

Les processeurs en CSS

Sommaire

- I. Les pré-processeurs
- II. Les post-processeurs
- III. L'automatisation

I. Les pré-processeurs

C'est quoi un préprocesseur CSS?

Un outil qui génère, à partir d'un autre langage, des fichiers CSS

NB: Voir le cours de SASS

CSS compatible tout navigateur

Les navigateurs ont tous un style de base pour chacun des éléments HTML.

Pour avoir le même résultat sur chaque navigateur, 2 démarches:

Reset -> Annuler tout le style par défaut des navigateurs

Normalize -> Aligner tous les navigateurs en comblant avec du CSS

Normalize.css

<https://github.com/necolas/normalize.css/>

```
npm install --save-dev normalize.css
```

```
/* Dans .scss */  
@import "normalize";
```

II. Les post-processeurs

POSTCSS est une librairie JavaScript qui va transformer notre CSS
<https://postcss.org/>

Se base sur un système de plugins

```
# Add postcss-cli to use in command line  
npm install --save-dev postcss postcss-cli
```

L'auto prefixer

Autoprefixer: Plugin qui ajoute Ajoute les propriétés spécifiques aux différents éditeurs de navigateurs (= *vendor prefixes*)

Permet une standardisation du comportement des navigateurs

<https://autoprefixer.github.io/>

La doc: <https://github.com/postcss/autoprefixer>

Utilisation en ligne de commande

```
npm install --save-dev autoprefixer  
npx postcss style.css --use autoprefixer -o dist/style.css
```

Dans un fichier `postcss.config.js`

```
module.exports = {  
  plugins: [  
    require('autoprefixer')  
  ]  
}
```

La minification

Technique qui permet de supprimer tout caractère inutile pour avoir un fichier moins lourd

Sont concernés:

- les espaces, tabulations ou autres sauts de ligne
- les commentaires

cssnano: Plugin POSTCSS qui minifie vos fichiers .css

```
npm install --save-dev cssnano  
npx postcss style.css --use cssnano -o dist/style.min.css
```

Dans un fichier `postcss.config.js`

```
module.exports = {  
  plugins: [  
    require('cssnano')  
  ]  
}
```

La purge du CSS inutilisé

Quand on utilise des frameworks comme Bootstrap, on se retrouve avec plein de morceaux CSS non exploités.

cf: <https://github.com/FullHuman/purgecss>

PurgeCSS: Librairie qui cherche dans vos pages `.html` tout ce qui est utilisé, et supprime le reste de vos fichiers `.css`
Existe en plugin de PostCSS

Installation

```
# -D === --save-dev  
npm i -D @fullhuman/postcss-purgecss
```

Paramétrage

Fichier `postcss.config.js`

```
const purgecss = require('@fullhuman/postcss-purgecss')

module.exports = {
  plugins: [
    purgecss({
      content: ['./**/*.html']
    })
  ]
}
```

III. L'automatisation

Dans `postcss.config.js`:

```
const purgecss = require('@fullhuman/postcss-purgecss')

module.exports = {
  plugins: [
    require('autoprefixer'),
    require('cssnano'),
    purgecss({
      content: ['./**/*.html']
    })
  ]
}
```

Dans `package.json`:

```
[...]  
"scripts": {  
  "dev": "node-sass -watch /src/assets/scss -o dist/css",  
  "build": "node-sass /src/assets/scss -o dist/css \  
    && postcss dist/css/*.css --replace"  
}  
[...]
```

Development

npm run dev

Production

npm run build

Aller plus loin

Tous les plugins POSTCSS

<https://github.com/postcss/postcss/blob/master/docs/plugins.md>