-loader' }, { loa
}
```

Your JavaScript

```
import './styles.css' /
require('./styles.css')
```

This allows you to use CSS
packages your CSS inside

PostCSS

Terminal

```
npm install --save-dev \ postcss-loader \ postcss-
cssnext
```

webpack.config.js

```
··· // Inside module.rules[]: { test: /\.css$/,
exclude: /node_modules/, use: [ { loader: 'style-
loader' }, { loader: 'css-loader' }, { loader:
'postcss-loader' } ···
```

postcss.config.js

```
module.exports = { plugins: [ require('postcss-
cssnext')() ] }
```

This example adds postcss-cssnext support to your CSS
files.

⍣ Other features

Dev server

package.json

```
{ ⋯ "scripts": { "dev": "webpack-dev-server" } }
```

Terminal

```
npm install --save-dev \ webpack-dev-server
```

```
npm run dev
```

This starts an HTTP server for development (port 8080 by default).

⚡

▶ 💬 **0 Comments** for this cheatsheet. [Write yours!](#)

devhints.io / **Search 380+ cheatsheets** 🔍 ⌂

Other JavaScript libraries cheatsheets   **Top cheatsheets**

Over 380 curated cheatsheets, by developers for developers.

Devhints home

Chai.js
cheatsheet ●

Flow
cheatsheet ●

Elixir
cheatsheet ●

ES2015+
cheatsheet ●

Jest
cheatsheet ●

Lodash
cheatsheet ●

React.js
cheatsheet ●

Vimdiff
cheatsheet ●

Pug
cheatsheet ●

Yarn
cheatsheet ●

Vim
cheatsheet ●

Vim scripting
cheatsheet ●