

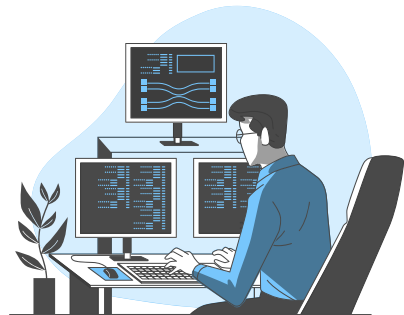
B__E--M

Sass



Khóa học Frontend

Bài 11: Học BEM, SASS



Nội dung

01

BEM

...

02

SASS

...

03

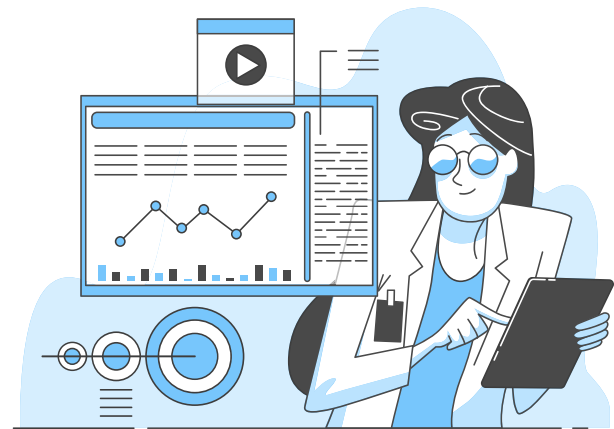
Giới thiệu Project 2
- Sử dụng PUG, SASS

...

04

Chuyển Project 1 từ HTML5,
CSS3 thành PUG, SCSS

...



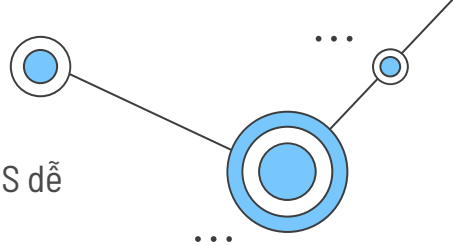


01

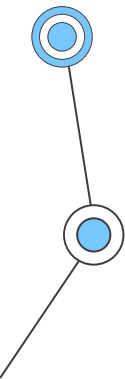
BEM



01. BEM



- **BEM (Block, Element, Modifier)** là một **phương pháp đặt tên lớp** CSS để tạo ra mã CSS dễ đọc, tái sử dụng và bảo trì.
- **Block** (Khối)
 - Đại diện cho một phần tử độc lập mà không phụ thuộc vào bất kỳ phần tử nào khác trên trang.
 - Đặt tên lớp của block dưới dạng: **block-name**
- **Element** (Phần tử)
 - Là một phần tử con của block và chỉ có ý nghĩa khi nằm trong block đó.
 - Đặt tên lớp của phần tử dưới dạng: block-name__**element-name**
- **Modifier** (Sửa đổi)
 - Mô tả trạng thái hoặc biến thể của block hoặc phần tử.
 - Đặt tên lớp modifier dưới dạng: block-name__element-name--**modifier-name**





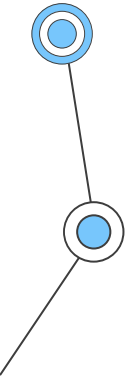
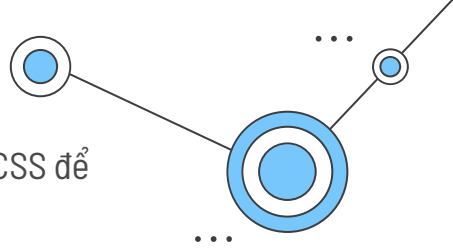
02

SASS



2.1. Khái niệm

- **SASS** (Syntactically **A**wesome **S**tyle **S**heets) là **bộ tiền xử lý CSS**, biên dịch mã thành CSS để giúp **viết CSS nhanh hơn**, tái sử dụng được, tối ưu hơn.
- **SCSS** (Sassy **CSS**) được giới thiệu trong **phiên bản thứ 3** của SASS, nó bổ sung thêm dấu **ngoặc { }** và dấu **chấm phẩy ;**
- Xem sự khác biệt tại: <https://sass-lang.com/guide>

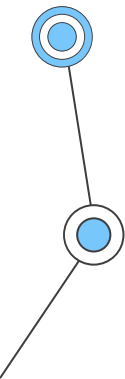
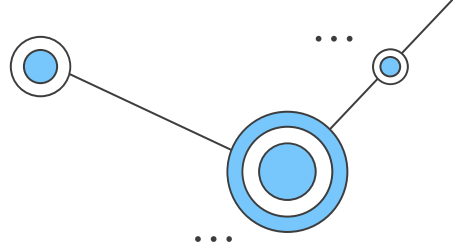


2.2. Hướng dẫn cài đặt

- Các bước cài đặt:
 - Bước 1: Chạy **npm init** để cài NPM cho project (Nếu chưa có)
 - Bước 2: Chạy lệnh **npm install -g sass** để cài SASS cho project
 - Bước 3: Trong folder assets, tạo folder **scss** và tạo file **style.scss**
 - Bước 4: Sửa file **package.json**

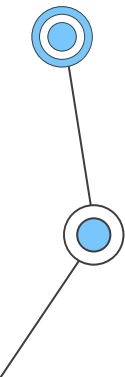
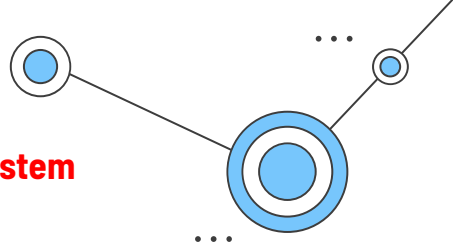
```
{
  "scripts": {
    "start": "sass --watch assets/scss:assets/css",
  }
}
```

- Bước 5: Chạy **npm start**



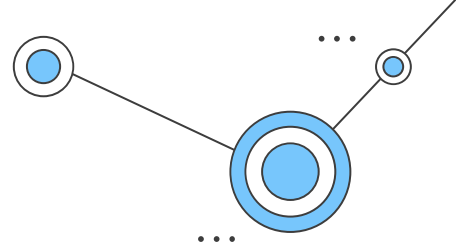
2.2. Hướng dẫn cài đặt

- Lưu ý: Nếu gặp lỗi **cannot be loaded because running scripts is disabled on this system**
 - Bước 1: Mở **Windows PowerShell**
 - Bước 2: Chạy câu lệnh:
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser
 - Bước 3: Sau đó chạy lại bước 5 (bên trên)



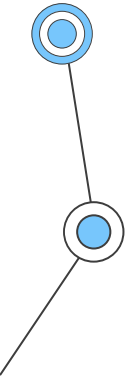
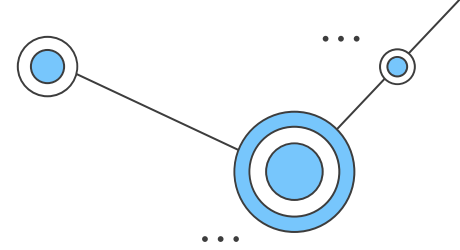
2.3. Các cú pháp SCSS

- Cú pháp **@extend**
 - Dùng để kế thừa lại các thuộc tính của một bộ chọn CSS khác.
 - *Hướng dẫn trong buổi học.*



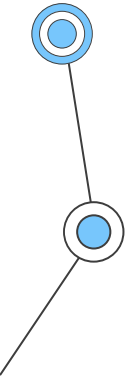
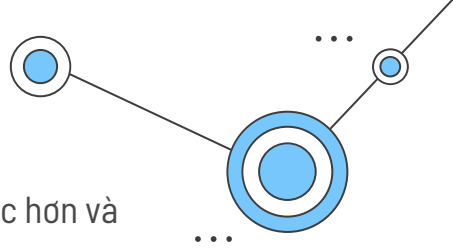
2.3. Các cú pháp SCSS

- **Variable** (Biến)
 - Cú pháp: **\$name : value;**
 - *Hướng dẫn trong buổi học.*



2.3. Các cú pháp SCSS

- **Nesting** (Phân cấp)
 - Nesting để lồng các bộ chọn CSS bên trong nhau, giúp mã nguồn trở nên dễ đọc hơn và giảm thiểu lặp code.
 - *Hướng dẫn trong buổi học.*



2.3. Các cú pháp SCSS

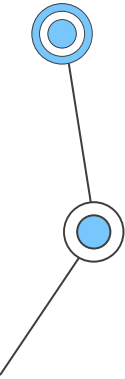
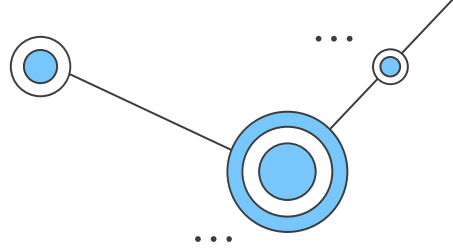
- **@mixin** và **@include**

- **@mixin** dùng để định nghĩa một khối CSS có thể **truyền giá trị vào được**.
- Cú pháp:

```
@mixin ten_mixin(tham_so_1, tham_so_2,...) {  
  // Code CSS  
}
```

- **@include** dùng để gọi @mixin.
- Cú pháp:

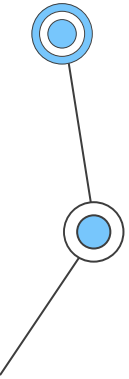
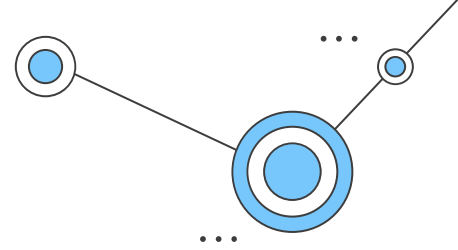
```
.selector {  
  @include ten_mixin(loi_so_1, loi_so_2,...);  
}
```



2.3. Các cú pháp SCSS

- Câu lệnh **@if @else**
 - Cú pháp:

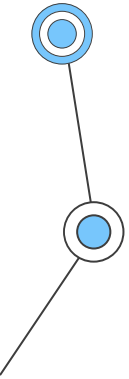
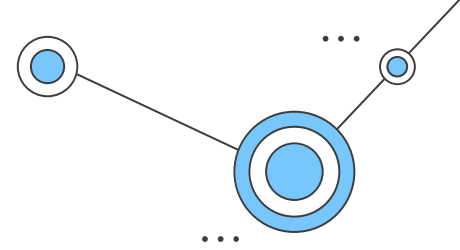
```
@if ($dieu_kien_1) {  
  // Code  
}  
@else if ($dieu_kien_2) {  
  // Code  
}  
@else {  
  // Code  
}
```



2.3. Các cú pháp SCSS

- Vòng lặp **@for**
 - Cú pháp:

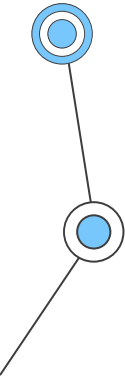
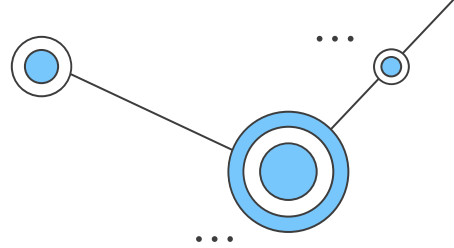
```
@for $ten_bien from $gia_tri_bat_dau through $gia_tri_ket_thuc {  
  // Code  
}
```



2.3. Các cú pháp SCSS

- Vòng lặp **@while**
 - Cú pháp:

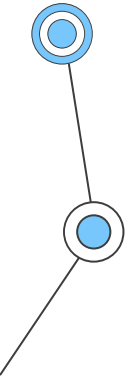
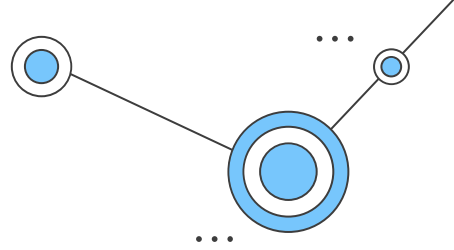
```
@while dieu_kien {  
  // Code  
}
```



2.3. Các cú pháp SCSS

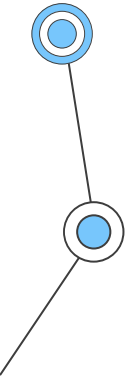
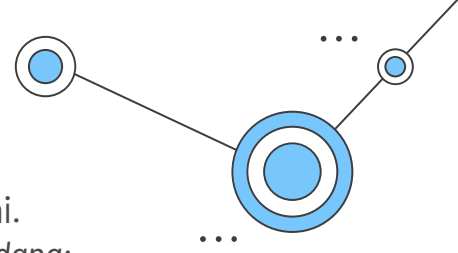
- Vòng lặp **@each**
 - Dùng để lặp qua một danh sách các biến.
 - Cú pháp:

```
$list : item1, item2, item3,...;  
@each $item in $list {  
  // Code  
}
```



2.3. Các cú pháp SCSS

- **@import** (Nhúng file)
 - @import dùng để nhúng nội dung của một file SCSS khác vào file SCSS hiện tại.
 - **Lưu ý:** Nếu file scss nào đó không cần biên dịch thành file css thì đặt tên file có dạng: ***_ten-file.scss***

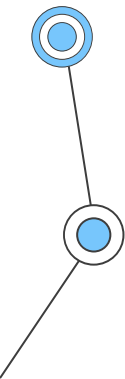
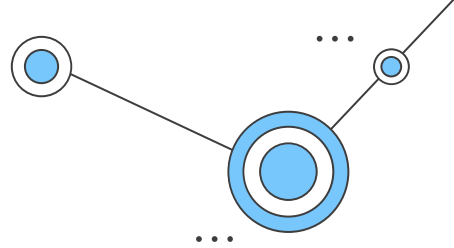


03

Giới thiệu Project 2 - Sử dụng PUG, SASS

03. Giới thiệu Project 2 - Sử dụng PUG, SASS

- Chuyển Project 1 từ HTML5, CSS3 thành PUG, SCSS
- Cấu giao diện trang Danh sách sản phẩm
- Cấu giao diện trang Chi tiết sản phẩm
- Link Figma của Project 1: *Đã gửi ở bài 7.*

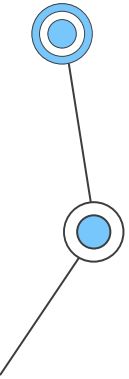
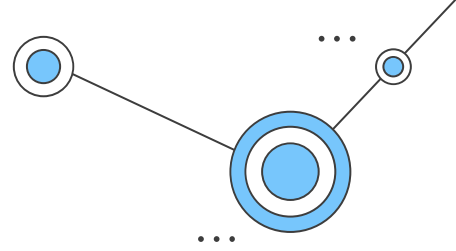


04

**Chuyển Project 1
từ HTML5, CSS3
thành PUG, SCSS**

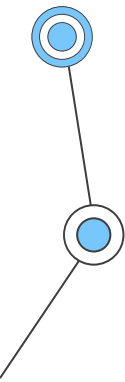
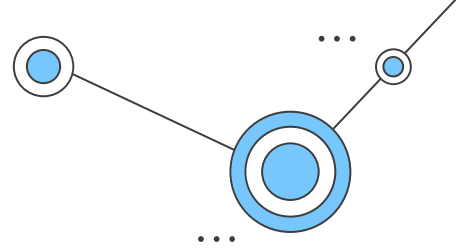
4.1. Cấu hình chung

- *Hướng dẫn trong buổi học.*



4.2. Cấu hình SCSS base

- *Hướng dẫn trong buổi học.*



4.3. Chuyển Header

- *Hướng dẫn trong buổi học.*

