SASS vs. LESS – a comparison of the two preprocessors

In addition to SASS, LESS has also established itself in developer circles. This stylesheet language is strongly oriented around CSS and **resembles SCSS in its syntax**. Both preprocessors share some of the same properties: Both SASS and LESS allow for the use of mixins and variables. One difference, though, is that SASS is based on Ruby, while LESS uses JavaScript. But even this doesn’t give either of the preprocessors an advantage over the others.

The real differences are found instead in the **logical functions**: LESS gives users the opportunity to only activate mixins when specific situations occur. This is a helpful feature, but it represents the extent of logical links in LESS. SASS, on the other hand, offers loops and case distinctions as known from programming languages.

With SASS, users are free to choose between “indented syntax” or SCSS. Each developer can decide for themselves whether they would like to move away from the CSS rules or stay closer to the original. LESS doesn’t offer this choice. Here, users have to stick with the old rules. **Code in LESS is automatically a superset of CSS**: All source text formulated in CSS also functions in LESS – just like with SCSS.

**SASS is much more popular** among web designers. But this could be because SASS is a bit older. LESS was originally supported by the well-regarded frontend framework [Bootstrap](https://www.ionos.com/digitalguide/websites/web-development/bootstrap-alternatives/), which relied on the younger preprocessor. But with version 4, the project officially switched to SASS, which boosted the popularity of syntactically awesome style sheets even further.