

# Variant Links

## Magento 2 Extension — Configuration & Storefront Guide

Adds smart “View Other Options / Finishes / Sizes” buttons to product pages that send shoppers straight to the matching variant products — by manual link, a point-and-click builder, bulk CSV, or fully automatic generation.

Applies to ETechFlow\_VariantLinks v1.1.1 · Magento 2.4.9 · Luma + Hyvä

### 1. What Variant Links does

Variant Links solves a common catalog problem: the “same” product often exists as several separate products — the same bag in three colours, the same key blank for five car makes, the same lock in four sizes. Customers landing on one of them have no easy way to discover the others. Variant Links adds a row of “View Other ...” buttons to the product page that link to a filtered category listing of the related variants, so shoppers can jump straight to the colour / finish / size they actually want.

Typical merchant problems it solves:

- Colour/finish ranges split across separate products with no cross-links between them.
- Customers bouncing because they can’t find the size or variation they need.
- Cross-sell links hand-coded into product descriptions that break when the domain or URL changes.
- No scalable way to add these links to hundreds of products at once.

### The three built-in buttons

Each button links to the same category as the current product, holding certain attributes fixed so that only the intended dimension varies:

Button	Holds fixed	What varies for the shopper
<b>View Other Options</b>	Manufacturer / brand	The brand’s full range in this category
<b>View Other Finishes</b>	Manufacturer + size	Finish / colour
<b>View Other Sizes</b>	Manufacturer + finish	Size / length

On top of these, you can define your own unlimited Custom Buttons (e.g. “View Other Lengths”). A button only appears when it has somewhere to point — either a link you set, or one the module generates.

### Four ways to set the links

- Visual builder — pick a target category and filters from dropdowns on the product page (no URLs).
- Simple link field — paste a relative URL; the module adds the domain automatically.
- Bulk CSV import — set links on hundreds of products by SKU in one upload.
- Dynamic auto-generation — the module builds the link on the fly from the product’s own category + attributes, with zero per-product setup.

## How this document is laid out

We walk through the module in the order a merchant actually experiences it:

- Section 2 — Module navigation: the admin entries Variant Links adds.
- Section 3 — Global configuration: buttons, labels, dynamic mode, custom buttons.
- Section 4 — Setting links on a product (the visual builder + simple fields).
- Section 5 — Bulk CSV import and command-line tools.
- Section 6 — Legacy button stripping.
- Section 7 — Storefront result: what the customer sees.
- Section 8 — Troubleshooting.

## 2. Module navigation

Variant Links adds its own group under the Catalog sidebar, with two entries.

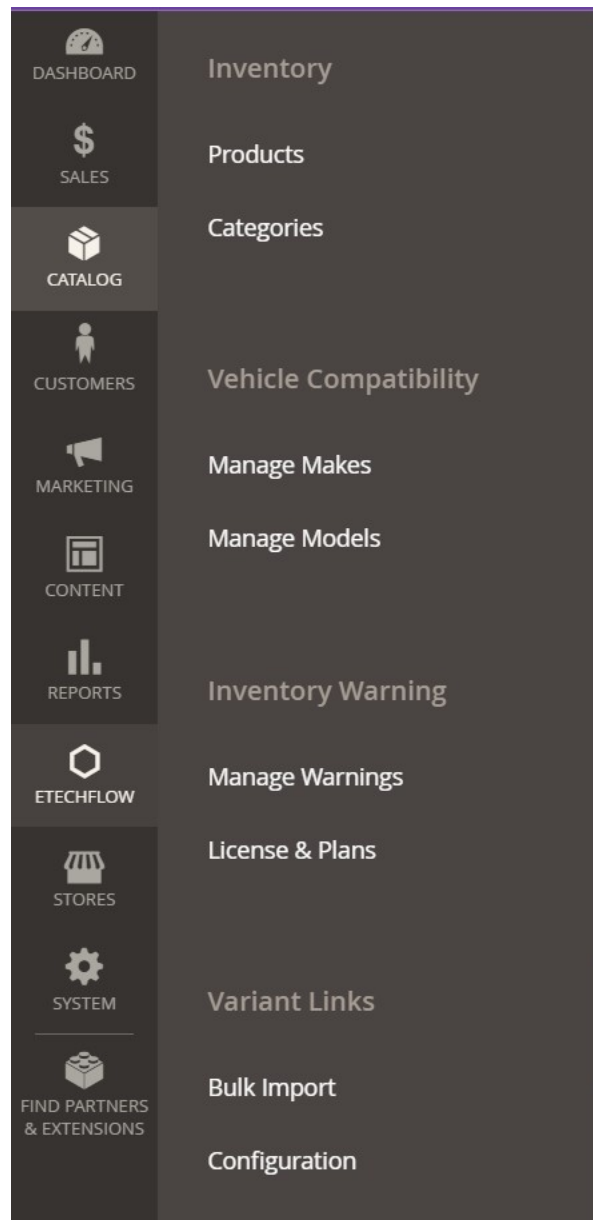


Figure 2.1 — Catalog → Variant Links: "Bulk Import" (CSV upload) and "Configuration" (jumps to the settings section). Per-product links are set on the product edit page itself (Section 4).

### 3. Global module configuration

Stores → Configuration → ETECHFLOW → Variant Links groups every store-wide setting. Beyond Module Status and License (above), there are three functional groups: Buttons, Auto-Generated Links (Dynamic), and Custom Buttons.

#### 3.1 Buttons — which buttons show, their labels, and legacy cleanup

This group controls the built-in buttons. "Buttons to show" is a multiselect that decides which of Options / Finishes / Sizes are allowed to appear store-wide — deselect one and that button never renders, even if a product has a link for it. The label fields rename each button (e.g. Finishes renamed to "See Other Colours"). "Strip old in-description buttons" removes hand-coded buttons from the rendered description (Section 6).

Add or remove a button here with no code change.

"Options" button label [store view]  **Rename any button**

"Finishes" button label [store view]

"Sizes" button label [store view]

Strip old in-description buttons [store view]

Removes the old hand-coded "View Other ..." anchors (and button-styled links) from the product description as it renders, so they don't show twice. The stored description is left untouched — turn this off to show it exactly as written.

Figure 3.1 — Per-button labels (Finishes shown renamed to "See Other Colours") and the "Strip old in-description buttons" toggle. Leaving a label blank falls back to the default text.

Setting	What it does
Buttons to show	Multiselect of which built-in buttons may render store-wide. Clear it = no built-in buttons.
"Options/Finishes/Sizes" label	Custom text for each button; blank = the default "View Other ...".
Strip old in-description buttons	Removes legacy hand-coded buttons from the description as it renders (reversible).

### 3.2 Auto-Generated Links (Dynamic)

When enabled, any product that has no manually-set link gets a "View Other" link generated automatically from its own primary category plus Manufacturer / Finish / Size. Manual per-product links always take priority — dynamic mode only fills the gaps.

Auto-Generated Links (Dynamic) ⌵

When ON, products WITHOUT a manually-set link get a "View Other" link generated automatically from their own category + Manufacturer / Finish / Size. Manual per-product links always take priority. Options = the brand; Finishes = the finish; Sizes = vary size (needs a size attribute mapped for the category).

**Turn on dynamic links**

Enable auto-generated links [store view]

Default "size" attribute code [store view]

Used for the Sizes button when a category has no specific mapping below. Leave blank to only show Sizes where mapped.

Per-category size attribute map [store view]

One per line: **category-url-path=attribute\_code**. E.g. *euro-double-cylinders-locks=length*. Tells the Sizes button which attribute is "size" for that category.

Custom Buttons (Advanced) ⌵

Figure 3.2 — Dynamic mode. "Default size attribute code" and the per-category size map tell the Sizes button which attribute counts as "size" for each category.

Button	Generated link (same category +)
--------	----------------------------------

<b>Options</b>	?manufacturer=... (the brand's range; needs a category)
<b>Finishes</b>	?manufacturer=... &size=... (only if the product has a finish)
<b>Sizes</b>	?manufacturer=... &finish=... (only if a size attribute is mapped for the category)

### 3.3 Custom Buttons (Advanced)

Define your own buttons with no code — one per line, in the form `Button Label|attributes_to_keep_the_same`. The button links to other products in the same category with those attributes held to this product's values, so everything else varies. Use `__size__` for the per-category size attribute. A button auto-hides on any product missing one of the held attributes.

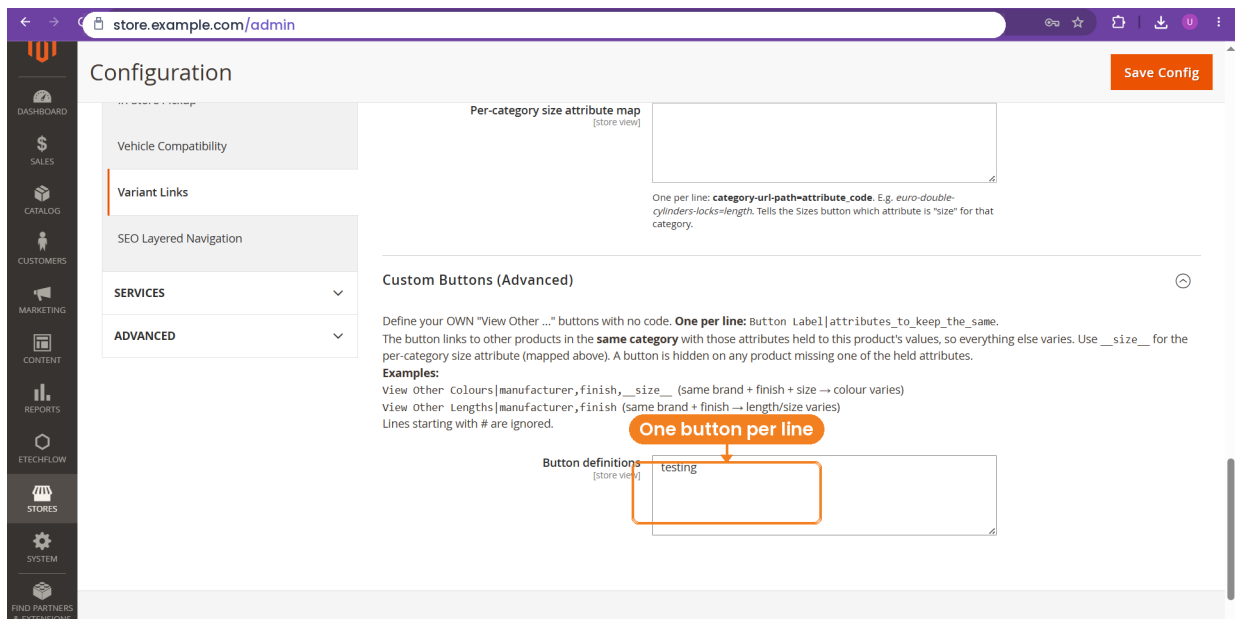


Figure 3.3 — Custom Buttons. Each line is "Label|attr1,attr2". Examples: "View Other Colours|manufacturer,finish,\_\_size\_\_" (colour varies); "View Other Lengths|manufacturer,finish" (length varies).

**Note** a label-only line with no "|attributes" (e.g. just "testing") creates a button that links to the whole category — useful as a "see everything in this category" link, but not a true variant filter.

## 4. Setting links on a product

Per-product links live in the "Variant Links" section on the product edit page (Catalog → Products → edit a product). There are two ways to fill them in.

### 4.1 The visual builder (recommended)

For each button you pick a Target category from a dropdown and add up to five Attribute = Value filter rows. The module assembles the correct filtered-category URL for you — no need to know or type any URL. Leave the Target category blank to fall back to dynamic auto-generation (Section 3.2); set one only to override.

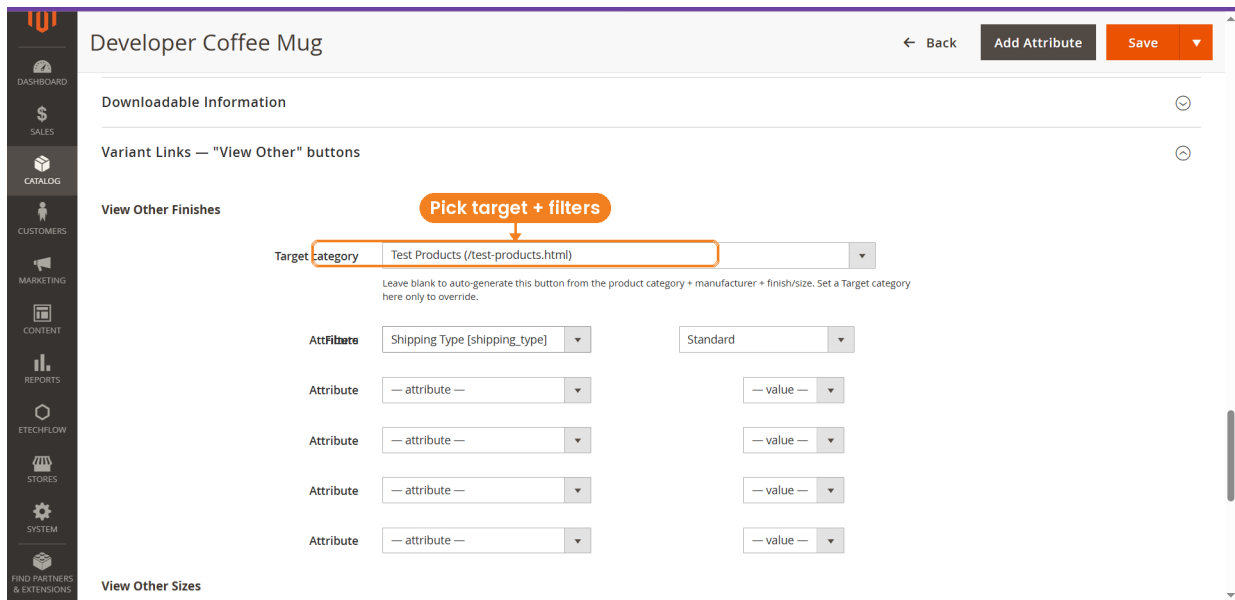


Figure 4.1 — The visual builder. "View Other Finishes" points to Test Products with the filter Shipping Type = Standard, so the button lands the shopper on exactly that filtered listing.

## 4.2 The simple link field

Alternatively, paste a relative URL (path + filters) straight into the field, e.g. /euro-cylinders?manufacturer=2081. Blank hides that button. You never include the domain — the module prepends the current store's domain at render time, so links never break if the store moves domains.

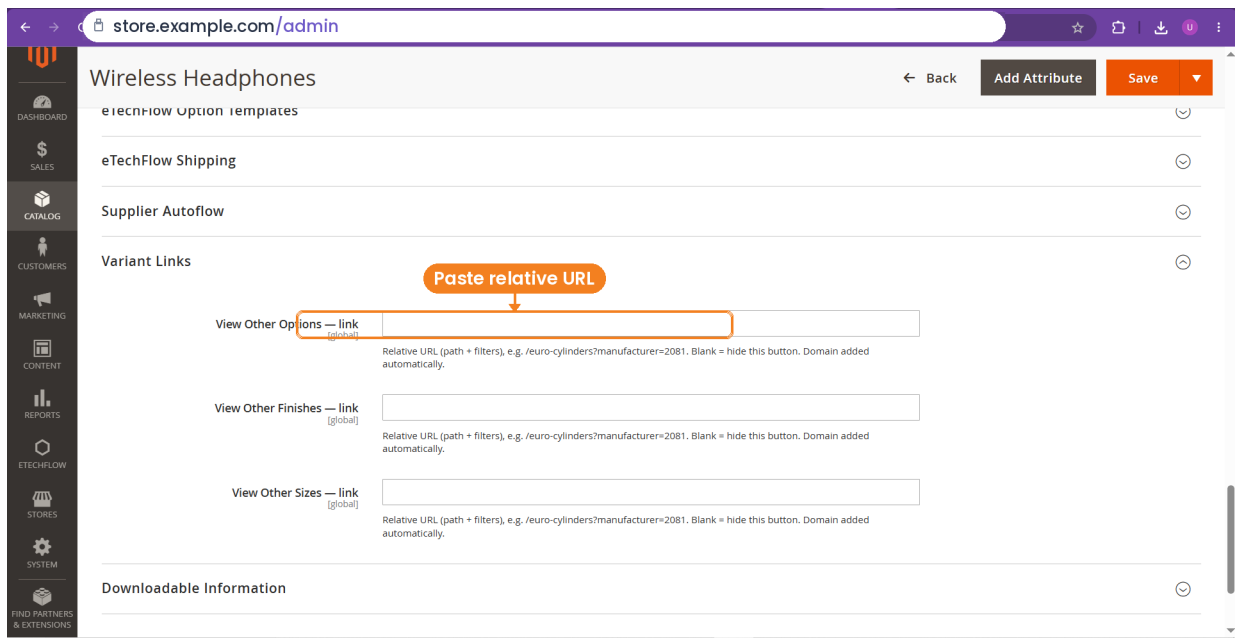


Figure 4.2 — The simple per-button link fields. The helper text confirms: "Relative URL (path + filters)... Blank = hide this button. Domain added automatically."

### 4.3 The attributes behind the filter

A filter only returns products that actually carry the chosen attribute value. In the example the button filters on Shipping Type = Standard, so the destination page lists every product whose Shipping Type attribute is set to Standard. If that attribute is empty on the other products, the filtered page will be empty — the button works, but there is nothing to show.

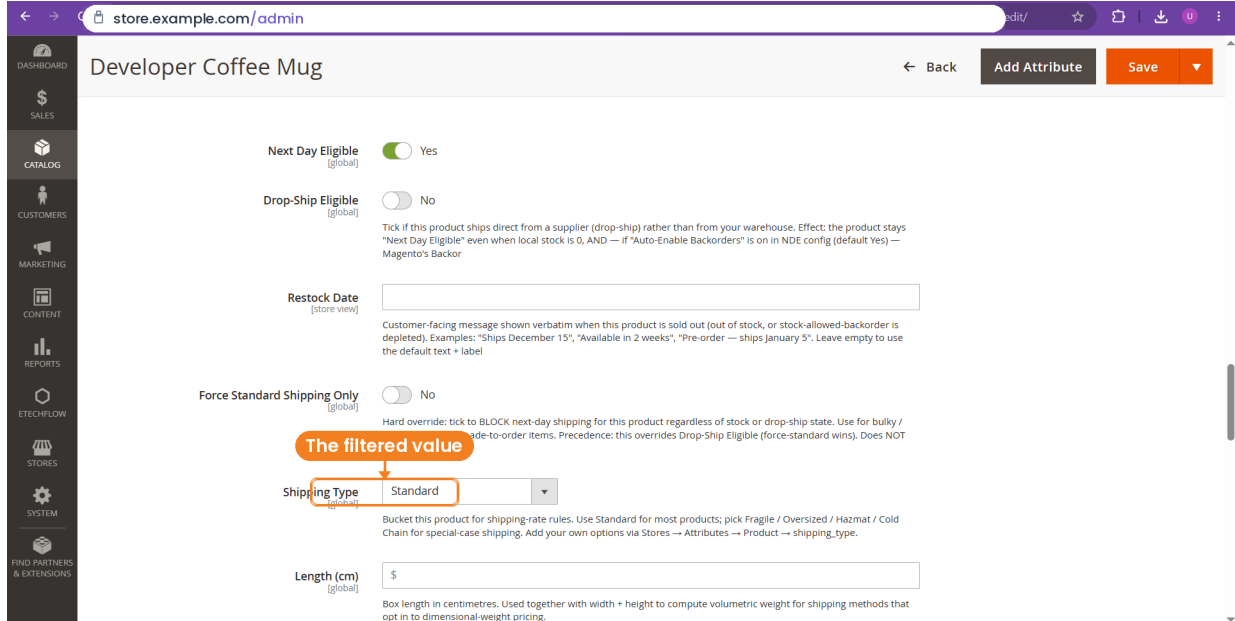


Figure 4.3 — The Shipping Type attribute set to "Standard" on the product. This is the value the Finishes filter in Figure 4.1 targets.

## 5. Bulk import & command line

### 5.1 Bulk Import (CSV)

Catalog → Variant Links → Bulk Import sets the "View Other" links on many products at once. Upload a CSV keyed by SKU; include only the columns you need. Tick "Dry run" first to preview how many rows match before anything is saved.

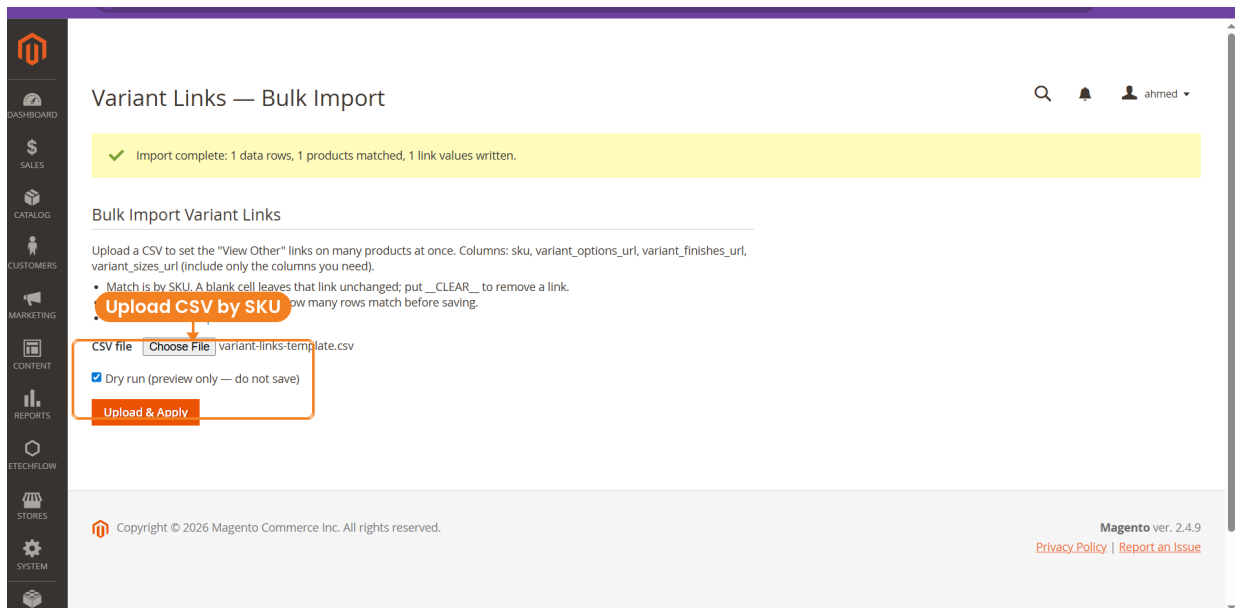


Figure 5.1 — Bulk Import. After upload it reports rows read, products matched, and links written. A "Download CSV template" link gives you the correct header row.

CSV column	Meaning
sku	The product to match (required).
variant_options_url	Relative link for the Options button.
variant_finishes_url	Relative link for the Finishes button.
variant_sizes_url	Relative link for the Sizes button.

A blank cell leaves that link unchanged; the special value `__CLEAR__` removes a link.

```
sku,variant_finishes_url
TEST-MUG-001,/test-products.html?shipping_type=standard
```

Figure 5.2 — A minimal CSV: one header row, then SKU + the relative URL for that button.

## 5.2 Console commands

Two CLI commands support automation and migration:

- `etchflow:variantlinks:import` — apply a CSV from the command line (same as the admin importer).
- `etchflow:variantlinks:migrate-legacy` — scan product descriptions for old hand-coded buttons and convert them into proper variant-link attributes.

## 6. Legacy button stripping

Many catalogs already have "View Other ..." buttons hand-coded into product descriptions. Showing those alongside the new module buttons would duplicate them. With "Strip old in-description buttons" = Yes (Figure 3.1), the module removes legacy buttons from the description as it renders — both the text-style "VIEW OTHER OPTIONS/FINISHES/SIZES" anchors and any anchor styled as a button (a border-radius in its inline style, e.g. an old red "VIEW CYLINDER LOCKS" link).

**Fully reversible** only the rendered HTML is cleaned — the stored description is never altered. Turn the toggle off to show it exactly as written.

**Theme-agnostic** it hooks the shared Catalog output helper, so it works on both Luma and Hyvä with no template edits.

## 7. Storefront result

On the product page the buttons render as a tidy row directly under the price / stock area.

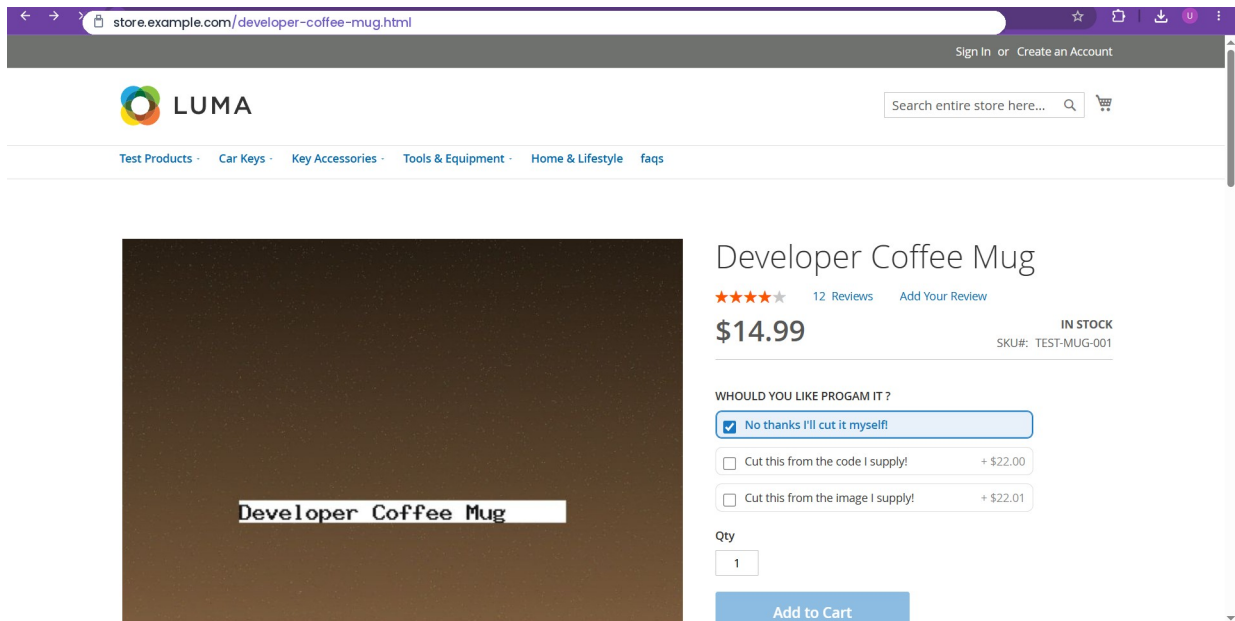


Figure 7.1 — The default "View Other Finishes" button on the product page.

Renaming the label in config (Section 3.1) changes the button text immediately:

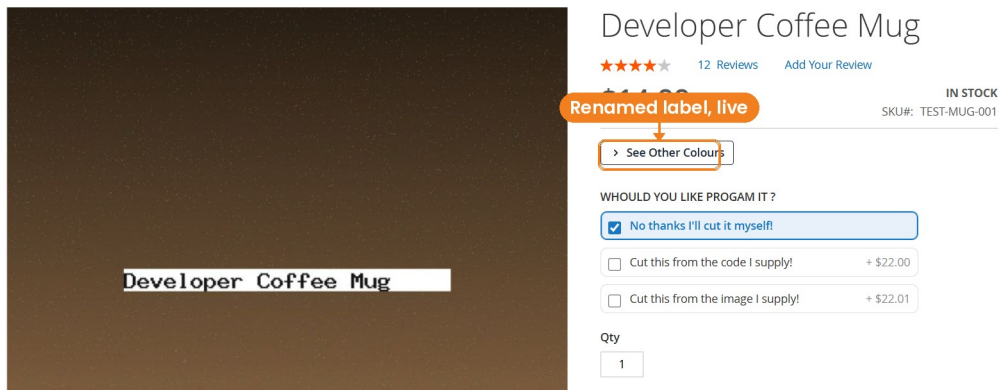


Figure 7.2 — The same button after renaming the Finishes label to "See Other Colours".

Multiple buttons — built-in and custom — render side by side:

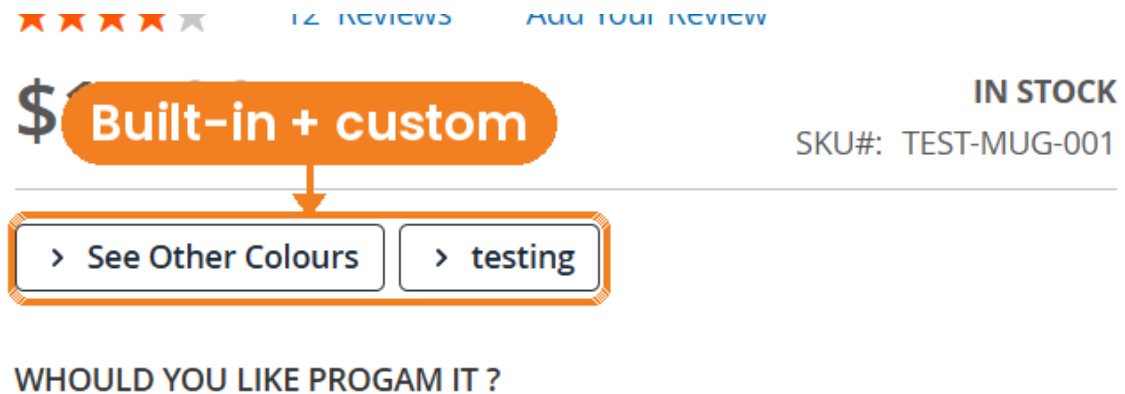


Figure 7.3 — Two buttons: the built-in "See Other Colours" plus a custom button ("testing").

Clicking a button lands the shopper on the filtered category listing — exactly the related variants:

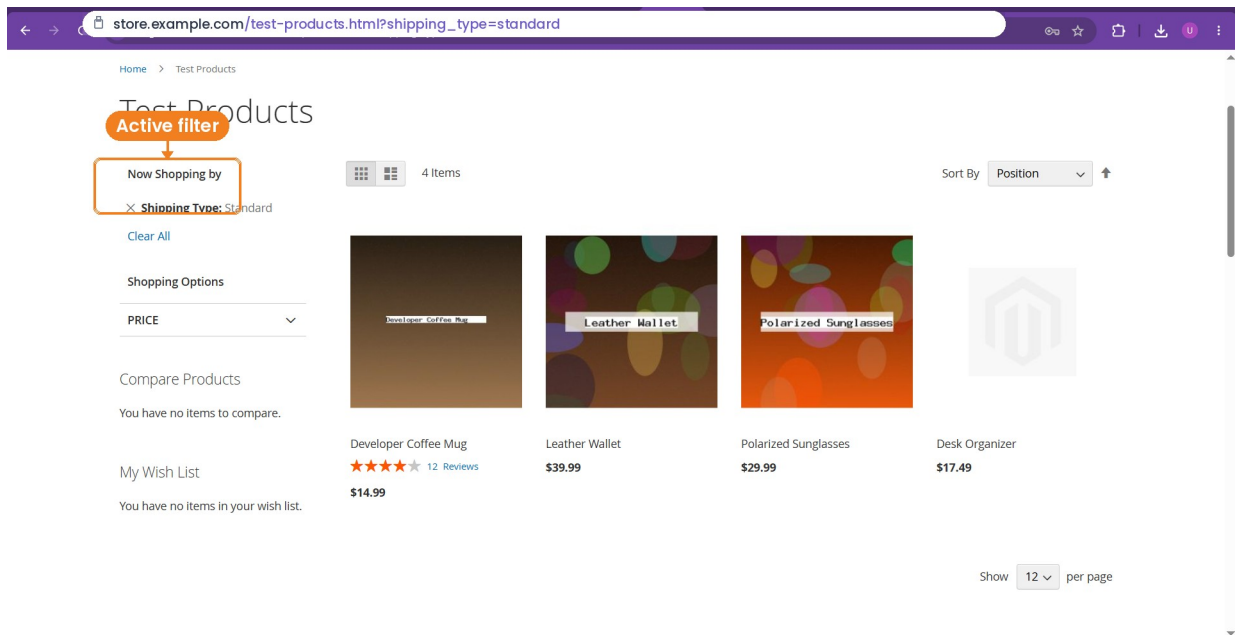


Figure 7.4 — The destination: Test Products filtered to "Shipping Type: Standard" (4 matching items). The active filter shows under "Now Shopping by".

## 8. Troubleshooting

Symptom	Likely cause	Fix
No buttons on any product	Module unlicensed	Ensure a valid licence key is saved and Module Status is green.
A button type never shows	Not in "Buttons to show"	Add it to the multiselect (Sec. 3.1).
One product has no button	No link set and dynamic off	Set a link (Sec. 4) or enable dynamic mode (Sec. 3.2).
Button shows but page is empty	No products carry the filtered value	Set the attribute on the other products / reindex.
Old button still in description	Strip legacy off	Set "Strip old in-description buttons" = Yes.
Link points to wrong domain	n/a — handled automatically	Store only the relative path; the domain is added at render.