

Dashboard

- [What are the available lenses?](#)
- [How do I create a custom dashboard lens?](#)
- [How do I add a custom dashboard lens to the dashboard?](#)
- [How can I remove an existing lens from the dashboard?](#)

What are the available lenses?

Key	Class	Permission
revenue-lens	Aero\Admin\Lenses\RevenueLens	reports.orders
average-order-value-lens	Aero\Admin\Lenses\AverageOrderValueLens	reports.orders
top-shipping-countries-lens	Aero\Admin\Lenses\TopShippingCountriesLens	reports.orders
return-rate-lens	Aero\Admin\Lenses>ReturnsRateLens	reports.orders
orders-lens	Aero\Admin\Lenses\OrdersLens	reports.orders
top-selling-manufacturers-lens	Aero\Admin\Lenses\TopSellingManufacturersLens	reports.catalog
top-selling-items-lens	Aero\Admin\Lenses\TopSellingItemsLens	reports.catalog

The default dashboard lenses are stored in the `admin` configuration file, inside the `dashboard_lenses` array.

How do I create a custom dashboard lens?

To create a dashboard lens you need to create a class that extends `Aero\Admin\AdminLens` and implement the `data` method:

```
class OrdersLens extends \Aero\Admin\AdminLens
{
    protected $title = 'Orders';

    protected static $permission = 'reports.orders';

    protected static $dateGetVariables = [
        'custom' => 'ordered-at-date',
        'start' => 'ordered-at-date-start-date',
        'end' => 'ordered-at-date-end-date',
    ];

    public function data(\Illuminate\Http\Request $request, array $date, array $compare):
    \Illuminate\Http\JsonResponse
    {
        $ordersCount = \Aero\Cart\Models\Order::visible()->whereBetween('ordered_at',
        [$date['start'], $date['end']])->count();

        $ordersCompareCount = \Aero\Cart\Models\Order::visible()->whereBetween('ordered_at',
        [$compare['date']['start'], $compare['date']['end']])->count();

        if ($ordersCompareCount !== 0) {
            $percentage = (($ordersCount - $ordersCompareCount) / ($ordersCompareCount > 0 ?
            $ordersCompareCount : 1)) * 100;
        }

        return response()->json([
            'text' => number_format($ordersCount),
            'compare' => [
                'text' => $ordersCompareCount === 0 ? 'No data to compare' : $compare['text'],
```

```

        'percentage' => isset($percentage) ? ($percentage >= 0 ? '+' : '') .
number_format($percentage, 2) . '%' : '',
        'up' => isset($percentage) ? $percentage >= 0 : '',
    ],
    ]]);
}

protected function link(): ?array
{
    return auth()->user()->can('reports.view') ? [
        'route' => route('admin.report', ['key' => 'order-breakdown']),
        'text' => 'View Report',
    ] : null;
}
}

```

Parameters

Parameter	Type	Description
<code>\$request</code>	<code>\Illuminate\Http\Request</code>	The current request.
<code>\$date</code>	<code>array</code>	Contains the selected <code>start</code> & <code>end</code> dates.
<code>\$compare</code>	<code>array</code>	Contains the compare <code>date</code> (<code>start</code> & <code>end</code> nested instead) and the <code>text</code> .

How do I add a custom dashboard lens to the dashboard?

First, you must register your lens:

```
\Aero\Admin\Facades\Admin::registerLens(\Aero\Admin\Lenses\OrdersLens::class);
```

Then you can add it to the Dashboard:

```
\Aero\Admin\Http\Responses\AdminDashboardPage::extend(function
(\Aero\Admin\Http\Responses\AdminDashboardPage $page, \Closure $next) {
    // Add it as the first lens
    $page->addLens(
        \Aero\Admin\Lenses\OrdersLens::key(),
        \Aero\Admin\Lenses\OrdersLens::class
    );

    // Add it after a specific lens
    $page->addLensAfter(
        \Aero\Admin\Lenses\OrdersLens::key(),
        \Aero\Admin\Lenses\OrdersLens::class,
        \Aero\Admin\Lenses\RevenueLens::key()
    );

    // Add it before a specific lens
    $page->addLensBefore(
        \Aero\Admin\Lenses\OrdersLens::key(),
        \Aero\Admin\Lenses\OrdersLens::class,
        \Aero\Admin\Lenses\AverageOrderValueLens::key()
    );
});
```


How can I remove an existing lens from the dashboard?

You can remove an existing lens by using the `removeLens()` method and passing the `key` of the lens you want to remove:

```
\Aero\Admin\Http\Responses\AdminDashboardPage::extend(function
(\Aero\Admin\Http\Responses\AdminDashboardPage $page, \Closure $next) {
    $page->removeLens(
        \Aero\Admin\Lenses\OrdersLens::key()
    );
});
```