

How to add sashes to your listing card / product page

There are several ways to achieve this feature. We'll highlight a few below.

Method 1: Using tags

Assigning a tag group for use as "sashes" will allow you to bulk tag many products so that they have the same sash.

Update AppServiceProvider

Add the following namespaces:

```
use Aero\Catalog\Models\Tag;  
use Aero\Catalog\Models\TagGroup;  
use Aero\Common\Facades\Settings;  
use Aero\Search\Elastic\Documents\ListingDocument;
```

Create a setting for your tag_sash id and extend the listings document:

```
class AppServiceProvider extends ServiceProvider  
{  
    public function boot()  
    {  
        Settings::group('search', function($group){  
            $group->eloquent('tag_sash', TagGroup::class);  
        });  
  
        ListingDocument::add('search-result', function ($document) {  
            if (! setting('search.tag_sash')) {  
                return [];  
            }  
        })  
    }  
}
```

```

    $listing = $document->getModel();

    $sashes = $listing->allTags
        ->where('tag_group_id', setting('search.tag_sash')->id)
        ->map(function (Tag $tag) use ($document) {
            return $tag->getTranslation('name', $document->getLanguage());
        })->unique()->values()->all();

    return ['sashes' => $sashes];
});
}
}

```

Add the following to your `listing-card.twig` snippet:

```

<template v-if="listing.sashes">
    <span v-for="sash in listing.sashes"
        class="aero-product-sash"
        v-text="sash.replace('-', ' ')"></span>
</template>

```

Set the Tag Sash setting in the admin

1. Log in to the admin and navigate to `/admin/settings/manage/search`
2. Set your **Tag Sashes** the tag collection that you'd like to display as sashes
3. Save

Method 2: Using additional attributes

Setting the sash values in additional attributes gives you control over aspects of the sash, for example its colour or position. Don't forget, additional attributes can be set via a bulk action in the admin, which will save time applying these sashes to multiple products at once.

Update AppServiceProvider

Add the following namespace:

```
use Aero\Search\Elastic\Documents\ListingDocument;
```

Extend the listing document so that the sash data is available when looping over each listing on the listings page:

```
class AppServiceProvider extends ServiceProvider
{
    public function boot()
    {
        ListingDocument::add('search-result', function ($document) {
            $product = $document->getModel()->product;

            return [
                'sash' => [
                    'text' => $product->additional('sash-text'),
                    'colour' => $product->additional('sash-colour'),
                ],
            ];
        });
    }
}
```

Listing card snippet

Add the following to your `listing-card.twig` snippet:

```
<template v-if="listing.sash && listing.sash.text">
    <span class="aero-product-sash" :style="{backgroundColor: listing.sash.colour}" v-
text="listing.sash.text"></span>
</template>
```

Revision #1

Created 2026-09-02 13:36:27 UTC by Michael Price

Updated 2026-09-02 13:36:27 UTC by Michael Price