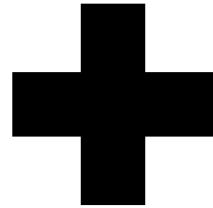


Nodejs 기반 업무 자동화

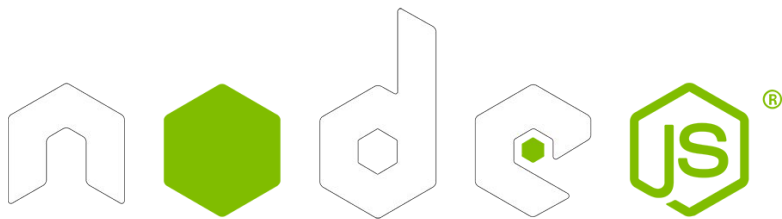
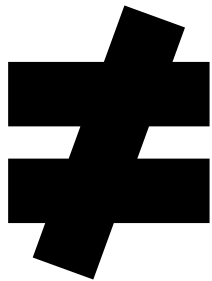
작성자 : 김광염

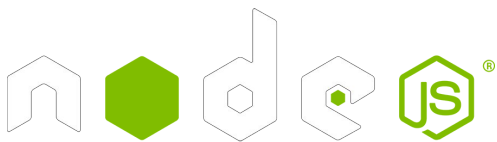
소속팀 / 상위부서 : UI개발 4팀

Sass



GRUNT





- npm을 이용하여 쉽게 동일한 환경 구축
- 대표적인 업무 자동화 툴인 Grunt와 Gulp의 기반 언어
- Javascript 기반으로 플러그인 개발 난이도가 낮은 편



Node-Sass

Node-sass is a library that provides binding for Node.js to libsass, the C version of the popular stylesheet preprocessor, Sass.

<https://github.com/sass/node-sass>

<https://github.com/sindresorhus/grunt-sass>

<https://github.com/dlmanning/gulp-sass>

```
$ npm install node-sass -D
```

```
$ npm install grunt-sass -D
```

```
$ npm install gulp-sass -D
```

Ruby Sass

Execution Time (2016-01-05 07:03:50 UTC)

loading tasks 323ms  16%

sass:dev 1.7s  84%

Total 2.1s

Node Sass

Execution Time (2016-01-05 07:00:09 UTC)

loading tasks 322ms  79%

sass:dev 86ms  21%

Total 408ms

Ruby Sass

sass:dev

1.7s



84%

Node Sass

sass:dev

86ms



21%

But

not include  compass

SpriteSmith

Convert images into spritesheets and coordinate maps.

<https://github.com/Enlighten/spritesmith>



```
$fork: (0px, 0px, 0px, 0px, 32px, 32px, 64px, 64px, 'spritesheet.  
handlebarsInheritance.png', 'fork', );  
$github: (32px, 0px, -32px, 0px, 32px, 32px, 64px, 64px,  
'spritesheet.handlebarsInheritance.png', 'github', );  
$twitter: (0px, 32px, 0px, -32px, 32px, 32px, 64px, 64px,  
'spritesheet.handlebarsInheritance.png', 'twitter', );  
$spritesheet-sprites: ($fork, $github, $twitter, );  
$spritesheet: (64px, 64px, 'spritesheet.handlebarsInheritance.png',  
$spritesheet-sprites, );  
  
/* ... */
```

Compass SASS Stylesheets

This is a repository to pull SASS style sheets on Bower, and enjoy the compass mixins by using libsass for faster compilation. This project makes minimal modifications to the original stylesheets from the original repository, intended to improve libsass compatibility and not change output.

<https://github.com/lgosuki/compass-mixins>

```
.pipe(sass({  
  outputStyle: 'expanded',  
  indentType: 'tab',  
  indentWidth: 1,  
  includePaths: 'node_modules/compass-mixins/lib'  
}))
```



PostCSS

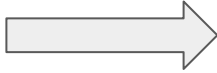
PostCSS is a tool for transforming styles with JS plugins. These plugins can lint your CSS, support variables and mixins, transpile future CSS syntax, inline images, and more.

<https://github.com/postcss/postcss>

PostCSS Plugins



Sass > css



Post process



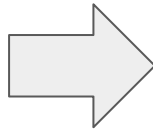


Autoprefixer

PostCSS plugin to parse CSS and add vendor prefixes to CSS rules using values from Can I Use.

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

```
:fullscreen a {  
  display: flex  
}
```



```
:webkit-full-screen a {  
  display: -webkit-box;  
  display: -webkit-flex;  
  display: flex  
}  
:-moz-full-screen a {  
  display: flex  
}  
:-ms-fullscreen a {  
  display: -ms-flexbox;  
  display: flex  
}  
:fullscreen a {  
  display: -webkit-box;  
  display: -webkit-flex;  
  display: -ms-flexbox;  
  display: flex  
}
```

CSS MQPacker

Pack same CSS media query rules into one media query rule.
Written with PostCSS.

<https://github.com/hail2u/node-css-mqpacker>

```
.foo::before {  
  content: "foo on small";  
}
```

```
@media screen and (min-width: 769px) {  
  .foo::before {  
    content: "foo on medium";  
  }  
}
```

```
.bar::before {  
  content: "bar on small";  
}
```

```
@media screen and (min-width: 769px) {  
  .bar::before {  
    content: "bar on medium";  
  }  
}
```

```
.foo::before {  
  content: "foo on small";  
}
```

```
.bar::before {  
  content: "bar on small";  
}
```

```
@media screen and (min-width: 769px) {  
  .foo::before {  
    content: "foo on medium";  
  }  
  .bar::before {  
    content: "bar on medium";  
  }  
}
```

doiuse

Lint CSS for browser support against caniuse database.

<https://github.com/anandthakker/doiuse>

```
.pipe(gulp.postcss([
  doiuse({
    browsers: ['ie >= 8', '> 1%'],
    ignore: ['rem'], // an optional array of features to ignore
    ignoreFiles: ['**/normalize.css'], // an optional array of file globs to match against original source
    file path, to ignore
    onFeatureUsage: function (usageInfo) {
      console.log(usageInfo.message)
    }
  })
])
```

/projects/website/main.css:5:3: CSS3 Box-sizing not supported by: IE (8,9,10,11), Chrome (36,37,38), Safari (8,7.1), Opera (24,25), iOS Safari (8,7.1,8.1), Android Browser (4.1,4.4,4.4.4), IE Mobile (10,11)

/projects/website/main.css:6:3: CSS3 Box-sizing not supported by: IE (8,9,10,11), Chrome (36,37,38), Safari (8,7.1), Opera (24,25), iOS Safari (8,7.1,8.1), Android Browser (4.1,4.4,4.4.4), IE Mobile (10,11)

/projects/website/main.css:8:3: CSS user-select: none not supported by: IE (8,9)

/projects/website/main.css:9:3: CSS user-select: none not supported by: IE (8,9)

/projects/website/main.css:10:3: CSS user-select: none not supported by: IE (8,9)

/projects/website/main.css:11:3: CSS user-select: none not supported by: IE (8,9)

/projects/website/main.css:12:3: CSS user-select: none not supported by: IE (8,9)

/projects/website/main.css:13:3: Pointer events not supported by: IE (8,9,10), Firefox (32,33), Chrome (36,37,38), Safari (8,7.1), Opera (24,25), iOS Safari (8,7.1,8.1), Android Browser (4.1,4.4,4.4.4), IE Mobile (10)

/projects/website/main.css:14:3: Pointer events not supported by: IE (8,9,10), Firefox (32,33), Chrome (36,37,38), Safari (8,7.1), Opera (24,25), iOS Safari (8,7.1,8.1), Android Browser (4.1,4.4,4.4.4), IE Mobile (10)

/projects/website/main.css:32:3: CSS3 Transforms not supported by: IE (8)

etc...

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

Packs

- `atcss` contains plugins that transform your CSS according to special annotation comments.
- `cssnano` contains plugins that optimize CSS size for use in production.
- `cssnext` contains plugins that allow you to use future CSS features today.
- `oldie` contains plugins that transform your CSS for older Internet Explorer compatibility.
- `precss` contains plugins that allow you to use Sass-like CSS.
- `rucksack` contains plugins to speed up CSS development with new features and shortcuts.
- `level4` contains only plugins that let you write CSS4 without the IE9 fallbacks.
- `short` adds and extends numerous shorthand properties.
- `stylelint` contains plugins that lint your stylesheets.

Future CSS Syntax

- `postcss-color-function` supports functions to transform colors.
- `postcss-color-gray` supports the `gray()` function.
- `postcss-color-hex-alpha` supports `#rrggbbaa` and `#rgba` notation.
- `postcss-color-hwb` transforms `hwb()` to widely compatible `rgb()`.
- `postcss-color-rebeccapurple` supports the `rebeccapurple` color.
- `postcss-conic-gradient` supports the `conic-gradient` background.
- `postcss-custom-media` supports custom aliases for media queries.
- `postcss-custom-properties` supports variables, using syntax from the W3C Custom Properties.
- `postcss-custom-selectors` adds custom aliases for selectors.
- `postcss-extend` supports spec-approximate `@extend` for rules and placeholders, recursively.
- `postcss-font-variant` transpiles human-readable `font-variant` to more widely supported CSS.
- `postcss-host` makes the Shadow DOM's `:host` selector work properly with pseudo-classes.
- `postcss-initial` supports `initial` keyword and `all: initial` to clean inherit styles.
- `postcss-logical-properties` transforms `start` and `end` properties to `left` and `right` depending on the writing direction of the document.
- `postcss-media-minmax` adds `<=` and `>=` statements to media queries.

CSScomb



CSScomb is a coding style formatter for CSS. You can easily write your own configuration to make your style sheets beautiful and consistent.

<https://github.com/csscomb/csscomb.js>

Build your own config

Choose the code sample you like more:

Remove empty rulesets

```
.a {  
  color: tomato;  
}
```

Leave code untouched

```
.a {  
  color: tomato;  
}  
.b{}
```

Option 1 of 24

Build your own config

Copy the config from here ↓ and save it as `.csscomb.json` in your project's dir:

```
{
  "block-indent": "\t",
  "color-shorthand": true,
  "quotes": "double",
  "space-before-colon": "",
  "space-after-colon": " ",
  "space-before-opening-brace": " ",
  "space-after-opening-brace": "\n",
  "space-after-selector-delimiter": " ",
  "space-before-selector-delimiter": "",
  "space-before-closing-brace": "\n",
  "strip-spaces": true,
  "tab-size": true,
  "unitless-zero": true
}
```

Thank You.