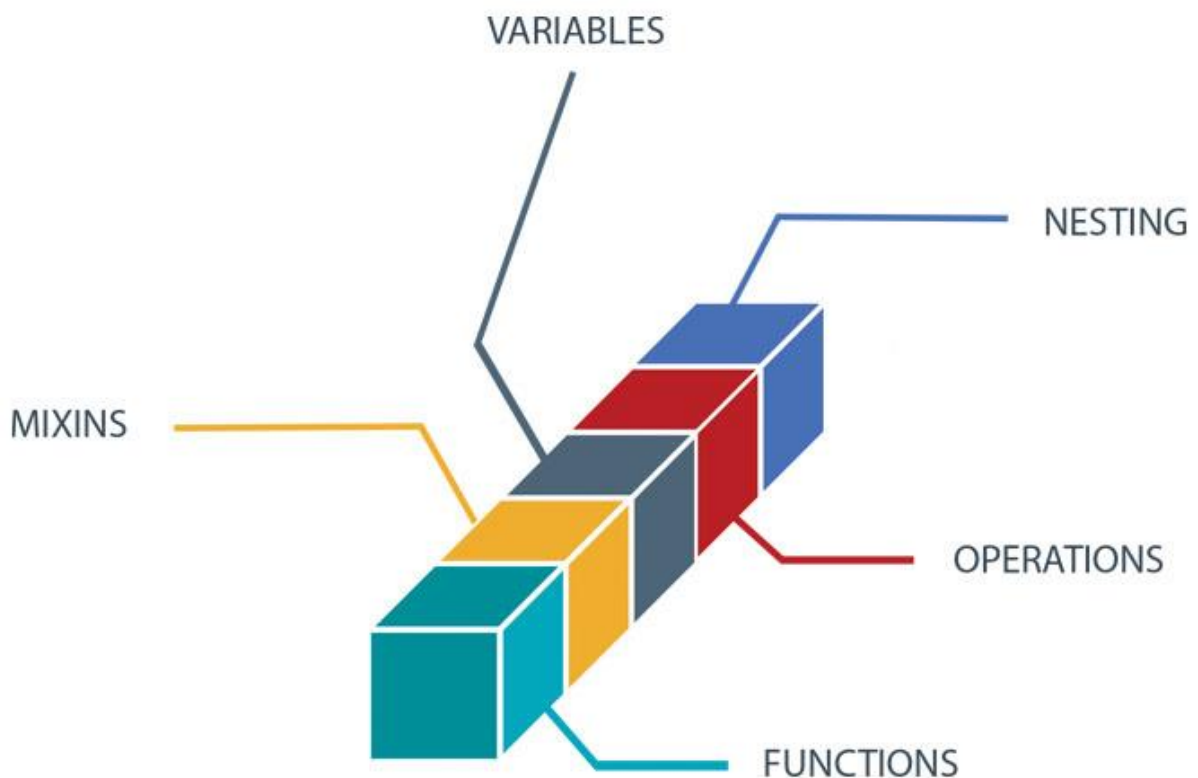# CSS preprocessors

What are CSS preprocessors, why do we use a CSS preprocessors, how to use a CSS preprocessor and which CSS preprocessors are being used by front-end development experts?

Bauke Roesink created this document on behalf of PSDtoWP.net. You are free to use this document on your website or blog.
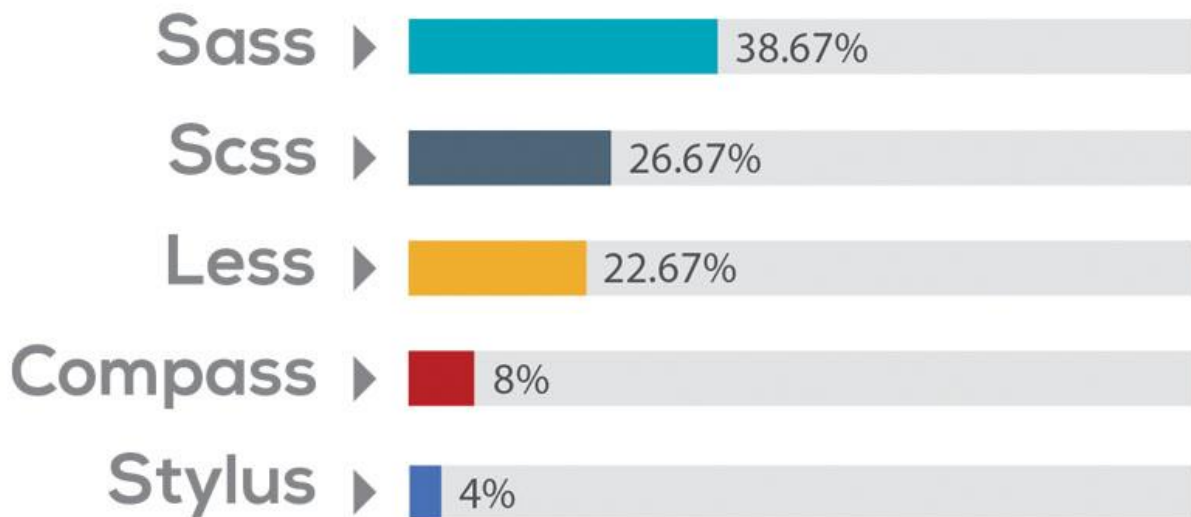
What are CSS preprocessors?

CSS preprocessors are also referred to as being 'dynamic style sheet languages'. They have been developed to add a programming functionality to the editing of Cascading Style Sheets. CSS Preprocessors convert code into a true CSS by taking the same written code from a simple preprocessed language (CSS with added extensions). Besides CSS preprocessors are used to add extensions which aren't used in CSS yet like: functions, mixins, nested rules, variables, operations and inheritance.

Common CSS preprocessors

Many websites can be found which contain lists of available CSS preprocessors. These lists might seem handy but honestly they aren't at all because when looking for a suitable CSS preprocessor they won't be of real use because you would still have to test them one by one which is a time-consuming exercise. To tackle this issue we have asked various front-end developers to inform us about their preferred CSS preprocessor and we have asked them to justify their CSS preprocessor choice.

Sass ▶ 38.67%

Scss ▶ 26.67%

Less ▶ 22.67%

Compass ▶ 8%

Stylus ▶ 4%

Created by PSDtoWP.net

Why use a CSS preprocessor?

More and more front-end developers get acknowledged with CSS preprocessors. Using CSS preprocessors has many advantages but also some disadvantages. According to the specific project we will opt for plain core vanilla CSS or for a CSS preprocessor.

Enhanced CSS

Duplication is discouraged

More flexible

Version control

DRY

Work in a team

Easy to work with

Deceiving file size

How to use CSS preprocessors

Many people prefer setting it up on their personal computer because this way a CSS file is created which is ready to be uploaded next to the SCSS file. It is important to take into account when working locally both the SCSS file and the corresponding CSS are kept together at all times. If you delete the CSS, you have to delete the SCSS file. The same goes for uploading or downloading them. Never perform an action for one file without performing the same action for the other file. (Besides when you would use git, don't forget to add the CSS files to .gitignore.)

Use your text editor

When working in a local environment your SCSS file should be watched at all times and you should use a tool to make sure that when you save your file, the SCSS is also turned into the corresponding CSS.

Doesn't your text editor support CSS preprocessors?

Tools like Dreamweaver, BBEdit or some other text editors don't have a so-called SCSS plugin or add-on. This shouldn't be too big an issue though because one could consider using a command line or one could opt for an app which watches your SCSS files at all times. Various applications are available to perform this task for you. The most popular and most commonly used ones are undeniably Koala (free), CodeKit ($29) and LiveReload ($9.99).