Tổng quan

Magento 2 kết hợp [LESS](http://lesscss.org/) , một bộ xử lý trước CSS đơn giản hóa việc quản lý các tệp CSS phức tạp. Để xác định phong cách của một cửa hàng Magento, bạn có thể sử dụng cả hai - stylesheet CSS và LESS.

Ứng dụng Magento cung cấp thư viện LESS UI được tích hợp sẵn, bạn có thể mở rộng tùy chọn.

Để tùy chỉnh kiểu cửa hàng , bạn cần phải [tạo chủ đề thiết kế tùy chỉnh](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-create.html) . Sau đó, bạn có thể sử dụng một trong các cách tiếp cận sau:

* Nếu chủ đề của bạn [kế thừa](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-inherit.html) từ chủ đề Trống hoặc Luma của Magento, bạn có thể ghi đè lên các tệp LESS mặc định; ví dụ để [thay đổi giá trị của các biến](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/theme-ui-lib.html#fedg_using-ui-lib_predef-vars) được sử dụng trong các tệp mặc định.
* Tạo các tệp LESS của riêng bạn bằng cách sử dụng bộ tiền xử lý LESS dựng sẵn.
* Tạo các tệp CSS của riêng bạn, tùy chọn đã biên dịch chúng bằng cách sử dụng bộ tiền xử lý CSS của bên thứ ba.

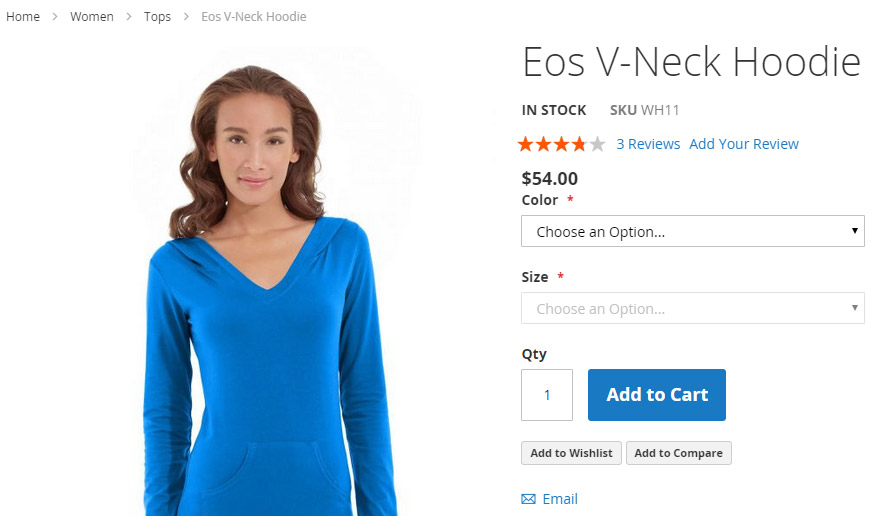
Những điều cần nhớ khi làm việc với các kiểu

Đảm bảo rằng bạn [đã đặt](https://devdocs.magento.com/guides/v2.2/config-guide/cli/config-cli-subcommands-mode.html) ứng dụng Magento của mình thành nhà phát triển hoặc [chế độ](https://devdocs.magento.com/guides/v2.2/config-guide/bootstrap/magento-modes.html) mặc định .

Nếu thay đổi kiểu của bạn không áp dụng sau khi làm mới trang, việc xóa bộ nhớ cache của tệp tĩnh có thể hữu ích. Xem [chủ đề Bộ nhớ cache tệp tĩnh](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/cache_for_frontdevs.html#clean_static_cache) để biết hướng dẫn cách thực hiện việc này.

Thay đổi kiểu: hướng dẫn

Đây là một minh họa đơn giản về việc thay đổi kiểu dáng bằng cách sử dụng cách tiếp cận đầu tiên: thay đổi màu sắc của các nút của một lớp nhất định. Trong chủ đề Trống, các nút của .action.primarylớp, được gọi là *các* nút *chính* , có màu xanh dương. Hình ảnh sau đây minh họa điều này:



OrangeCo muốn thay đổi màu của các nút chính thành màu cam. Để đạt được điều này, họ làm như sau:

1. Tạo chủ đề màu cam mới, được kế thừa từ chủ đề Trống .
2. Trong thư mục chủ đề màu cam, hãy thêm app/design/frontend/OrangeCo/orange/web/css/source/\_theme.lesstệp ghi đè bằng mã sau:
3. // Primary button
4. @button-primary\_\_color: @color-white;
5. @button-primary\_\_hover\_\_color: @color-white;
6. @button-primary\_\_background: @color-orange-red1;
7. @button-primary\_\_hover\_\_background: @color-orange-red4;
8. @button-primary\_\_border: 1px solid @color-orange-red2;
9. @button-primary\_\_hover\_\_border: 1px solid @color-orange-red2;

Khi OrangeCo [áp dụng chủ đề của chúng](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-apply.html) , các nút chính sẽ trông giống như hình dưới đây:



## Include CSS

Trong ứng dụng Magento, các tệp CSS được bao gồm trong các layout.

Về mặt kỹ thuật, có một tùy chọn để đưa chúng vào tệp mẫu, nhưng chúng tôi khuyên bạn nên tránh điều này.

Tên lớp CSS có thể được chỉ định trong cả template và layout.

Chủ đề này mô tả cách các bản định kiểu được đặt theo mặc định trong hệ thống tệp ứng dụng Magento và cách được khuyến nghị để bao gồm các tệp CSS trong bố cục.

Cách sắp xếp tệp biểu định kiểu Magento

Thông thường, các tệp CSS và LESS chỉ được lưu trữ trong các chủ đề. Thư mục mô-đun không chứa bất kỳ kiểu mặc định nào.

Trong [thư mục chủ đề](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-structure.html) , bảng định kiểu được lưu trữ ở các vị trí sau:

|  |  |
| --- | --- |
| **Thư mục, liên quan đến <theme\_dir>** | **Sự miêu tả** |
| /<Namespace>\_<Module>/web/css | Kiểu mô-đun cụ thể |
| /web/css | Chứa như sau:   * print.less: được sử dụng để tạo kiểu cho phiên bản in của trang cửa hàng. * \_styles.less- tệp tổng hợp, bao gồm tất cả các tệp LESS được sử dụng trong chủ đề . Dấu gạch dưới ("\_") trong tên tệp thông thường có nghĩa là một tệp không được sử dụng độc lập, nhưng được bao gồm trong các tệp khác. * styles-m.less: được sử dụng để tạo kiểu dành riêng cho thiết bị di động, bao gồm \_styles.less * styles-l.less: được sử dụng để tạo kiểu dành riêng cho máy tính để bàn, bao gồm \_styles.less. * /source: thư mục con này chứa các tệp cấu hình LESS gọi các mixin từ thư viện giao diện người dùng Magento * /source/\_theme.less: ghi đè các giá trị biến thư viện giao diện người dùng mặc định của Magento. |

Include CSS

Trong ứng dụng Magento, cách được đề xuất để bao gồm các bảng định kiểu là chỉ định chúng trong các tệp bố cục .

Thông thường, các bảng định kiểu bạn đưa vào sẽ có sẵn cho tất cả các trang cửa hàng. Để đạt được điều này, bao gồm CSS của bạn trong default\_head\_blocks.xmlcác Magento\_Theme module, trong đó xác định mặc định <head>phần trang cho tất cả các trang Magento. Cách được khuyến nghị để thực hiện việc này là thêm [mở rộng](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/layouts/layout-extend.html) default\_head\_blocks.xml trong chủ đề của bạn và bao gồm các bảng định kiểu được yêu cầu trong tệp này.

Tùy chỉnh của bạn default\_head\_blocks.xmlnên được đặt như sau:

<theme\_dir>/Magento\_Theme/layout/default\_head\_blocks.xml.

Để bao gồm tệp CSS, hãy thêm <css src="<path>/<file>" media="print|<option>"/>khối trong <head>phần trong tệp bố cục. <path>được chỉ định tương đối so với thư mục web chủ đề ( <theme\_dir>/web)

Ví dụ, phần sau minh họa cách các bản định kiểu được bao gồm trong chủ đề trống mặc định:

[ /Magento\_Theme/layout/default\_head\_blocks.xml]

<page xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="urn:magento:framework:View/Layout/etc/page\_configuration.xsd">

<head>

<css src="css/styles-m.css" />

<css src="css/styles-l.css" media="screen and (min-width: 768px)"/>

<css src="css/print.css" media="print" />

</head>

</page>

Nếu hệ thống không tìm thấy tệp CSS được bao gồm, nó sẽ tìm kiếm cùng tên tệp với .lessphần mở

## CSS và LESS được xử lý trước và cách gỡ lỗi chúng

Chủ đề mô tả cách các bảng định kiểu được xử lý trước và được biên dịch thành CSS trong ứng dụng Magento. Nó cung cấp nền tảng lý thuyết mà một nhà phát triển giao diện người dùng cần phải gỡ rối các bảng định kiểu một cách hiệu quả.

## Điều khoản sử dụng

|  |  |
| --- | --- |
| **Kỳ hạn** | **Sự miêu tả** |
| Tệp nguồn gốc | Các .lesstệp mà từ đó các .csstệp được [bao gồm trong bố cục](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-themes.html) được biên dịch. Ví dụ, trong một trong các [tệp bố cục của chủ đề Magento Blank](https://github.com/magento/magento2/blob/2.2/app/design/frontend/Magento/blank/Magento_Theme/layout/default_head_blocks.xml) , các .csstệp sau được bao gồm trong head:  <head>  <css src = "css / kiểu-m.css" />  <css src = "css / styles-l.css" media = "màn hình và (min-width: 768px)" />  <css src = "css / print.css" phương tiện = "in" />  </ head>  Các tệp nguồn gốc cho chủ đề Trống:   * [Magento\_Blank\_theme\_dir / web / css / styles-m.less](https://github.com/magento/magento2/blob/2.0/app/design/frontend/Magento/blank/web/css/styles-m.less) * [Magento\_Blank\_theme\_dir / web / css / styles-l.less](https://github.com/magento/magento2/blob/2.0/app/design/frontend/Magento/blank/web/css/styles-l.less) * [Magento\_Blank\_theme\_dir / web / css / print.less](https://github.com/magento/magento2/blob/2.0/app/design/frontend/Magento/blank/web/css/print.less) |

## Chế độ biên dịch LESS

Trong ứng dụng Magento, các chế độ biên dịch .lesstệp CSS sau đây được triển khai:

1. Biên dịch LESS phía máy chủ.

Đây là chế độ biên dịch mặc định và là tùy chọn duy nhất trong [chế độ ứng dụng sản xuất](https://devdocs.magento.com/guides/v2.2/config-guide/bootstrap/magento-modes.html#production-mode) . Trong trường hợp này việc biên dịch được thực hiện trên máy chủ, sử dụng [thư viện LESS PHP](https://github.com/oyejorge/less.php) .

1. Biên dịch LESS phía máy khách.

Khi ứng dụng của bạn không ở chế độ sản xuất, bạn có thể đặt Magento để biên dịch .lesscác tệp trong trình duyệt, sử dụng [thư viện gốcless.js](http://lesscss.org/usage/#using-less-in-the-browser)

Để đặt chế độ biên dịch, hãy làm như sau:

1. Trong Magento Admin, điều hướng đến **Cửa hàng** > **Cấu hình** > ADVANCED> **Nhà phát triển** .
2. Trong trường thả xuống **Chế độ xem cửa hàng** , chọn **Cấu hình mặc định** .
3. Trong **Luồng công việc phát triển Front-end** , trong trường **loại luồng công việc** , hãy chọn chế độ biên dịch.
4. Để lưu cài đặt, hãy nhấp vào **Lưu cấu hình** .

### Biên soạn LESS phía máy chủ

Đoạn sau mô tả cách thức xử lý tiền xử lý LESS hoạt động ở chế độ biên dịch phía máy chủ. Đối với mỗi tệp CSS được bao gồm trong các bố trí, bộ tiền xử lý LESS thực hiện các thao tác sau:

1. Kiểm tra xem .csstập tin được yêu cầu có được tìm thấy không. Nếu nó được tìm thấy, thì bộ tiền xử lý sẽ ngừng thực thi nó. Nếu không, nó sẽ chuyển sang bước tiếp theo.
2. Thay đổi phần mở rộng của tập tin được yêu cầu .lessvà cố gắng tìm tệp bằng [cơ chế dự phòng Magento](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-inherit.html#theme-inherit-static) . Nếu .lesskhông tìm thấy tập tin, bộ tiền xử lý LESS sẽ ngừng thực thi nó. Nếu không, nó sẽ chuyển sang bước tiếp theo.
3. Đọc .lessnội dung tệp và giải quyết [@magento\_import](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-preprocess.html#fedg_css-magento-import)và @importchỉ thị LESS mặc định .
4. Giải quyết tất cả các đường dẫn trong .lesscác tệp đến các đường dẫn tương đối trong hệ thống bằng cách sử dụng cơ chế dự phòng Magento. Tất cả các tệp được xử lý bởi bộ tiền xử lý LESS được sao chép vào var/view\_preprocessed/less. Các tệp đã nhập được xử lý đệ quy.
5. Tất cả các tệp nguồn được chuyển tới trình biên dịch PHP LESS. Các .csstập tin kết quả được biên dịch được xuất bản lên pub/static/frontend/<Vendor>/<theme>/<locale>.

#### Gỡ lỗi kiểu ở chế độ biên dịch phía máy chủ

Ở chế độ biên dịch LESS phía máy chủ, để áp dụng các thay đổi của bạn, bạn cần thực hiện các thao tác sau:

1. Xóa pub/static/frontend/<Vendor>/<theme>/<locale>bằng cách xóa thư mục trong hệ thống tệp.
2. Xóa các thư mục var/cachevà var/view\_preprocessedbằng cách xóa thư mục trong hệ thống tập tin. (nếu chúng đã tồn tại ở đó).
3. Kích hoạt các tập tin và biên dịch tĩnh . Điều này có thể được thực hiện theo một trong các cách sau:
   * Tải lại trang nơi áp dụng các kiểu sửa đổi.
   * Chạy [công cụ triển khai tệp tĩnh](https://devdocs.magento.com/guides/v2.2/config-guide/cli/config-cli-subcommands-static-view.html) .

Tải lại trang chỉ kích hoạt biên dịch và xuất bản các kiểu được sử dụng trên trang này, và không cung cấp cho bạn thông tin về các lỗi nếu có. Vì vậy, nếu bạn đã thực hiện thay đổi trong .lesscác tệp được sử dụng trên nhiều trang và muốn gỡ lỗi chúng, việc sử dụng công cụ triển khai là tùy chọn tốt hơn.

##### Gỡ lỗi bằng cách sử dụng công cụ triển khai tệp xem tĩnh

Một khi bạn lưu các thay đổi của bạn và chạy magento setup:static-content:deploylệnh cli từ <your\_Magento\_instance>/binthư mục, công cụ tiền xử lý (bao gồm biên dịch) và xuất bản các tệp xem tĩnh.

Tất cả các lỗi xảy ra trong quá trình .lessbiên dịch tập tin được xử lý bởi [oyejorge/less.php](https://github.com/oyejorge/less.php)thư viện của bên thứ ba.

Lỗi được xem là ngoại lệ và được ghi vào nhật ký hệ thống (theo mặc định var/log/system.log) và hiển thị trên màn hình. Đối với mỗi lỗi, thông tin sau được viết:

* Đường dẫn đến tệp được xử lý trong var/view\_preprocessedthư mục.
* Mô tả lỗi, bao gồm đường dẫn đến tệp nơi xảy ra lỗi thực tế. Nó có thể là tệp được xử lý hoặc tệp được nhập.
* Dòng lỗi và số cột.
* Nội dung của .lessmã trong các dòng trước và sau.

Ví dụ về thông báo lỗi:

Compilation from source: /var/www/magento2/app/design/adminhtml/Magento/backend/web/css/styles.less

variable @variable-x is undefined in file /var/www/magento2/var/view\_preprocessed/css/adminhtml/Magento/backend/en\_US/css/styles.less in styles.less on line 56, column 17

margin-left: 0;

width: 100%;

height: @variable-x;

}

.menu-wrapper,

##### Gỡ lỗi bằng cách sử dụng Grunt

Ngoài ra, để đơn giản hóa quá trình áp dụng và gỡ lỗi các tùy chỉnh kiểu, ở chế độ biên dịch phía máy chủ, bạn có thể

Xem phần [Biên dịch LESS với](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css_debug.html) chủ đề [Grunt](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css_debug.html) để biết chi tiết cách cài đặt, cấu hình và sử dụng Grunt.

### Biên dịch LESS phía máy khách

Luồng biên dịch phía máy khách tương tự như phía máy chủ. Sự khác biệt nằm trong tập hợp các tệp, được xuất bản pub/staticở [bước cuối cùng](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-preprocess.html#compile_last) . Ở chế độ phía máy khách, các tệp sau được xuất bản vào pub/static/frontend/<Vendor>/<theme>/<locale>thư mục:

* các tệp gốc (.less) với @magento\_importchỉ thị được giải quyết
* [các liên kết tượng trưng](http://en.wikipedia.org/wiki/Symbolic_link) đến tệp nguồn gốc không chứa@magento\_import
* liên kết tượng trưng cho tất cả 'file .less` khác nhập khẩu đệ quy bằng @magento\_importvà @importchỉ thị

Liên kết tượng trưng không được tạo và bản sao tệp được xử lý được xuất bản để pub/staticthay thế, nếu tệp nguồn khác với tệp được xử lý. Một trong những lý do của sự khác biệt này có thể là việc sử dụng @importchỉ thị không có phần mở rộng tệp trong tệp nguồn. Xem [Cách sử dụng chỉ thị @import](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-preprocess.html#fedg_css-import) để biết thêm chi tiết.

#### Gỡ lỗi kiểu trong chế độ biên dịch phía máy khách

Biên dịch LESS phía máy khách được triển khai bằng less.jsthư viện gốc . Cấu hình mặc định được đặt trong lib/web/less/config.less.js; bạn có thể thay đổi nó nếu cần.

Bạn có thể tìm thấy thông tin chi tiết về cấu hình và các tùy chọn khác less.jsđược sử dụng trong trình duyệt tại <http://lesscss.org/usage/#using-less-in-the-browser> .

Ở chế độ biên dịch phía máy khách, hầu hết các tùy chỉnh biểu định kiểu hiển thị ngay lập tức sau khi bạn tải lại một trang trong trình duyệt.

##### Khi bạn cần xóa các tệp xem tĩnh

Có một số loại thay đổi nhất định, yêu cầu bạn xóa pub/static/frontend/<Vendor>/<theme>/<locale>thư mục và kích hoạt quá trình biên dịch và [xuất bản](https://devdocs.magento.com/guides/v2.2/config-guide/cli/config-cli-subcommands-static-view.html#config-cli-static-overview) lại.

Điều này là bắt buộc trong các trường hợp sau:

* Nếu bạn thay đổi [tệp nguồn gốc](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-preprocess.html#css_preprocess_terms) chứa @magento\_importchỉ thị hoặc @importchỉ thị nơi tệp được nhập được chỉ định mà không có phần mở rộng.
* Nếu bạn đổi tên, xóa hoặc thêm .lesstệp được nhập bằng chỉ thị @magento\_importhoặc @importchỉ thị nhưng bạn không sửa các chỉ thị tương ứng.

Để xóa pub/static/frontend/<Vendor>/<theme>/<locale>thư mục, hãy xóa thư mục trong hệ thống tệp và tải lại các trang cửa hàng trong trình duyệt để kích hoạt biên dịch và xuất bản.

## Các @importquy tắc sử dụng chỉ thị

Bạn có thể nhập các tệp cục bộ và từ xa .lessvà .csstệp trong .lessbảng định kiểu Magento của bạn bằng cách sử dụng [@importchỉ thị](http://lesscss.org/features/#import-directives-feature) LESS tiêu chuẩn . Theo @importcú pháp, việc chỉ định phần mở rộng tệp cho tệp được nhập không phải là bắt buộc. Ví dụ: ký hiệu sau được cho phép:

@import 'source/lib/\_lib';

@import (css) 'styles';

Nhưng trong quá trình giải quyết đường dẫn tệp, Magento thêm .lessphần mở rộng cho các tệp đã nhập trong tất cả các @importentrees. Vì vậy, trong các tệp được xử lý, các câu lệnh từ ví dụ trước sẽ trông giống như sau:

@import 'source/lib/\_lib.less';

@import (css) 'styles.less';

Do đó, các tệp được xử lý khác với các tệp nguồn. Vì vậy, trong [chế độ biên dịch phía máy khách](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-preprocess.html#client-side) hoặc khi sử dụng [các lệnh grunt](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css_debug.html) , Magento không thể sử dụng các liên kết tượng trưng đến các tệp nguồn. Thay vào đó, nó sử dụng các bản sao của các tệp đã xử lý và chúng được xuất bản vào pub/staticthư mục. Trong trường hợp nhập tài nguyên CSS, điều này cũng dẫn đến việc không tìm và không nhập các tệp được yêu cầu.

### Nhập tệp CSS từ xa

Nếu bạn cần nhập tệp CSS từ xa vào .lessnguồn của mình , hãy sử dụng url()ký hiệu. Ví dụ: để nhập phông chữ của Google, hãy sử dụng ký hiệu sau:

@import url('//fonts.googleapis.com/css?family=Titillium+Web:400,300,200,600.css');

Bằng cách này, Magento sẽ bỏ qua @importchỉ thị trong khi giải quyết các đường dẫn đến các tài nguyên cục bộ.

## Chỉ thị @magento\_import

@magento\_importlà một chỉ thị LESS dành riêng cho Magento cho phép bao gồm nhiều tệp theo mẫu tên. Nó được sử dụng để bao gồm các tệp có cùng tên từ các vị trí khác nhau, ví dụ: các mô-đun khác nhau. @importChỉ thị chuẩn bao gồm một tệp duy nhất, được tìm thấy theo các [tệp dự phòng tĩnh](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-inherit.html#theme-inherit-static) .

@magento\_import chỉ có thể được sử dụng trong các tệp nguồn gốc của một chủ đề.

@magento\_import quy tắc sử dụng

Để bao gồm một .lesstệp bằng @magento\_importchỉ thị:

1. Để tránh bất kỳ xung đột nào với cú pháp LESS ban đầu, @magento\_importphải được nhận xét với hai dấu gạch chéo. Nếu không, bộ tiền xử lý LESS bỏ qua nó. **Thí dụ:**
2. // Comment in a LESS document
4. // Standard LESS import directive
5. // ---------------------------------------------
7. @import 'source/\_reset';
8. @import '\_styles';
10. //
11. // Custom Magento LESS import directives
12. // ---------------------------------------------
14. //@magento\_import 'source/\_module.less'; // Theme modules
15. //@magento\_import 'source/\_widgets.less'; // Theme widgets
16. //@magento\_import 'source/\_extend.less'; // Extend for minor customization
17. @magento\_importphải chứa đường dẫn tệp. Đường dẫn được chỉ định tương đối với tệp, trong đó chỉ thị được gọi và đặt trong dấu nháy đơn ('') hoặc dấu ngoặc kép (“”). Cách tốt nhất là chỉ định phần mở rộng của tệp trong đường dẫn, mặc dù về mặt kỹ thuật, bạn có thể bỏ qua điều này.

@magento\_import processing

Trong phạm vi tiền xử lý tĩnh, bộ tiền xử lý LESS dựng sẵn thực hiện như sau:

1. Tìm kiếm tất cả các @magento\_importchỉ thị.
2. Thay thế các @magento\_importchỉ thị ban đầu bằng các @importchỉ thị tiêu chuẩn . Sau đó xác định đường dẫn đến các tệp cụ thể tương ứng với mẫu được chỉ định trong @magento\_import.

Ví dụ về cách @magento\_importsử dụng và xử lý trong <Magento\_Blank\_theme\_dir>/web/css/styles-l.less:

|  |  |
| --- | --- |
| **Trước** | **Sau** |
| Trong <Magento\_Blank\_theme\_dir>/web/css/styles-l.lessđó có một chỉ thị sau đây:    ..  // @ magento\_import 'nguồn / \_widgets.less'; // Tiện ích con chủ đề  .. | Trong tệp được xử lý, điều này dẫn đến:  @import '../Magento\_Catalog/css/source/\_widgets.less';  @import '../Magento\_Cms/css/source/\_widgets.less';  @import '../Magento\_Reports/css/source/\_widgets.less';  @import '../Magento\_Sales/css/source/\_widgets.less';  // Tiện ích con chủ đề |

## Compile LESS using Grunt

### Thêm theme vào cấu hình grunt

Để biên dịch .less tệp, hãy thêm chủ đề của bạn module.exports vào cấu hình Grunt, hoặc trong [tệp](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/tools/using_grunt.html#grunt_config) mặc định dev/tools/grunt/configs/themes.jshoặc trong [tệp cấu hình tùy chỉnh](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/tools/using_grunt.html#grunt_config) . Ví dụ:

1. Cài đặt [node.js] đến bất kỳ vị trí nào trên máy của bạn.
2. Cài đặt Grunt CLI trên toàn cầu:

npm install -g grunt-cli

1. Đổi tên các tệp sau trong thư mục gốc Magento của bạn:
   * package.json.sample đến package.json
   * Gruntfile.js.sample đến Gruntfile.js
2. Cài đặt (hoặc làm mới) node.jsphụ thuộc dự án, bao gồm Grunt, cho cá thể Magento của bạn. Để thực hiện việc này, hãy chạy các lệnh sau trong dấu kiểm nhắc lệnh:

cd <your\_Magento\_instance\_directory>

npm install

npm update

1. Thêm chủ đề của bạn vào cấu hình Grunt. Để thực hiện việc này, trong dev/tools/grunt/configs/themes.jstệp, hãy thêm chủ đề của bạn vào module.exportsnhư sau:

module.exports = {

<theme>: {

area: 'frontend',

name: '<Vendor>/<theme>',

locale: '<language>',

files: [

'<path\_to\_file1>', //path to root source file

'<path\_to\_file2>'

],

dsl: 'less'

},

Trường hợp ký hiệu sau được sử dụng:

* + <theme>: mã chủ đề của bạn, thông thường phải tương ứng với tên thư mục chủ đề.
  + <language>: được chỉ định trong code\_subtagđịnh dạng, chẳng hạn en\_US. Chỉ có thể chỉ định một ngôn ngữ ở đây. Để gỡ lỗi chủ đề bằng một ngôn ngữ khác, hãy tạo một khai báo chủ đề khác, có một giá trị được chỉ định khác cholanguage
  + <path\_to\_file>: đường dẫn đến tệp nguồn gốc, liên quan đến app/design/frontend/<Vendor>/<theme>/webthư mục. Bạn cần chỉ định tất cả [các tệp nguồn gốc của chủ đề](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/css-topics/css-preprocess.html#css_preprocess_terms) . Nếu chủ đề của bạn [kế thừa] từ một chủ đề nhất định và không chứa các tệp nguồn gốc của riêng nó, hãy chỉ định tệp nguồn gốc của chủ đề gốc.

1. (Tùy chọn) Nếu bạn muốn sử dụng Grunt để tự động thay đổi "xem", mà không tải lại các trang trong trình duyệt mỗi lần, hãy cài đặt [Tiện ích mở rộng LiveReload] trong trình duyệt của bạn.

### Lệnh grant

Bảng sau đây mô tả các lệnh grunt bạn có thể sử dụng để thực hiện các tác vụ tùy chỉnh khác nhau. Chạy tất cả các lệnh từ thư mục cài đặt Magento của bạn.

| **Nhiệm vụ Grunt** | **Hoạt động** |
| --- | --- |
| grunt clean | Loại bỏ các tệp tĩnh liên quan đến chủ đề trong thư mục pub/staticvà var. |
| grunt exec | Tái xuất bản các liên kết tượng trưng đến các tệp nguồn vào pub/static/frontend///thư mục. |
| grunt less | Biên dịch .csscác tệp bằng cách sử dụng các liên kết tượng trưng được xuất bản trong pub/static/frontend///thư mục. |
| grunt watch | Theo dõi các thay đổi trong tệp nguồn, biên dịch lại .csstệp và tải lại trang trong trình duyệt. |

## Sử dụng các trường hợp theo dõi thay đổi bằng cách sử dụng Grunt

Sau đây cho thấy các tác vụ Grunt nào để sử dụng để gỡ lỗi:

* Sau khi bạn chuyển đổi chế độ biên dịch từ phía máy khách sang phía máy chủ, hãy chạy execlệnh.
* Sau khi bạn tùy chỉnh nội dung của bất kỳ .lesstệp nào , ngoại trừ tệp nguồn gốc, hãy chạy lesstác vụ và tải lại trang.
* Sau khi bạn [tùy chỉnh các tệp nguồn gốc hoặc di chuyển các tệp được đưa vào tệp gốc], hãy chạy execlệnh và tải lại trang.

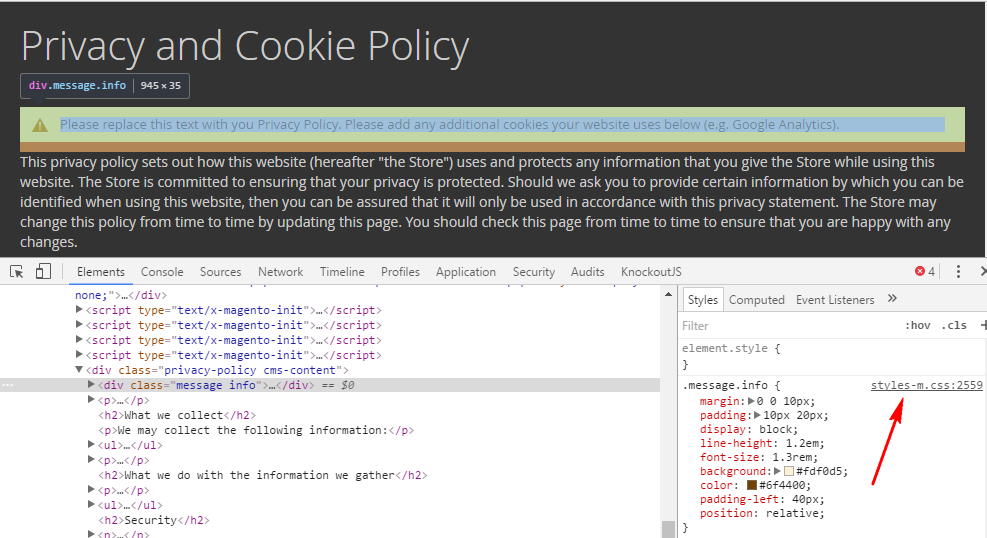
Nếu bạn đã cài đặt LiveReload, hãy chạy grunt watchlệnh và luồng đơn giản hơn:

* Sau khi bạn tùy chỉnh nội dung của bất kỳ .lesstệp nào , các thay đổi sẽ được áp dụng và trang sẽ tự động tải lại. Không có thay đổi bổ sung
* Sau khi bạn [tùy chỉnh các tập tin nguồn gốc hoặc di chuyển các tập tin bao gồm các tập tin gốc], chạy cleanvà execlệnh, trong đó tải lại trang trong trình duyệt.

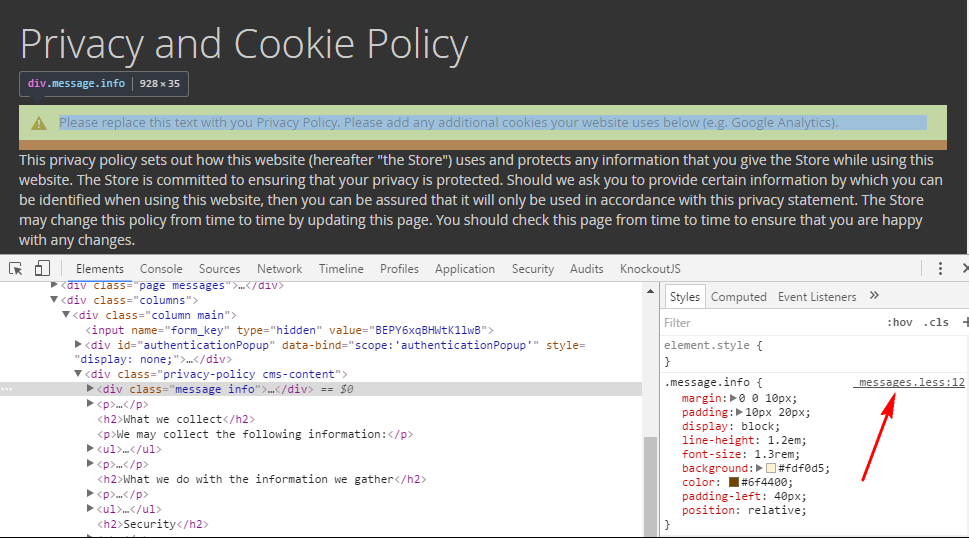
## Bản đồ nguồn CSS

Khi sử dụng Grunt để xử lý trước các kiểu, bạn có thể cho phép tạo bản đồ nguồn CSS trong trình duyệt của mình. Nó sẽ làm cho các kiểu chủ đề gỡ lỗi dễ dàng hơn.

Đối với mỗi chủ đề, Magento biên dịch tất cả .lesscác tệp chủ đề thành hai tệp CSS: styles-m.cssvà styles-l.css. Vì vậy, khi bạn gỡ lỗi một chủ đề, trình duyệt của bạn chỉ thấy styles-m.cssvà có thể khó xác định tệp .csshoặc .lesstệp nào yêu cầu chỉnh sửa. Ví dụ:



Bản đồ nguồn CSS giải quyết vấn đề này. Họ giúp tìm .lesstệp, trong đó kiểu được chỉ định. Ví dụ:

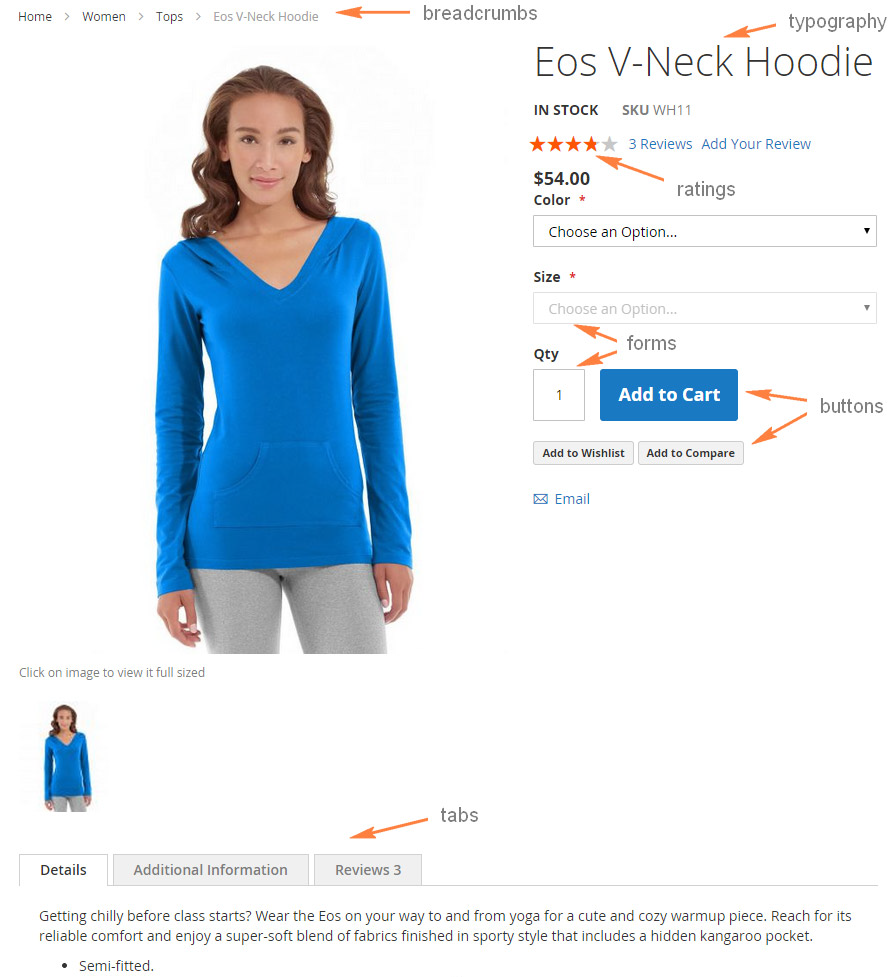


Bản đồ nguồn CSS được tạo tự động khi bạn biên dịch CSS cho chủ đề của mình bằng cách sử dụng grunt less: <theme>lệnh. Để sử dụng chúng, bạn cần bật hiển thị bản đồ nguồn trong trình duyệt của mình. Ví dụ: trong Chrome, bạn sẽ mở Công cụ dành cho nhà phát triển, đi tới bảng điều khiển **Cài đặt** , chọn **Tùy chọn** , sau đó chọn hộp kiểm **Bật bản đồ nguồn CSS** .

## Magento UI library

Thư viện giao diện người dùng Magento là thư viện frontend linh hoạt dựa trên [LESS](http://lesscss.org/) được thiết kế để hỗ trợ các nhà phát triển chủ đề Magento . Nó sử dụng một tập hợp các mixin cho các phần tử cơ sở để dễ dàng phát triển và tùy biến chủ đề giao diện người dùng.

## Các thành phần được cung cấp bởi thư viện UI

* actions-toolbar
* breadcrumbs
* buttons
* drop-downs
* forms
* icons
* layout
* loaders
* messages
* pagination
* popups
* ratings
* sections
* tabs and accordions
* tables
* tooltips
* typography
* list of theme variables
* Hình minh họa sau đây hiển thị trang sản phẩm mặt tiền cửa hàng có chứa một số yếu tố trước:
* 

## Vị trí Mixin

Bạn có thể tìm thấy thư viện giao diện người dùng Magento bên dưới [lib/web/css](https://github.com/magento/magento2/blob/2.1/lib/web/css). Các .lesstệp nguồn thư viện được lưu trữ trong sourcethư mục, mỗi tệp chứa các mixin để định cấu hình một phần tử nhất định và trong hầu hết các trường hợp, phần tử trùng với tên tệp:

lib/web

├── css/

│ ├── docs/ (Library documentation)

│ ├── source/

│ │ ├── lib/ (Library source files)

| | | ├── variables/ (Predefined variables for each mixin)

│ │ │ ├── \_actions-toolbar.less

│ │ │ ├── \_breadcrumbs.less

│ │ │ ├── \_buttons.less

│ │ │ ├── \_dropdowns.less

│ │ │ ├── \_forms.less

| | | ├── \_grids.less

│ │ │ ├── \_icons.less

│ │ │ ├── \_layout.less

│ │ │ ├── \_lib.less

│ │ │ ├── \_loaders.less

│ │ │ ├── \_messages.less

│ │ │ ├── \_navigation.less

│ │ │ ├── \_pages.less

│ │ │ ├── \_popups.less

│ │ │ ├── \_rating.less

│ │ │ ├── \_resets.less

│ │ │ ├── \_responsive.less

│ │ │ ├── \_sections.less

│ │ │ ├── \_tables.less

│ │ │ ├── \_tooltips.less

│ │ │ ├── \_typography.less

│ │ │ ├── \_utilities.less

│ │ │ └── \_variables.less

│ │ └── \_extend.less

│ │ └── \_theme.less

│ │ └── \_variables.less

│ └── styles.less

├── fonts/

│ └── Blank-Theme-Icons/ (Library custom icons font)

├── images/

│ └── blank-theme-icons.png (Library icons sprite)

└── jquery/ (Library javascript files)

## Biến được xác định trước

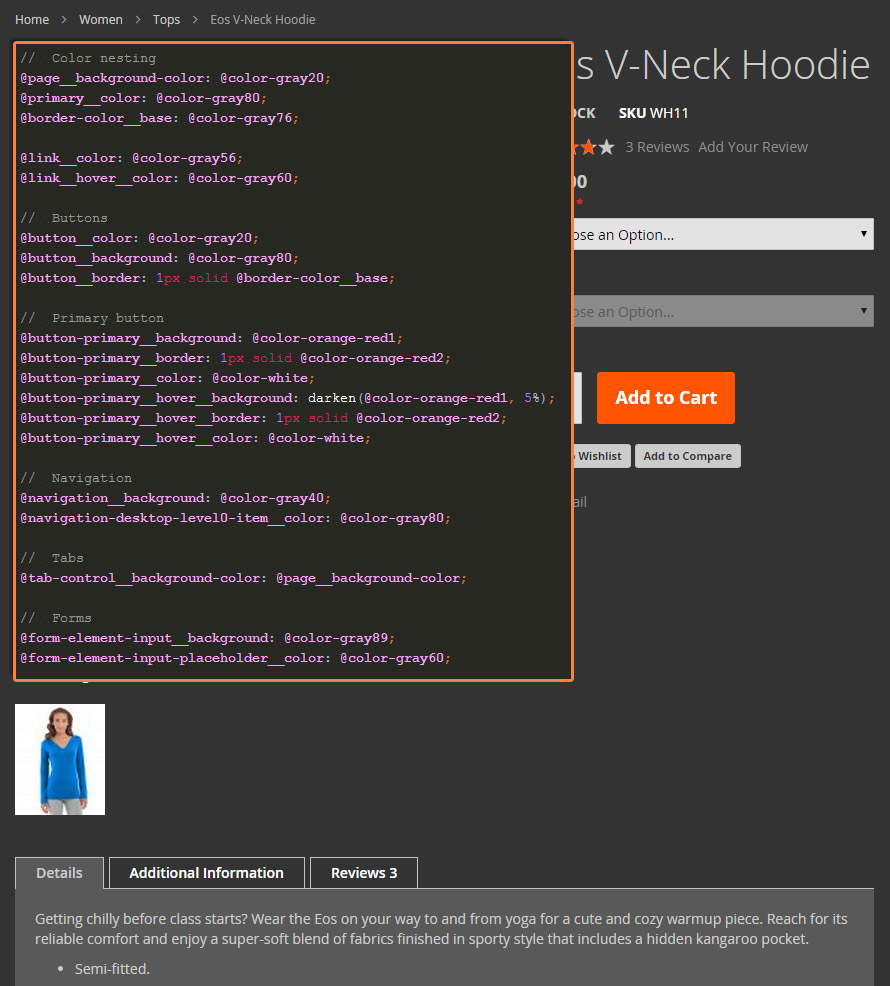
Nếu chủ đề của bạn [kế thừa](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-inherit.html) từ bất kỳ chủ đề nào khác của Magento, chẳng hạn như Blank, bạn có thể dễ dàng tùy chỉnh bất kỳ phần tử nào của trang cửa hàng mà không thay đổi bất kỳ mã CSS hoặc mẫu nào. Tùy biến có thể được thực hiện bằng cách thay đổi trong chủ đề của bạn các giá trị của các biến được xác định trước được sử dụng trong thư viện UI hoặc hỗn hợp chủ đề gốc.

Danh sách đầy đủ của các biến này và giá trị mặc định của chúng được lưu trữ trong [lib/web/css/source/lib/variables](https://github.com/magento/magento2/blob/2.1/lib/web/css/source/lib/variables). Thư mục này chứa một tập hợp các tệp, tương ứng với tập các phần tử thư viện UI và mỗi tệp liệt kê các biến phần tử cụ thể. Ví dụ, [lib/web/css/source/lib/variables/\_breadcrumbs.less](https://github.com/magento/magento2/blob/2.1/lib/web/css/source/lib/variables/_breadcrumbs.less)chứa các biến được sử dụng trong breadcrumbs() mixin .

Để thay đổi giá trị biến thư viện mặc định, hãy chỉ định giá trị mới cho các biến bắt buộc trong <theme\_dir>/web/css/source/\_theme.lesstệp.

Xin lưu ý rằng <theme\_dir>/web/css/source/\_theme.lesstệp của bạn ghi đè \_theme.lesschủ đề gốc (nếu chủ đề của bạn có bố mẹ). Vì vậy, nếu bạn muốn kế thừa giá trị biến của chủ đề gốc bổ sung cho các thay đổi của bạn, hãy thêm nội dung của phụ huynh \_theme.lessvào tệp của bạn.

Hình dưới đây cho thấy trang sản phẩm được hiển thị trước đó trong chủ đề này, sau khi một chủ đề tùy chỉnh được áp dụng. Chủ đề tùy chỉnh Blank bằng cách chỉ định nghĩa lại các biến.



## Biến tùy chỉnh của bạn

Khi đặt tên biến tùy chỉnh, vui lòng thực hiện theo [quy ước đặt tên Magento cho các biến LESS](https://devdocs.magento.com/guides/v2.2/coding-standards/code-standard-less.html#naming) .

## Sử dụng mixin

Bạn có thể sử dụng mixin với các giá trị biến mặc định hoặc bạn có thể xác định lại chúng khi gọi mixin. Các đoạn sau mô tả cả hai cách gọi một mixin.

Để sử dụng mixin với các giá trị mặc định, hãy gọi mixin mà không chỉ định bất kỳ tham số nào. Ví dụ:

.breadcrumbs {

.breadcrumbs();

}

Để gọi một mixin với các giá trị tham số khác với mặc định, hãy đặt các giá trị này khi gọi mixin, như trong ví dụ sau:

.example-button {

.button(

@\_button-padding: @button-padding,

@\_button-color: #fff,

@\_button-color-hover: #ccc

);

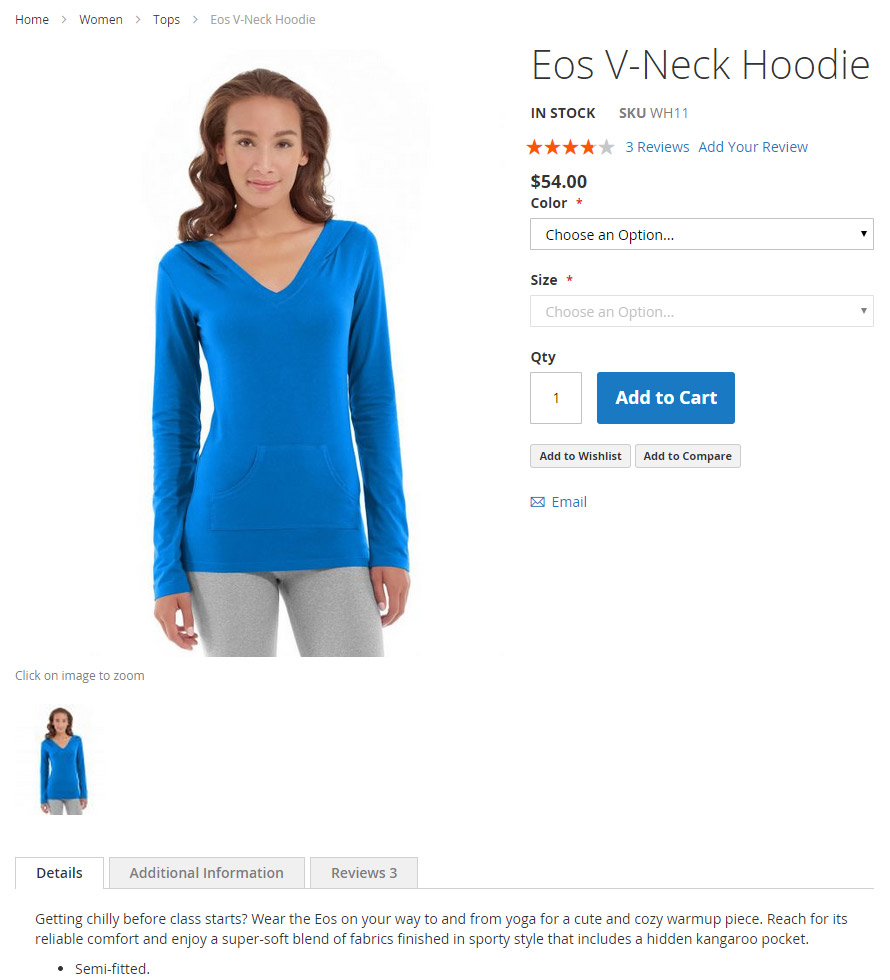
}

Các biến bắt đầu với @\_các biến mixin riêng được sử dụng chỉ trong mixin này. Các biến bắt đầu bằng @(không có dấu gạch dưới) là toàn cục và được liệt kê trong [lib/web/css/source/lib/variables](https://github.com/magento/magento2/blob/2.1/lib/web/css/source/lib/variables).

## Customizing styles illustration (tùy chỉnh style)

## Thay đổi lược đồ màu chủ đề

OrangeCo đã tạo ra một chủ đề tùy chỉnh kế thừa từ chủ đề Blank cơ bản của Magento. Hình ảnh sau đây minh họa cách các trang cửa hàng trông như thế nào khi chủ đề Trống được áp dụng:



Trong chủ đề Xám của họ, OrangeCo muốn thay đổi bảng màu từ trắng sang xám.

Thư mục chủ đề Xám là app/design/frontend/OrangeCo/grey.

OrangeCo quyết định sử dụng thư viện giao diện người dùng Magento, do đó, để thay đổi bảng màu, họ cần xác định các giá trị mới cho các biến LESS mặc định nhất định. Để thực hiện việc này, họ đã thêm \_theme.lesstệp ghi đè vào app/design/frontend/OrangeCo/grey/web/css/sourcethư mục, với nội dung sau:

// Color nesting

@page\_\_background-color: @color-gray20;

@sidebar\_\_background-color: @color-gray40;

@primary\_\_color: @color-gray80;

@border-color\_\_base: @color-gray76;

@link\_\_color: @color-gray56;

@link\_\_hover\_\_color: @color-gray60;

// Buttons

@button\_\_color: @color-gray20;

@button\_\_background: @color-gray80;

@button\_\_border: 1px solid @border-color\_\_base;

// Primary button

@button-primary\_\_background: @color-orange-red1;

@button-primary\_\_border: 1px solid @color-orange-red2;

@button-primary\_\_color: @color-white;

@button-primary\_\_hover\_\_background: darken(@color-orange-red1, 5%);

@button-primary\_\_hover\_\_border: 1px solid @color-orange-red2;

@button-primary\_\_hover\_\_color: @color-white;

// Navigation

@navigation-level0-item\_\_color: @color-gray80;

@submenu-item\_\_color: @color-gray80;

@navigation\_\_background: @color-gray40;

@navigation-desktop-level0-item\_\_color: @color-gray80;

@navigation-desktop-level0-item\_\_hover\_\_color: @color-gray34;

@navigation-desktop-level0-item\_\_active\_\_color: @navigation-desktop-level0-item\_\_color;

// Tabs

@tab-control\_\_background-color: @page\_\_background-color;

// Forms

@form-element-input\_\_background: @color-gray89;

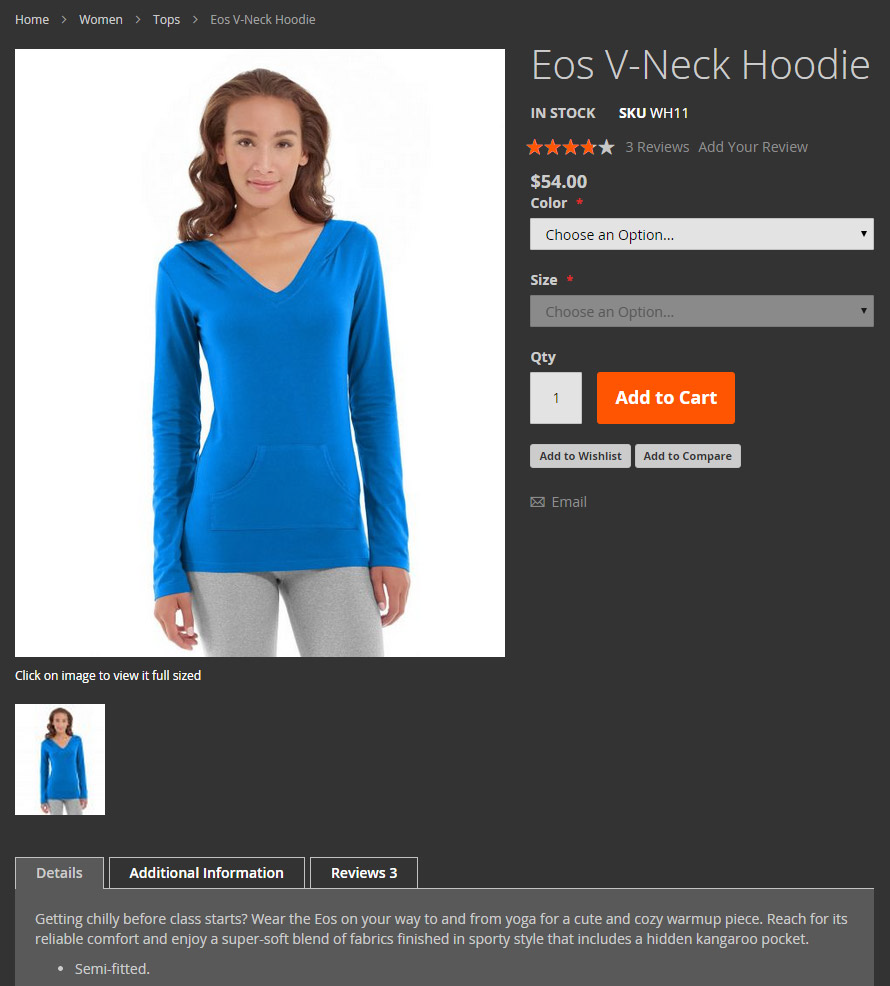
@form-element-input-placeholder\_\_color: @color-gray60;

// Header icons

@header-icons-color: @color-gray89;

@header-icons-color-hover: @color-gray60;

Sau khi chủ đề Màu xám được áp dụng (và [các tệp bộ nhớ cache bị xóa](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/cache_for_frontdevs.html#clean_static_cache) ), các trang lưu trữ sẽ trông giống như sau:



## Tùy chỉnh front chữ

Ứng dụng Magento chứa một bộ phông chữ tích hợp, nhưng bạn có thể dễ dàng bao gồm các phông chữ tùy chỉnh. Chủ đề này mô tả cách bao gồm phông chữ tùy chỉnh được lưu trữ cục bộ trong chủ đề Magento của bạn .

Nếu bạn chưa quen với việc phát triển chủ đề Magento, hãy xem [Tạo chủ đề](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/themes/theme-create.html) để làm quen với các khái niệm cơ bản.

Để đảm bảo tính ổn định của các tùy chỉnh của bạn và ngăn các nâng cấp ghi đè tùy chỉnh của bạn, không thay đổi các tệp chủ đề Magento mặc định. Bạn phải bao gồm các phông chữ tùy chỉnh trong biểu định kiểu của chủ đề.

1. Thêm các tệp phông chữ vào thư mục chủ đề cục bộ của bạn. Ví dụ app/design/frontend/<your\_vendor\_name>/<your\_theme\_name>/web/fonts,.

Để thêm phông chữ *bên ngoài* , hãy thêm tham chiếu phông chữ vào tệp cấu hình trang như được mô tả trong [Bao gồm tài nguyên tĩnh (JavaScript, CSS, phông chữ)](https://devdocs.magento.com/guides/v2.2/frontend-dev-guide/layouts/xml-manage.html#layout_markup_css) .

1. Nếu bạn xây dựng một chủ đề bằng cách sử dụng thư viện giao diện người dùng Magento, hãy khai báo phông chữ bằng cách thêm tệp .lib-font-face mixin vào app/design/frontend/<your\_vendor\_name>/<your\_theme\_name>/web/css/source/\_typography.lesstệp:
2. .lib-font-face(
3. @family-name:'<any\_font\_name>',
4. @font-path: '@{baseDir}fonts/<path\_to\_font\_file>',
5. @font-weight: <font\_weight>,
6. @font-style: <font\_style>
7. );

Ở đâu:

* + {@baseDir}viết tắt của app/design/frontend/<you\_vendor\_name>/<your\_theme\_name>/webthư mục.
  + <path\_to\_font\_file>bao gồm tên tệp phông chữ, nhưng không có phần mở rộng. Ví dụ: @font-path: '@{baseDir}fonts/Luma-Icons'đối với phông chữ được lưu trữ web/fonts/Luma-Icons.woff.

Mixin tạo CSS, bao gồm phông chữ. Ví dụ sau đây cho thấy cách tạo CSS cho phông chữ Open Sans trong chủ đề Trống:

@font-face {

font-family: 'Open Sans';

src: url('../fonts/opensans/light/opensans-300.eot');

src: url('../fonts/opensans/light/opensans-300.eot?#iefix') format('embedded-opentype'), url('../fonts/opensans/light/opensans-300.woff2') format('woff2'), url('../fonts/opensans/light/opensans-300.woff') format('woff'), url('../fonts/opensans/light/opensans-300.ttf') format('truetype'), url('../fonts/opensans/light/opensans-300.svg#Open Sans') format('svg');

font-weight: 300;

font-style: normal

}

Nếu chủ đề của bạn không sử dụng thư viện giao diện người dùng Magento, hãy bao gồm phông chữ trong các tệp CSS của chủ đề của bạn bằng cách sử dụng @font-facequy tắc CSS.