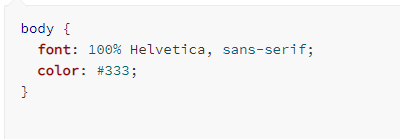
* SASS-SCSS:
* Variables
* Ta định nghĩa các biến (variable) như sau :



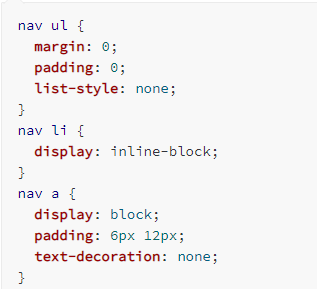
* SASS



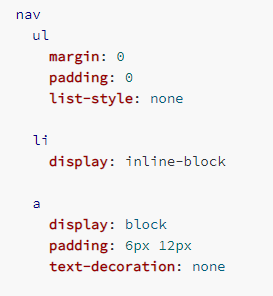
* SCSS



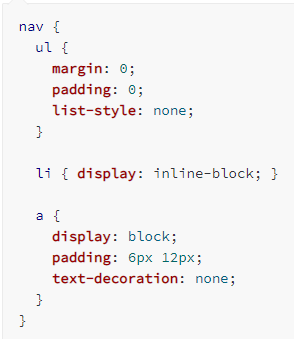
* Nesting
* CSS



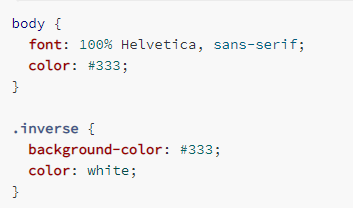
* SASS



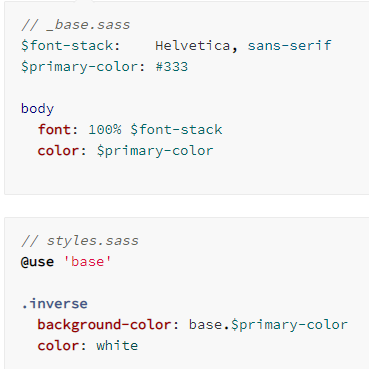
* SCSS



* Modules
* Ta có thể chia các files scss/sass thành các file nhỏ. Ta sẽ dùng @use
* CSS



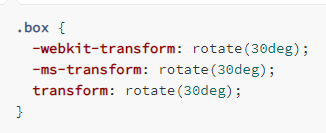
* SASS



* SCSS



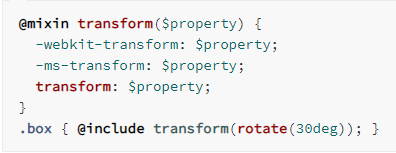
* Mixins
* Ta thường dùng mixins cho các vendor prefixes, 1 mixin cho phép ta tạo 1 nhóm css declaration mà ta muốn tái sử dụng trong site.
* Ta sẽ dùng từ khóa @mixin và theo sau là name ở đây là transform. Ta còn có dùng $property bên trong () sau đó ta có thể truyền vào transform. Sau đó ta có thể dùng CSS declaration bắt đầu với @include theo sau là tên của mixin.
* CSS :



* SASS :



* SCSS :



* Extend/Inheritance
* CSS



* SASS :



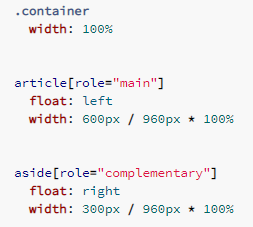
* SCSS :



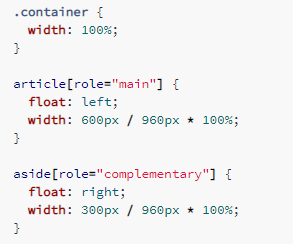
* OPERATORS :
* Hỗ trợ các phép toán :
* CSS :



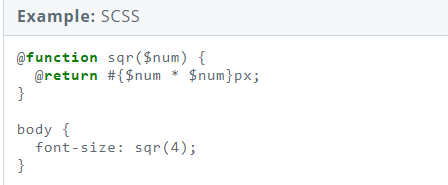
* SASS :



* SCSS :



Nhúng biến vào #{$var}px

* @mixin name(par1 = value1, par2 = value2) {}
* @include name(par1, par2)
* Giống như hàm @mixin định nghĩa hàm và gọi hàm bằng @include
* Value1 value2 là các value default nếu khi gọi hàm không truyền vào para.
* @function giống như @mixin nhưng khác ở :
* Ta dùng @function
* @function sẽ trả về 1 giá trị cho user
* @funtion không cần dùng @include
* 
* Hàm thường được dùng để tính giá trị.
* Sử dụng @extends
* 