

# THAM KHẢO NHANH SASS/SCSS

Biên, lồng nhau, mixin, hàm và luồng điều khiển

## Cú Pháp

### SCSS vs Sass

```
// SCSS (superset of CSS – uses braces)
.nav { display: flex; }

// Sass (indented – no braces or semicolons)
.nav
  display: flex
```

SCSS là cú pháp được dùng phổ biến nhất

### So Sánh

**SCSS (.scss)** Tương thích CSS, dùng dấu ngoặc & chấm phẩy

**Sass (.sass)** Dựa trên thụt lề, không có dấu ngoặc

**Đầu ra** Cả hai đều biên dịch ra CSS chuẩn

**Khuyến nghị** SCSS (phổ biến hơn, dễ chuyển đổi hơn)

## Biên

### Định Nghĩa & Sử Dụng

```
$primary: #3498db;
$spacing: 16px;
$font-stack: "Helvetica", Arial, sans-serif;

.btn {
  color: $primary;
  padding: $spacing;
  font-family: $font-stack;
}
```

### Phạm Vi Biên

```
$color: red; // global
.card {
  $color: blue; // local to .card
  color: $color; // blue
}
.other { color: $color; } // red
```

## Cờ

**!default** Chỉ gán nếu chưa được định nghĩa

**!global** Đưa biến cục bộ lên phạm vi toàn cục

## Lồng Nhau

### Lồng Selector

```
.nav {
  ul { list-style: none; }
  li { display: inline-block; }
  a { text-decoration: none; }
  &:hover { color: blue; } // & = parent selector
}
```

### Selector Cha (&)

```
.btn {
  &-primary { background: blue; } // BEM: .btn-primary
  &_icon { margin-right: 4px; } // BEM: .btn_icon
  &.dark { color: white; } // .dark .btn
}
```

## Lồng Thuộc Tính

```
.box {
  border: { width: 1px; style: solid; color: #ccc; }
  // compiles to: border-width, border-style, border-color
}
```

## Mixin

### Định Nghĩa & Sử Dụng

```
@mixin flex-center($direction: row) {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: $direction;
}

.hero { @include flex-center(column); }
```

### Khối Nội Dung

```
@mixin responsive($breakpoint) {
  @media (min-width: $breakpoint) { @content; }
}

.sidebar {
  width: 100%;
  @include responsive(768px) { width: 300px; }
}
```

### Tính Năng Mixin

**@mixin name(\$args)** Định nghĩa khối style tái sử dụng

**@include name()** Sử dụng mixin

**Tham số mặc định** \$arg: value cho tham số tùy chọn

**\$args...** Tham số biến đổi (rest params)

**@content** Chèn khối nội dung từ caller

## Hàm

### Hàm Tùy Chỉnh

```
@function rem($px, $base: 16) {
  @return math.div($px, $base) * 1rem;
}

.title { font-size: rem(24); } // 1.5rem
```

### Hàm Có Sẵn

**darken(\$color, 10%)** Làm tối màu

**lighten(\$color, 10%)** Làm sáng màu

**mix(\$c1, \$c2, 50%)** Pha trộn hai màu

**rgba(\$color, 0.5)** Đặt kênh alpha

**math.div(\$a, \$b)** Phép chia (thay thế '/')

**math.round(\$n)** Làm tròn số

**string.quote(\$s)** Thêm dấu nháy vào string

**if(\$cond, \$t, \$f)** Điều kiện nội tuyến

## Extend

### Extend & Placeholder

```
%flex-center { // placeholder – not emitted
  display: flex;
  justify-content: center;
  align-items: center;
}
```

```
.hero { @extend %flex-center; }
.modal { @extend %flex-center; }
// Both share one CSS rule via selector grouping
```

## Extend vs Mixin

**@extend** Nhóm selector — CSS đầu ra nhỏ hơn

**@mixin** Sao chép khai báo — hỗ trợ tham số

**% placeholder** Chỉ dùng để extend (không xuất nếu không dùng)

**Khuyến nghị** Ưu tiên mixin cho style có tham số

## Partial & Import

### Tổ Chức File

```
// variables.scss (partial – not compiled alone)
$primary: #3498db;

// main.scss
@use "variables"; // modern: namespaced
.btn { color: variables.$primary; }

@use "variables" as v; // alias
.btn { color: v.$primary; }
```

### Hệ Thống Module

**@use 'file'** Nạp module với namespace

**@use 'file' as \*** Nạp không có namespace

**@use 'file' as alias** Namespace tùy chỉnh

**@forward 'file'** Tái xuất các thành viên module

**\_partial.scss** File không được biên dịch độc lập

@import đã bị loại bỏ — dùng @use và @forward thay thế

## Luồng Điều Khiển

### Điều Kiện

```
@mixin theme($mode) {
  @if $mode == dark {
    background: #333; color: #fff;
  } @else {
    background: #fff; color: #333;
  }
}
```

### Vòng Lặp

```
@for $i from 1 through 4 {
  .col-#{$i} { width: 25% * $i; }
}

@each $name, $color in (primary: blue, danger: red) {
  .text-#{$name} { color: $color; }
}
```

### Chỉ Thị

**@if / @else if / @else** Logic điều kiện

**@for \$i from a through b** Vòng lặp số (bao gồm cả hai đầu)

**@for \$i from a to b** Vòng lặp số (không bao gồm đầu cuối)

**@each \$item in \$list** Duyệt list hoặc map

**@while** Lặp khi điều kiện đúng

**#{var:}** Nội suy trong selector/thuộc tính

## Map & List

### Map

```
$colors: (primary: #3498db, danger: #e74c3c, success: #2ecc71);

.alert { color: map.get($colors, danger); }

@each $name, $color in $colors {
  .bg-#{$name} { background: $color; }
}
```

### List

```
$sizes: 8px 16px 24px 32px;
.box { padding: list.nth($sizes, 2); } // 16px
```

### Hàm Map & List

**map.get(\$map, \$key)** Lấy giá trị theo key

**map.merge(\$m1, \$m2)** Hợp nhất hai map

**map.keys(\$map)** Danh sách tất cả key

**map.has-key(\$map, \$key)** Kiểm tra key tồn tại

**list.nth(\$list, \$n)** Lấy phần tử tại vị trí (bắt đầu từ 1)

**list.length(\$list)** Số lượng phần tử

**list.append(\$list, \$val)** Thêm phần tử vào list

## Mẫu Phổ Biến

### Breakpoint Responsive

```
$breakpoints: (sm: 576px, md: 768px, lg: 992px, xl: 1200px);

@mixin bp($name) {
  @media (min-width: map.get($breakpoints, $name)) {
    @content;
  }
}

.sidebar { width: 100%; @include bp(md) { width: 300px; } }
```

### Tạo Lớp Tiện Ích

```
$spaces: (0: 0, 1: 4px, 2: 8px, 3: 16px, 4: 32px);

@each $key, $val in $spaces {
  .m-#{$key} { margin-top: $val; }
  .mb-#{$key} { margin-bottom: $val; }
  .p-#{$key} { padding: $val; }
}
```

### Chế Độ Tối

```
@mixin dark { @media (prefers-color-scheme: dark) { @content; } }

body {
  background: #fff;
  @include dark { background: #1a1a1a; color: #eee; }
}
```