Informatics Ubaya

UBAYA
UNIVERSITAS SURABAYA

Web Framework Programming

Topic 6:
Designing your Website using Template

WEEK 06
Informatics Engineering
Universitas Surabaya

Outline

1

Templating and Layout in Web Development

2

Defining your Layout Template

3

Defining your Child Template

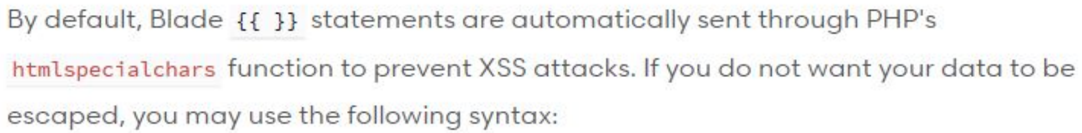
4

Implementation using Existing Case Study

Templating

Templating

- We will discuss how to use Custom HTML from a UI/UX designer into your Laravel Framework.
- We use Blade Template (https://laravel.com/docs/10.x/blade#introduction)
- Commonly used Template Syntax:
  - @section @endsection
  - @yield
  - @extends
  - {{ ….. }} -> displaying Data
  - {!! …….. !!}} -> Displaying Unescaped Data
  - @json (json encode to render json data)



By default, Blade `{{ }}` statements are automatically sent through PHP's `htmlspecialchars` function to prevent XSS attacks. If you do not want your data to be escaped, you may use the following syntax:

Templating

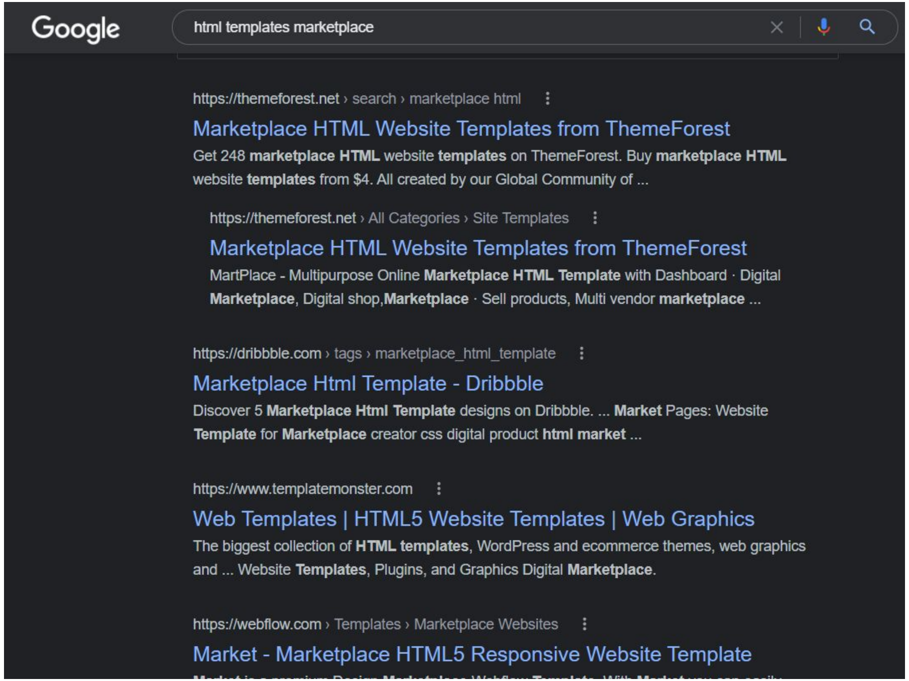
- Templates, which embedded in the Laravel Framework, have various types according to their functions.
- There are 2 types of templates: Administrative Templates (backend) and Public Templates (front-end).
- The backend is part of the portal that is used for the administration of all data and is accessed by the company's internals. Usually has a simple UI/UX but displays complete and detailed data.
- The front end is the part of the portal that is used to publish information to public. The main challenge is how to manage interactions with general users. Public Side has an attractive UI/UX and accommodated various kinds of interactions
- The technique of installing templates both Front-end and Backend has the same method.

Practice #1: Installing Templates in Our Project

Steps

1. **Search & download** HTML templates
2. **Extract** the HTML template source code
3. **Choose and Recognize Layouts** that are used as the basic UI framework
4. **Slicing Layout** to determine which is the parent (base layout) and child blade
5. **Apply Layouts and Child Templates** in Laravel's Resources View
6. **Customize child templates** with existing templates
7. **Integrate child templates** with layout templates with @yield and @extends syntax
8. **Integrate template assets** in Laravel project path with @asset . syntax

1. Find your Template



html templates marketplace

https://themeforest.net › search › marketplace html

Marketplace HTML Website Templates from ThemeForest

Get 248 **marketplace HTML** website **templates** on ThemeForest. Buy **marketplace HTML** website **templates** from $4. All created by our Global Community of ...

https://themeforest.net › All Categories › Site Templates

Marketplace HTML Website Templates from ThemeForest

MartPlace - Multipurpose Online **Marketplace HTML Template** with Dashboard · Digital **Marketplace**, Digital shop,**Marketplace** · Sell products, Multi vendor **marketplace** ...

https://dribbble.com › tags › marketplace_html_template

Marketplace Html Template - Dribbble

Discover 5 **Marketplace Html Template** designs on Dribbble. ... **Market** Pages: Website **Template** for **Marketplace** creator css digital product **html market** ...

https://www.templatemonster.com

Web Templates | HTML5 Website Templates | Web Graphics

The biggest collection of **HTML templates**, WordPress and ecommerce themes, web graphics and ... Website **Templates**, Plugins, and Graphics Digital **Marketplace**.

https://webflow.com › Templates › Marketplace Websites

Market - Marketplace HTML5 Responsive Website Template

You can get custom Template from any resources like: themeforest, templatemonster, and so on.

For this lesson, we already provided 1 template for you.

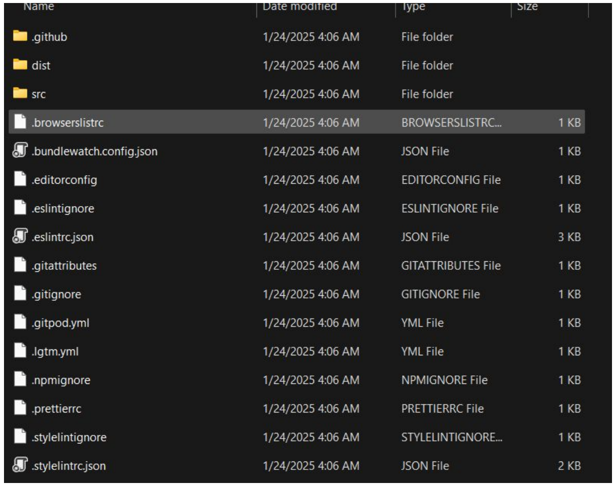
We use: AdminLTE

FILE

AdminLTE Template

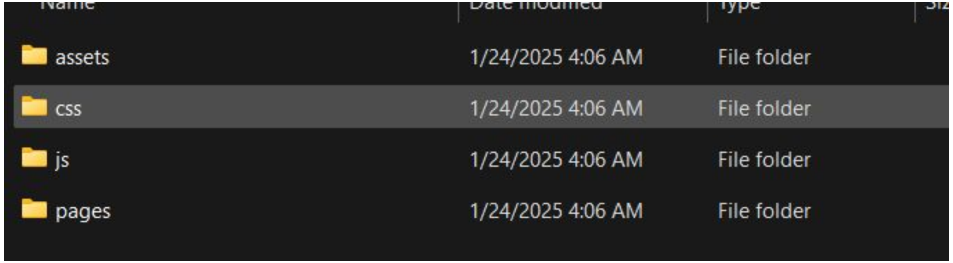
2. Extract source HTML

AdminLTE-master.zip



Name	Date modified	Type	Size
.github	1/24/2025 4:06 AM	File folder	
dist	1/24/2025 4:06 AM	File folder	
src	1/24/2025 4:06 AM	File folder	
.browserslistrc	1/24/2025 4:06 AM	BROWSERSLISTRC...	1 KB
.bundlewatch.config.json	1/24/2025 4:06 AM	JSON File	1 KB
.editorconfig	1/24/2025 4:06 AM	EDITORCONFIG File	1 KB
.eslintignore	1/24/2025 4:06 AM	ESLINTIGNORE File	1 KB
.eslintrc.json	1/24/2025 4:06 AM	JSON File	3 KB
.gitattributes	1/24/2025 4:06 AM	GITATTRIBUTES File	1 KB
.gitignore	1/24/2025 4:06 AM	GITIGNORE File	1 KB
.gitpod.yml	1/24/2025 4:06 AM	YML File	1 KB
.lgtm.yml	1/24/2025 4:06 AM	YML File	1 KB
.npmignore	1/24/2025 4:06 AM	NPMIGNORE File	1 KB
.prettierrc	1/24/2025 4:06 AM	PRETTIERRC File	1 KB
.stylelintignore	1/24/2025 4:06 AM	STYLELINTIGNORE...	1 KB
.stylelintrc.json	1/24/2025 4:06 AM	JSON File	2 KB

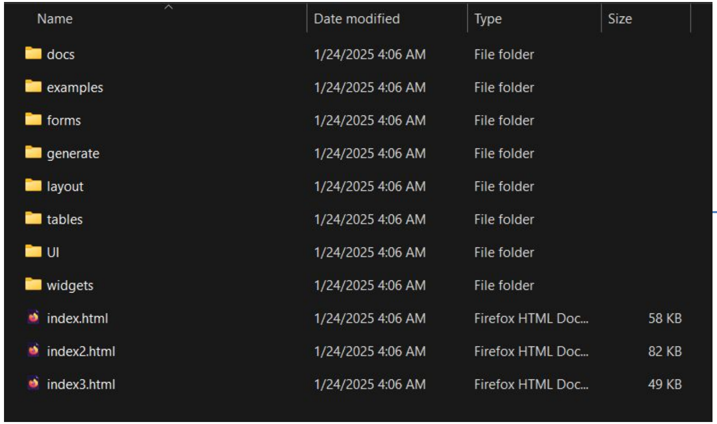
1. Copy and extract this zip
2. In AdminLTE, our layout is located on dist/pages.
3. Copy your element & assets template into directory **folder.** In this case: <u>assets</u>, <u>css</u> and <u>js</u> inside **dist** directory



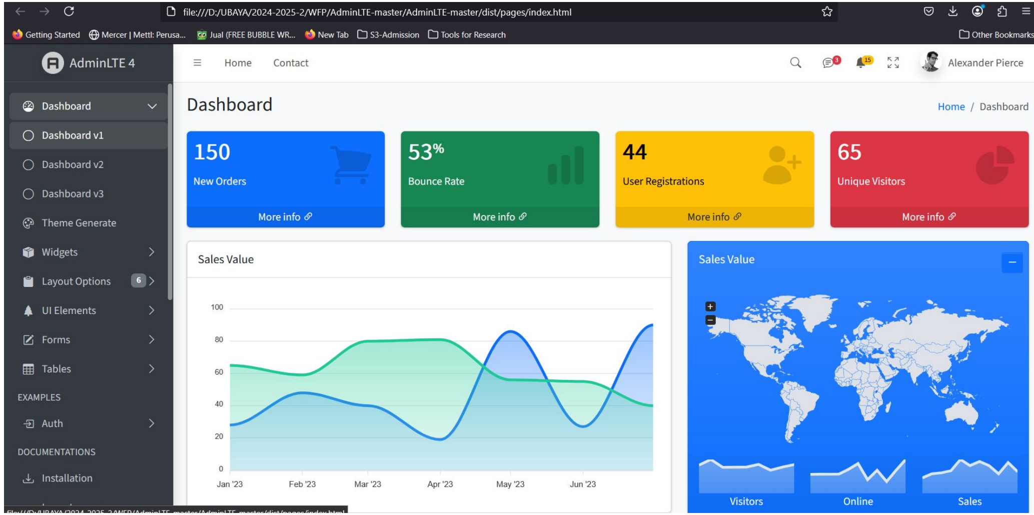
Name	Date modified	Type	Siz
assets	1/24/2025 4:06 AM	File folder	
css	1/24/2025 4:06 AM	File folder	
js	1/24/2025 4:06 AM	File folder	
pages	1/24/2025 4:06 AM	File folder	

2. Extract source HTML (2)

Open your template and we analyze our `main` template. Open `index.html` in a web browser



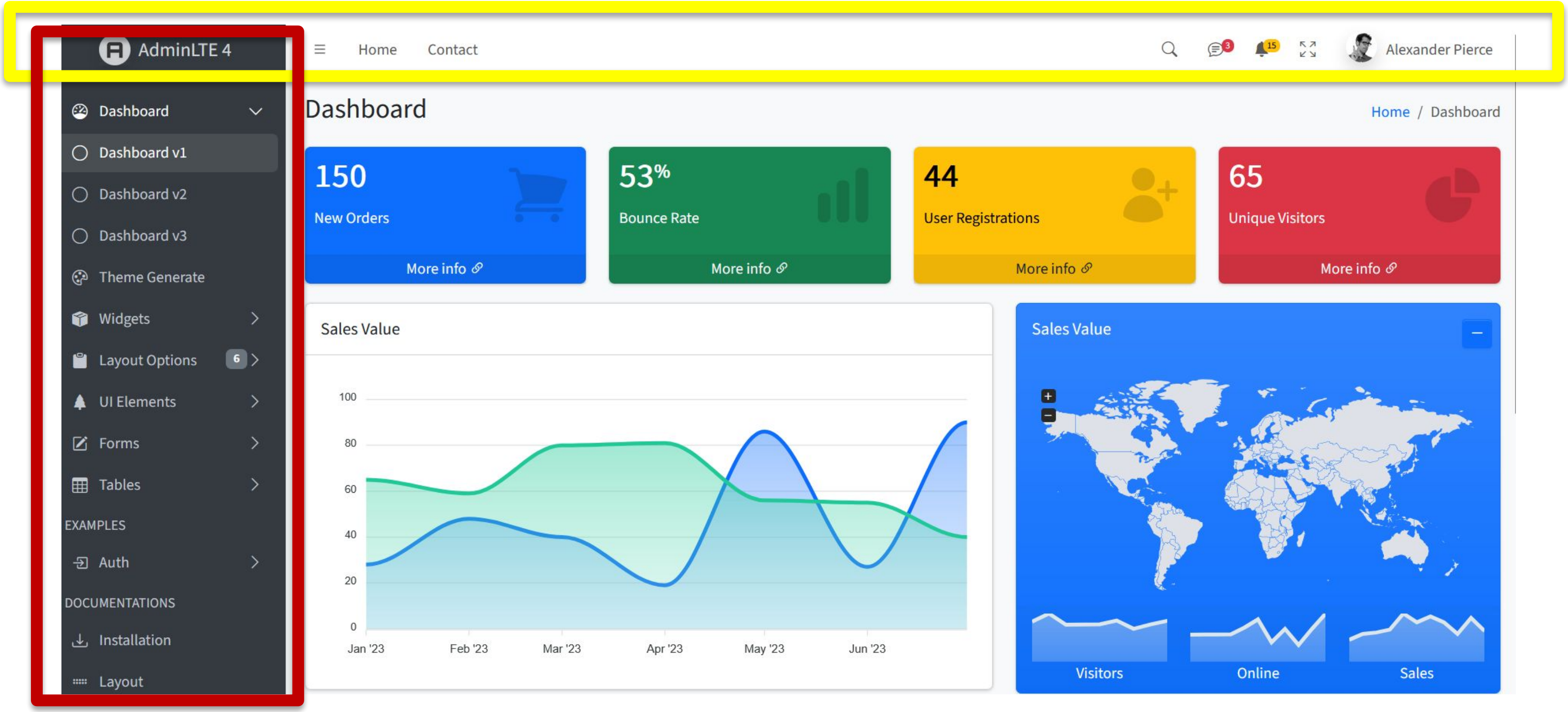
Name	Date modified	Type	Size
docs	1/24/2025 4:06 AM	File folder	
examples	1/24/2025 4:06 AM	File folder	
forms	1/24/2025 4:06 AM	File folder	
generate	1/24/2025 4:06 AM	File folder	
layout	1/24/2025 4:06 AM	File folder	
tables	1/24/2025 4:06 AM	File folder	
UI	1/24/2025 4:06 AM	File folder	
widgets	1/24/2025 4:06 AM	File folder	
index.html	1/24/2025 4:06 AM	Firefox HTML Doc...	58 KB
index2.html	1/24/2025 4:06 AM	Firefox HTML Doc...	82 KB
index3.html	1/24/2025 4:06 AM	Firefox HTML Doc...	49 KB



file:///D:/UBAYA/2024-2025-2/WFP/AdminLTE-master/AdminLTE-master/dist/pages/index.html

Informatics Engineering | Universitas Surabaya

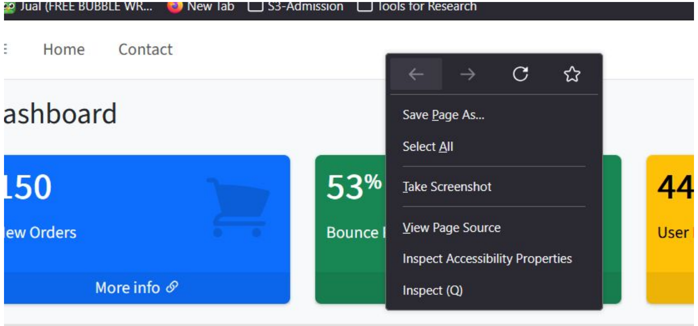
3. Identify your Layout



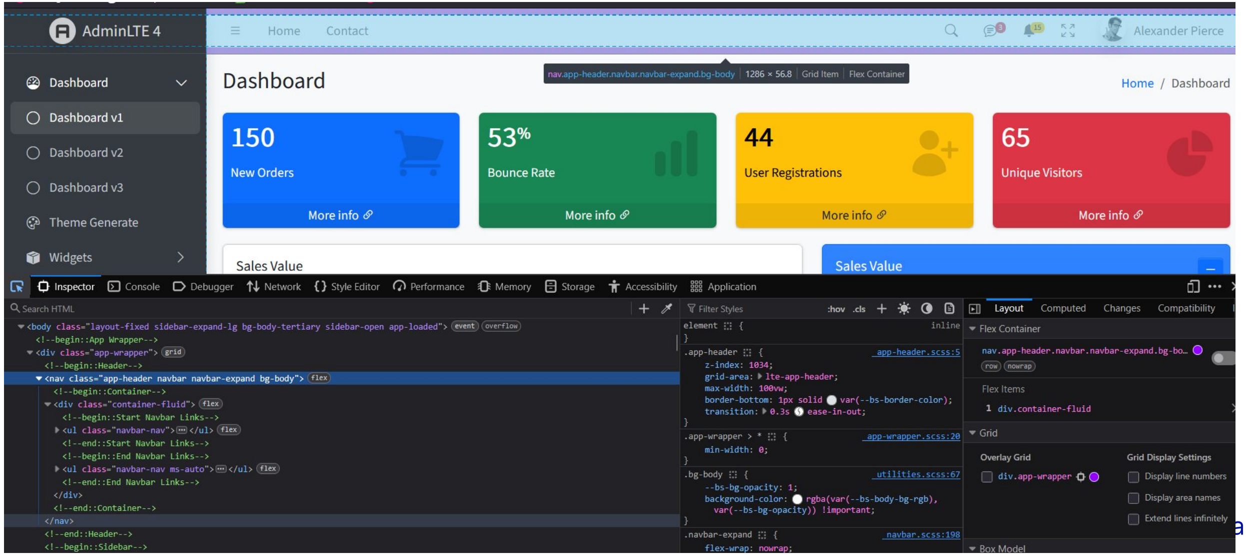
Layout Analysis: Headers

Use the Inspect Element help tool in each browser.
The **header** in AdminLTE is found as follows
The syntax or class declaring **header** is

<nav class="app-header navbar navbar-expand bg-body">



Use [Pick inspector] (Ctrl+Shift+C) at Firefox



```
<body class="layout-fixed sidebar-expand-lg bg-body-tertiary sidebar-open app-loaded"> event overflow
  <!--begin::App Wrapper-->
  <div class="app-wrapper"> grid
    <!--begin::Header-->
    <nav class="app-header navbar navbar-expand bg-body"> flex
      <!--begin::Container-->
      <div class="container-fluid"> flex
        <!--begin::Start Navbar Links-->
        <ul class="navbar-nav">…</ul> flex
        <!--end::Start Navbar Links-->
        <!--begin::End Navbar Links-->
        <ul class="navbar-nav ms-auto">…</ul> flex
        <!--end::End Navbar Links-->
      </div>
      <!--end::Container-->
    </nav>
    <!--end::Header-->
    <!--begin::Sidebar-->
```

```
element {                                              inline
}
.app-header {                                app-header.scss:5
    z-index: 1034;
    grid-area: lte-app-header;
    max-width: 100vw;
    border-bottom: 1px solid var(--bs-border-color);
    transition: 0.3s ease-in-out;
}
.app-wrapper > * {                         app-wrapper.scss:20
    min-width: 0;
}
.bg-body {                                  utilities.scss:67
    --bs-bg-opacity: 1;
    background-color: rgba(var(--bs-body-bg-rgb),
      var(--bs-bg-opacity)) !important;
}
.navbar-expand {                              navbar.scss:198
    flex-wrap: nowrap;
```

Flex Container

nav.app-header.navbar.navbar-expand.bg-bo… row nowrap

Flex Items

1 div.container-fluid

Grid

Overlay Grid

□ div.app-wrapper

Grid Display Settings

□ Display line numbers

□ Display area names

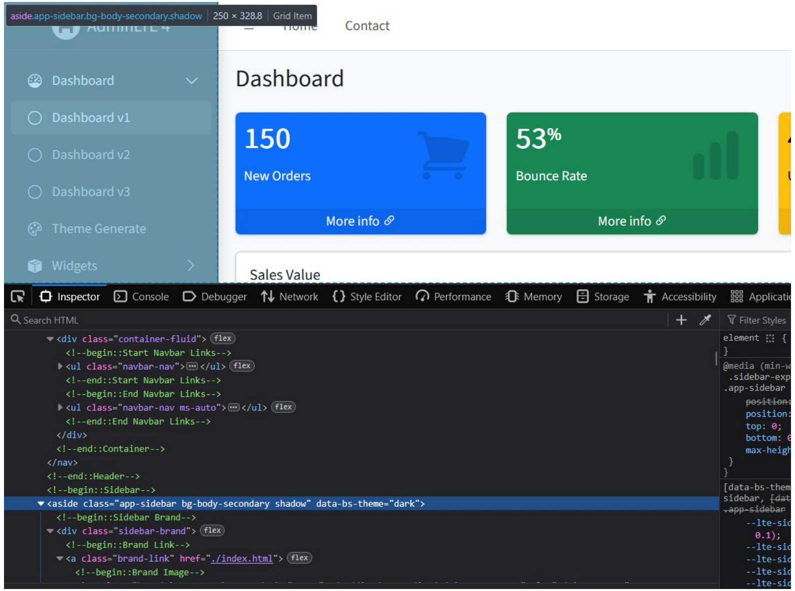
□ Extend lines infinitely

Box Model

Layout Analysis: Sidebar

- Use the Inspect Element help tool in each browser.
- Sidebar on AdminLTE

The syntax or class declaring it left-menu is a
<aside class="app-sidebar bg-body-secondary shadow" data-bs-theme="dark">

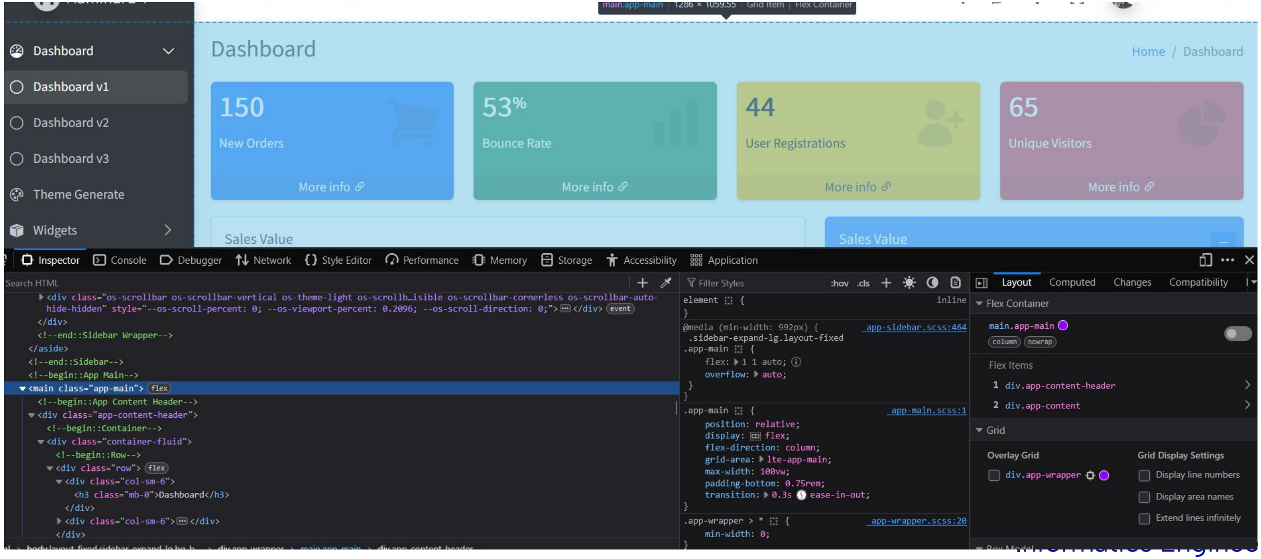


```
<div class="container-fluid"> flex
  <!--begin::Start Navbar Links-->
  <ul class="navbar-nav">…</ul> flex
  <!--end::Start Navbar Links-->
  <!--begin::End Navbar Links-->
  <ul class="navbar-nav ms-auto">…</ul> flex
  <!--end::End Navbar Links-->
 </div>
 <!--end::Container-->
</nav>
<!--end::Header-->
<!--begin::Sidebar-->
<aside class="app-sidebar bg-body-secondary shadow" data-bs-theme="dark">
 <!--begin::Sidebar Brand-->
 <div class="sidebar-brand"> flex
  <!--begin::Brand Link-->
  <a class="brand-link" href="./index.html"> flex
   <!--begin::Brand Image-->
```

```
element {
}
@media (min-w
 .sidebar-exp
.app-sidebar
    ~~position:~~
    position:
    top: 0;
    bottom: 0
    max-heigh
 }
}
[data-bs-them
sidebar, ~~[dat~~
~~.app-sidebar~~
    --lte-sid
      0.1);
    --lte-sid
    --lte-sid
    --lte-sid
    --lte-sid
```

Layout Analysis: Content

Use the Inspect Element help tool in each browser.
The **Content** on AdminLTE was found as follows
The syntax or class that declares **content** is …
<main class="app-main">



```
▸<div class="os-scrollbar os-scrollbar-vertical os-theme-light os-scrollb…isible os-scrollbar-cornerless os-scrollbar-auto-
 hide-hidden" style="--os-scroll-percent: 0; --os-viewport-percent: 0.2096; --os-scroll-direction: 0;">⋯</div> event
</div>
<!--end::Sidebar Wrapper-->
</aside>
<!--end::Sidebar-->
<!--begin::App Main-->
▾<main class="app-main"> flex
  <!--begin::App Content Header-->
 ▾<div class="app-content-header">
    <!--begin::Container-->
  ▾<div class="container-fluid">
      <!--begin::Row-->
    ▾<div class="row"> flex
      ▾<div class="col-sm-6">
          <h3 class="mb-0">Dashboard</h3>
        </div>
      ▸<div class="col-sm-6">⋯</div>
      </div>
```

```
element {                                          inline
}
@media (min-width: 992px) {          app-sidebar.scss:464
 .sidebar-expand-lg.layout-fixed
.app-main {
    flex: ▸1 1 auto;
    overflow: ▸ auto;
 }
}
.app-main {                              app-main.scss:1
    position: relative;
    display: flex;
    flex-direction: column;
    grid-area: ▸ lte-app-main;
    max-width: 100vw;
    padding-bottom: 0.75rem;
    transition: ▸ 0.3s ease-in-out;
}
.app-wrapper > * {                   app-wrapper.scss:20
    min-width: 0;
}
```

Flex Container
main.app-main
column nowrap
Flex Items
1 div.app-content-header
2 div.app-content
Grid
Overlay Grid
□ div.app-wrapper
Grid Display Settings
□ Display line numbers
□ Display area names
□ Extend lines infinitely

3. Identify Layout (Summary)

- The AdminLTE Admin template has 3 parts, namely
  - Header
  - Sidebar
  - Content
- When choosing a different menu, only the content section always show different content based on the selected menu.
- When selecting a different menu, the 'sidebar' section has minor change in content. Only different CSS for the active menu
- When choosing a different menu, the 'header' section has minor change

- **Hence, the ‘content' part must be sliced from the main layout**

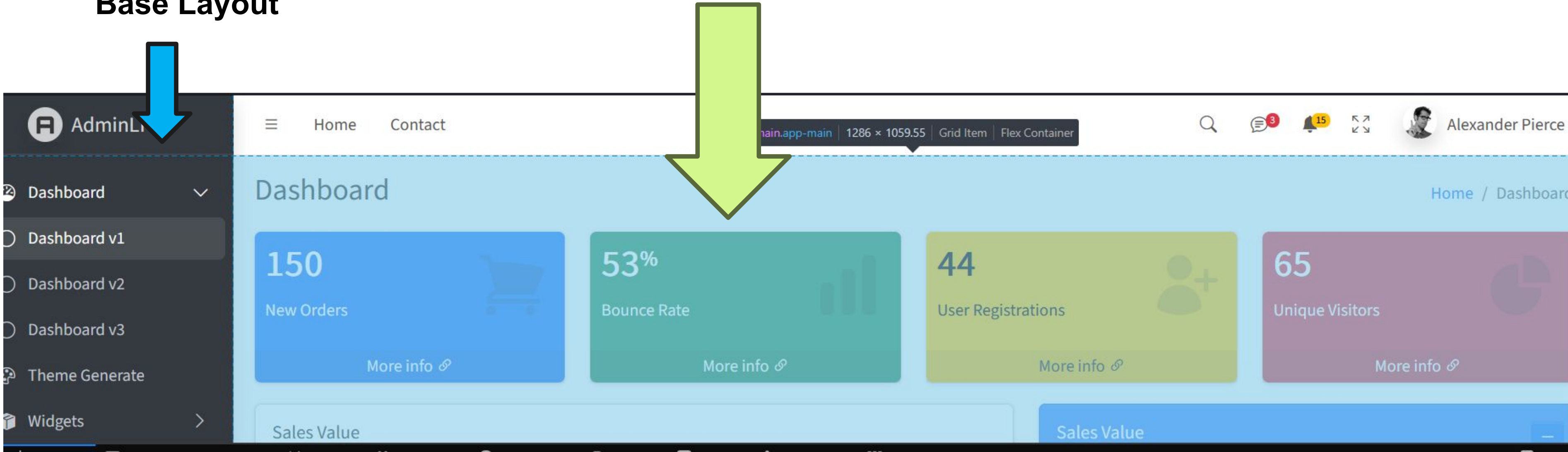
4. Define Base Layout and Child Layout

Based on the previous work steps, we can define:

- **Child Layout** is the program snippet from the inside ‘content’
- **Base Layout** is the entire HTML code of the 'index.html' file except ‘content’ section

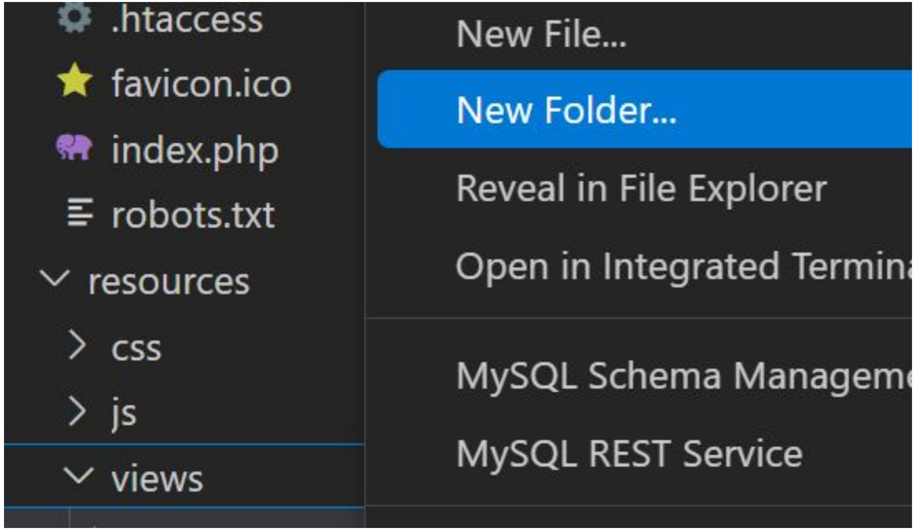
Child Layout

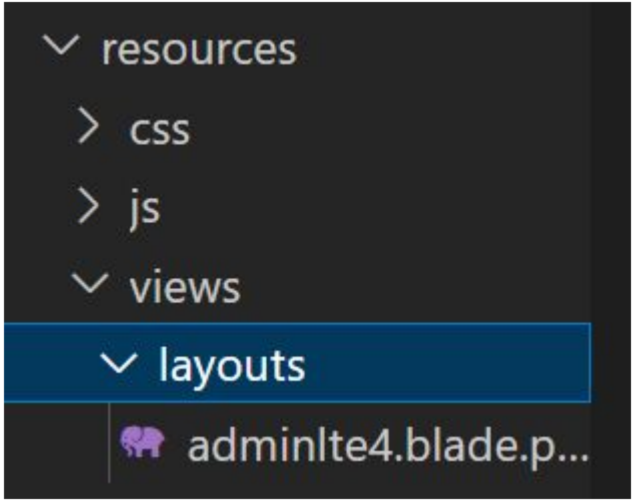
Base Layout



5. Apply Layouts and Child Templates on Laravel Resources View (1)

Create a folder called 'layouts' in resources/views

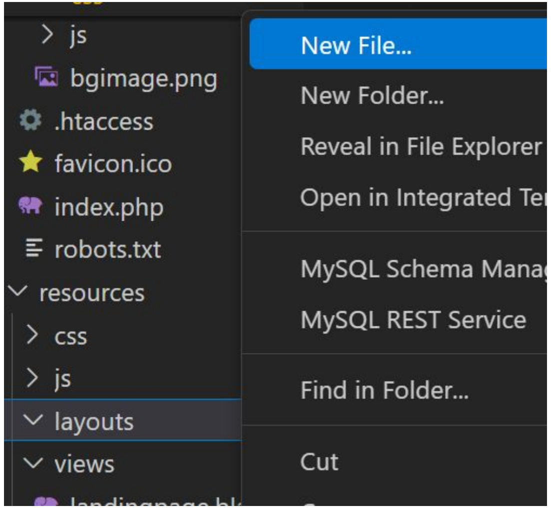


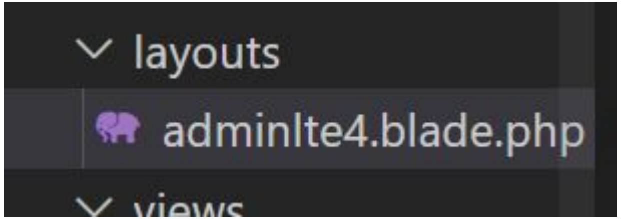


5. Apply Layouts and Child Templates on Laravel Resources View (2)

Create a new blade file.
(extension: *.blade.php)

Filenames are basically free, but are recommended according to the `theme name`

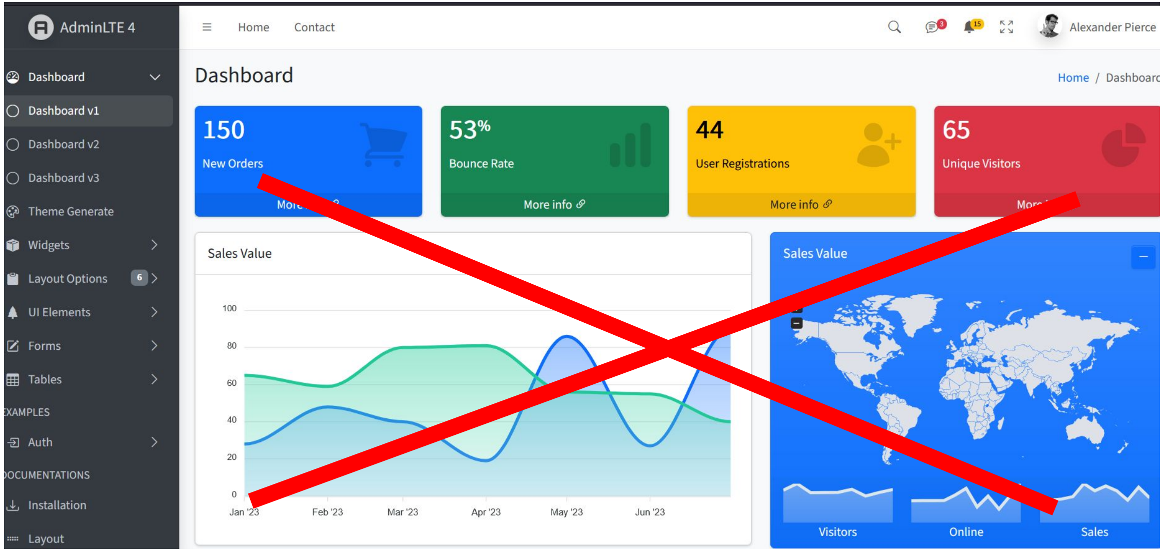




5. Apply Layouts and Child Templates on Laravel Resources View (3)

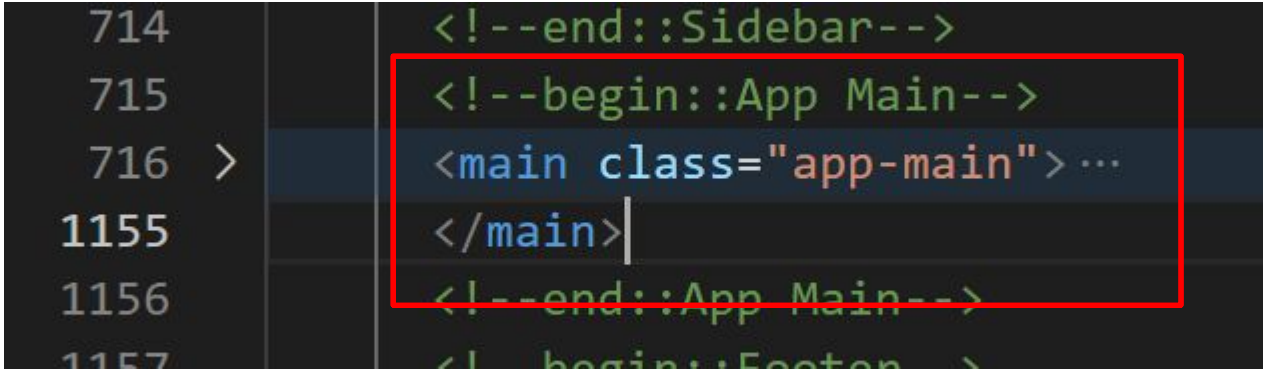
Copy all the HTML scripts in index.html and **paste** them into layouts `adminlte4.blade.php`

(see next slide) Move the `content` section and move script html into `child layout` (see next slide)

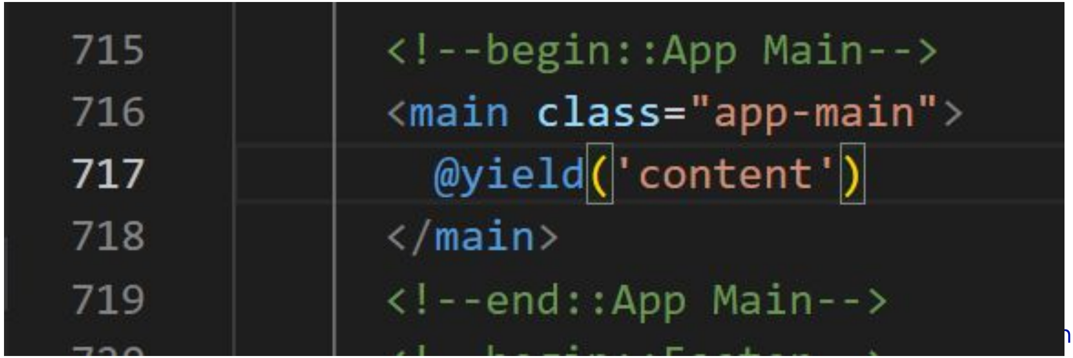


5. Apply Layouts and Child Templates on Laravel Resources View (4)

Collapse <main class> ‘page-content’ and remove “….”. Because we substitute with @yield



```
<!--end::Sidebar-->
<!--begin::App Main-->
<main class="app-main">···
</main>
<!--end::App Main-->
<!--begin::Footer-->
```



```
<!--begin::App Main-->
<main class="app-main">
  @yield('content')
</main>
<!--end::App Main-->
<!--begin::Footer-->
```

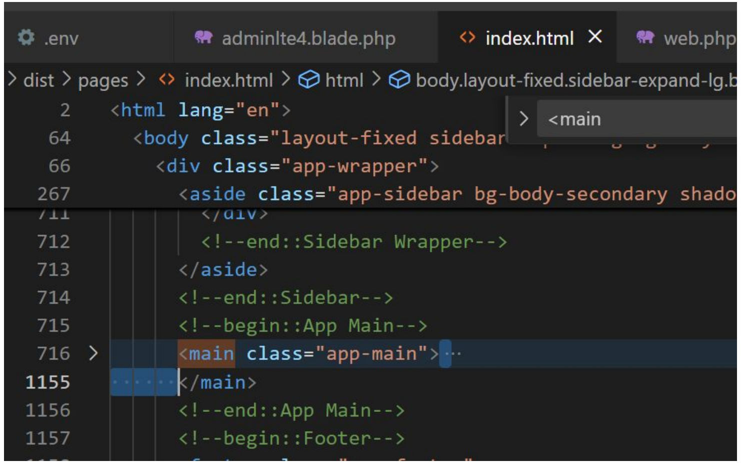
6. Integrate child layout with our `main template` using @yield, @section and @extends syntax

In the main-layout file, named `adminlte4.blade.php`, please add @yield('content’) near the red boxed (previous slide) to indicate that the section will be filled with a child.

In the beginning of child-layout file, programmer must create @section syntax to synchronize the child & main layout.

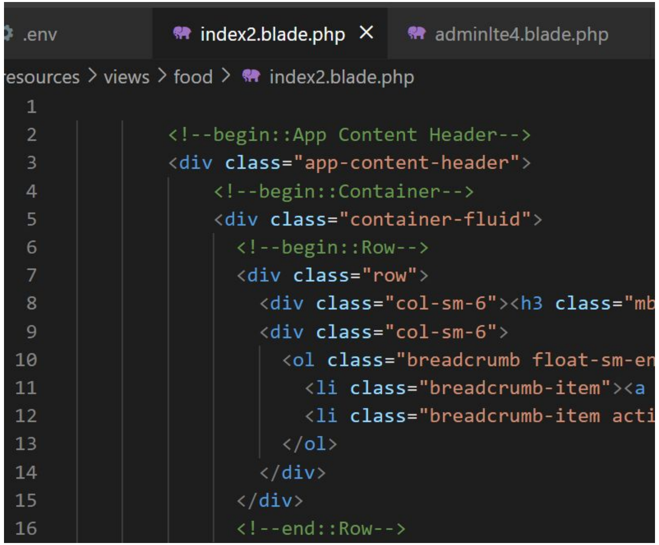
Inside the section of `child-layout` please paste the script content inside the red-boxed (from the previous slide) and modify the page title to make it more appropriate

From your template, cut your content data


.env
adminlte4.blade.php
index.html
web.php
dist > pages > index.html > html > body.layout-fixed.sidebar-expand-lg.b
> <main

```
<html lang="en">
 <body class="layout-fixed sidebar
  <div class="app-wrapper">
   <aside class="app-sidebar bg-body-secondary shado
    </div>
    <!--end::Sidebar Wrapper-->
   </aside>
   <!--end::Sidebar-->
   <!--begin::App Main-->
   <main class="app-main">···
   </main>
   <!--end::App Main-->
   <!--begin::Footer-->
```

Cut (Ctrl+X)

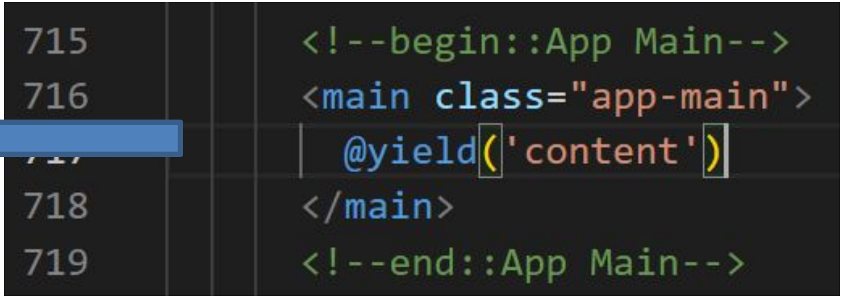


```

<!--begin::App Content Header-->
<div class="app-content-header">
  <!--begin::Container-->
  <div class="container-fluid">
   <!--begin::Row-->
   <div class="row">
    <div class="col-sm-6"><h3 class="mb
    <div class="col-sm-6">
     <ol class="breadcrumb float-sm-en
      <li class="breadcrumb-item"><a
      <li class="breadcrumb-item acti
     </ol>
    </div>
   </div>
   <!--end::Row-->
```

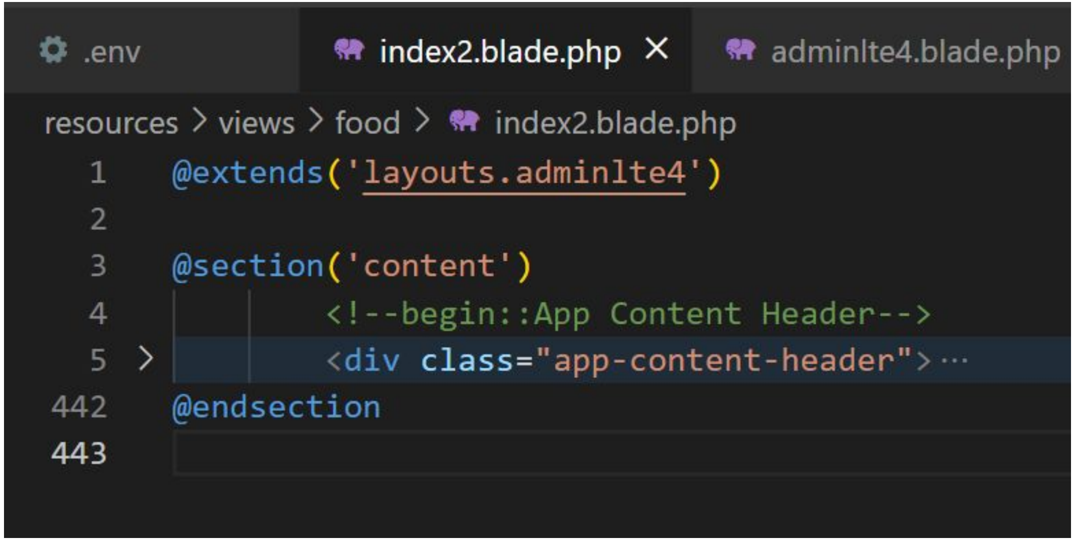
New Child File
(ex: index2.blade.php)

```
@extends('layouts.adminlte4’)

@section('content')
<!--begin::App Content Header-->
        <div class="app-content-header">
         …..
       </div>
@endsection
```

In adminlte4.blade.php



```
<!--begin::App Main-->
<main class="app-main">
  @yield('content')
</main>
<!--end::App Main-->
```



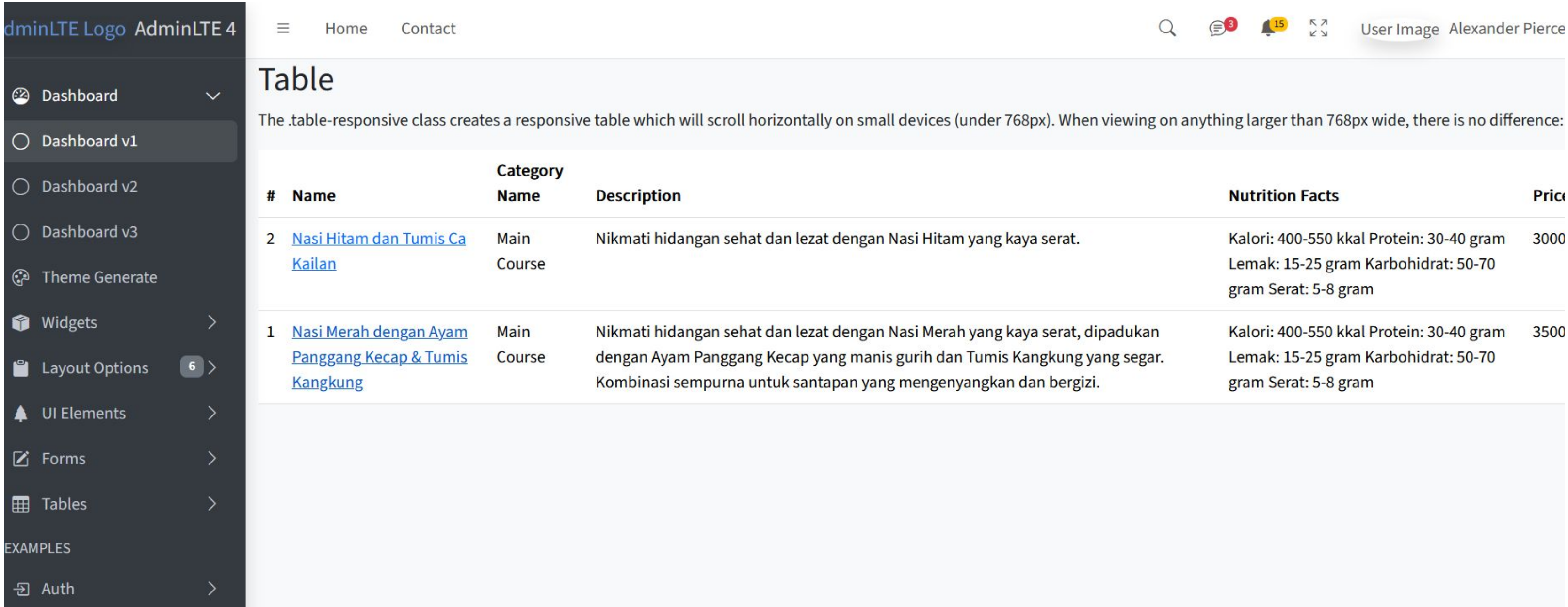
```
@extends('layouts.adminlte4')

@section('content')
        <!--begin::App Content Header-->
        <div class="app-content-header">···
@endsection
```

Practice #2: Implement the Template in Index of Food View

Adapt your Food/index.blade.php into your Template

Add @extends and @section
Also take the page-title and page-bar and modify it to fit the contents of the page

Expected Results :

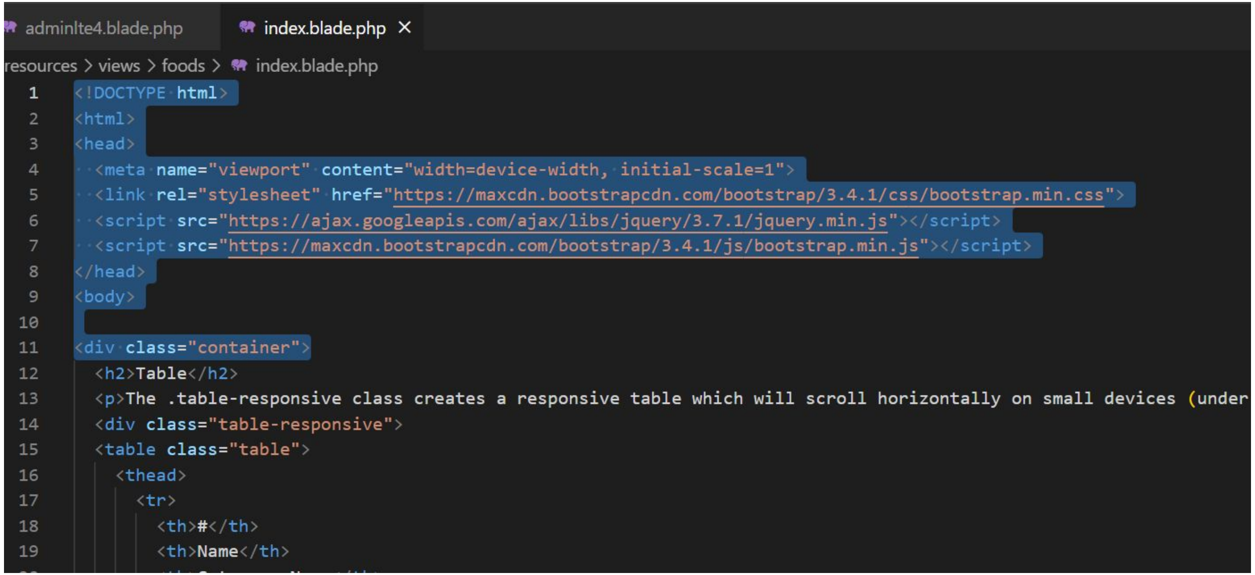


Table

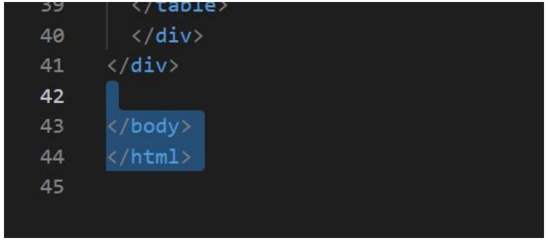
The .table-responsive class creates a responsive table which will scroll horizontally on small devices (under 768px). When viewing on anything larger than 768px wide, there is no difference:

#	Name	Category Name	Description	Nutrition Facts	Pric
2	Nasi Hitam dan Tumis Ca Kailan	Main Course	Nikmati hidangan sehat dan lezat dengan Nasi Hitam yang kaya serat.	Kalori: 400-550 kkal Protein: 30-40 gram Lemak: 15-25 gram Karbohidrat: 50-70 gram Serat: 5-8 gram	3000
1	Nasi Merah dengan Ayam Panggang Kecap & Tumis Kangkung	Main Course	Nikmati hidangan sehat dan lezat dengan Nasi Merah yang kaya serat, dipadukan dengan Ayam Panggang Kecap yang manis gurih dan Tumis Kangkung yang segar. Kombinasi sempurna untuk santapan yang mengenyangkan dan bergizi.	Kalori: 400-550 kkal Protein: 30-40 gram Lemak: 15-25 gram Karbohidrat: 50-70 gram Serat: 5-8 gram	3500

Remove non-essential content



```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container">
  <h2>Table</h2>
  <p>The .table-responsive class creates a responsive table which will scroll horizontally on small devices (under
  <div class="table-responsive">
  <table class="table">
    <thead>
      <tr>
        <th>#</th>
        <th>Name</th>
```

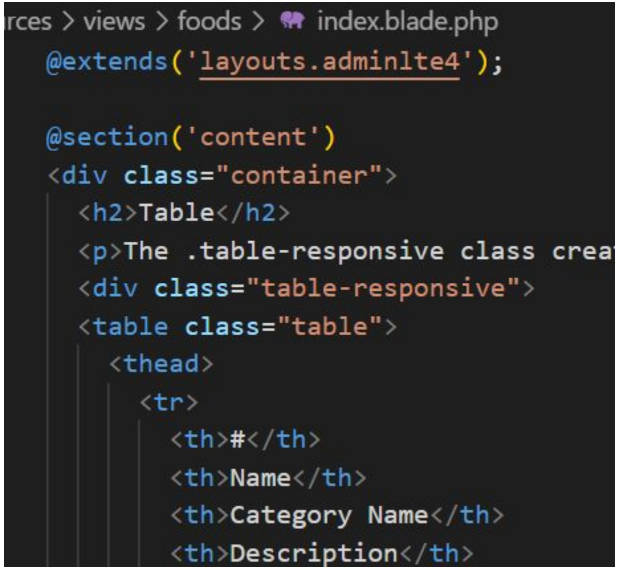


```
  </table>
  </div>
</div>

</body>
</html>
```

Apply Layouts and Child Templates in Laravel's Resources View

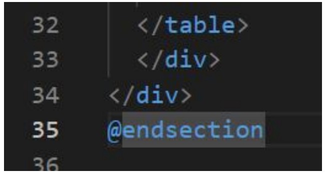
Extend the parent/base layout and add a section for the content.

Put your content script data inside the section(‘content’)



rces > views > foods > index.blade.php

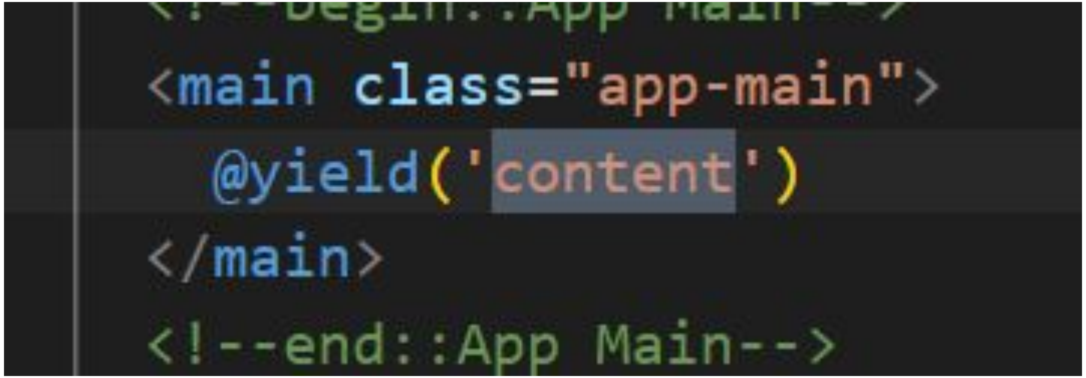
```
@extends('layouts.adminlte4');

@section('content')
<div class="container">
  <h2>Table</h2>
  <p>The .table-responsive class crea
  <div class="table-responsive">
  <table class="table">
    <thead>
      <tr>
        <th>#</th>
        <th>Name</th>
        <th>Category Name</th>
        <th>Description</th>
```



```
    </table>
    </div>
  </div>
  @endsection
```

Integrate child templates with layout templates with @yield and @extends syntax

Don’t forget to check your `main-layout` and `child-layout` has similar name



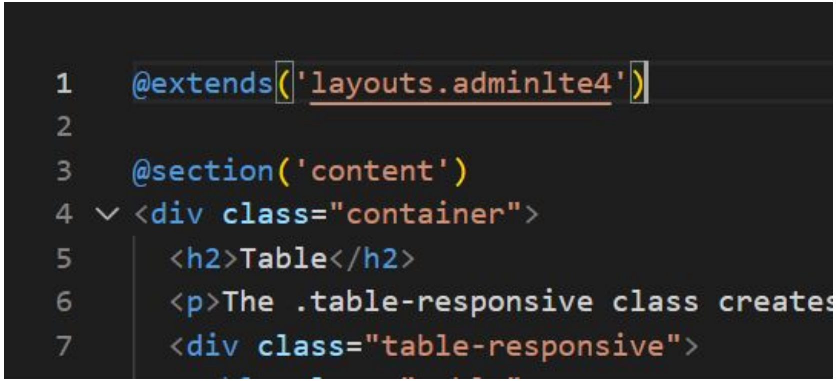
```
<!--begin::App Main-->
<main class="app-main">
  @yield('content')
</main>
<!--end::App Main-->
```

Integrate child templates with layout templates with @yield and @extends syntax (2)

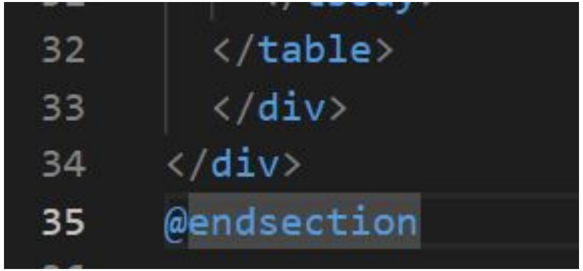
In the child-layout file (eg: food/index.blade.php), it is written **@extends** and **@section**.

@extends has a layout name parameter that is used as a benchmark.
@extends is placed on top of the same as import in the controller file.

@section has a yield name parameter to be entered.



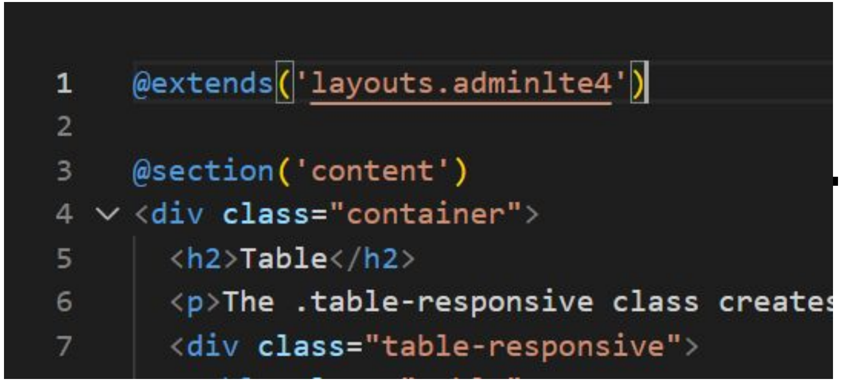
```
@extends('layouts.adminlte4')

@section('content')
<div class="container">
  <h2>Table</h2>
  <p>The .table-responsive class creates
  <div class="table-responsive">
```



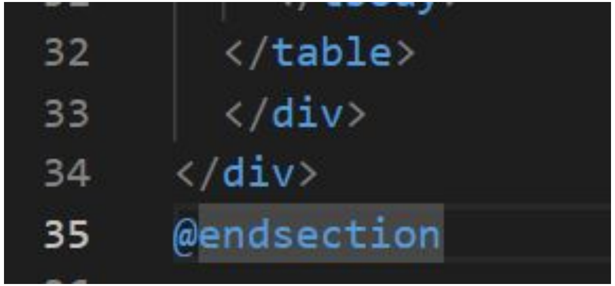
```
    </table>
    </div>
  </div>
  @endsection
```

Integrate child templates with layout templates with @yield and @extends syntax (3)

Add @section ('content') on the 3rd line (above the container-fluid class div because it indicates that from line 4 to @endsection will be pasted on @yield('content') in layouts.

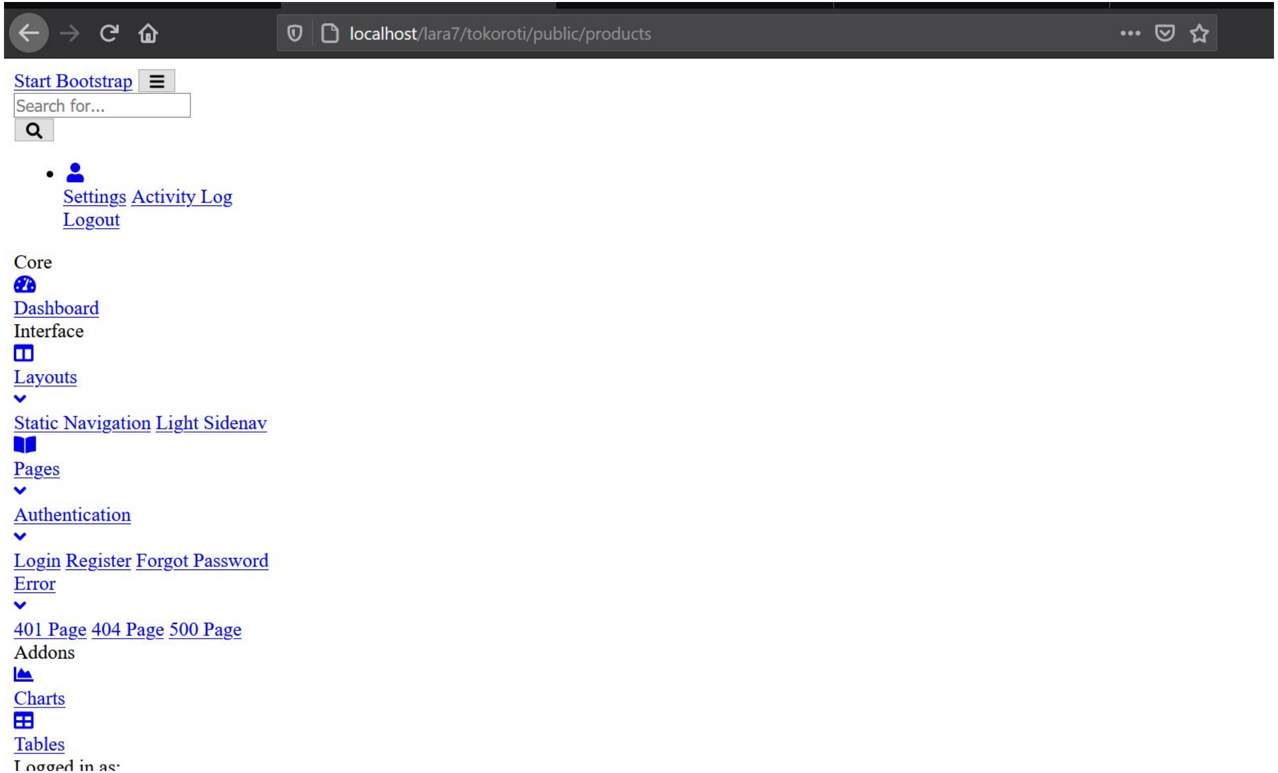


```
1   @extends('layouts.adminlte4')
2
3   @section('content')
4   <div class="container">
5     <h2>Table</h2>
6     <p>The .table-responsive class creates
7     <div class="table-responsive">
```



```
32      </table>
33      </div>
34    </div>
35    @endsection
```

Result



localhost/lara7/tokoroti/public/products

Start Bootstrap ☰

Search for...

🔍

- 👤
  Settings Activity Log
  Logout

Core

Dashboard

Interface

Layouts

Static Navigation Light Sidenav

Pages

Authentication

Login Register Forgot Password

Error

401 Page 404 Page 500 Page

Addons

Charts

Tables

Logged in as:

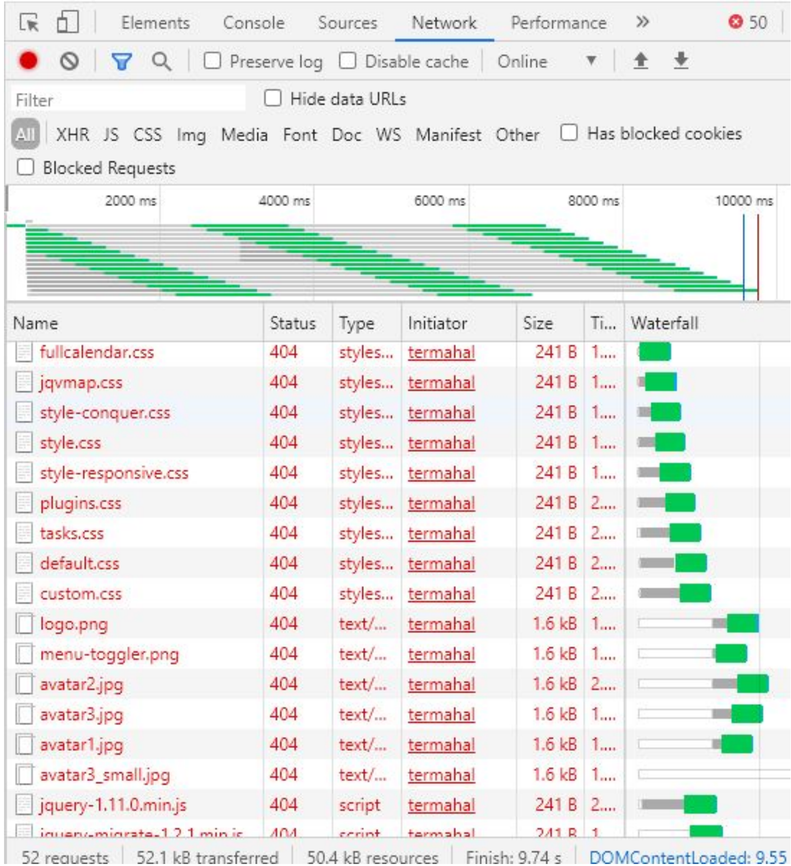
What’s wrong?

Your template isn’t load correctly. Please check your “Inspect Element” and “Network tab”.

This case caused by your asset template is missing.

The components of template are

- Asset css
- Asset js
- Asset img



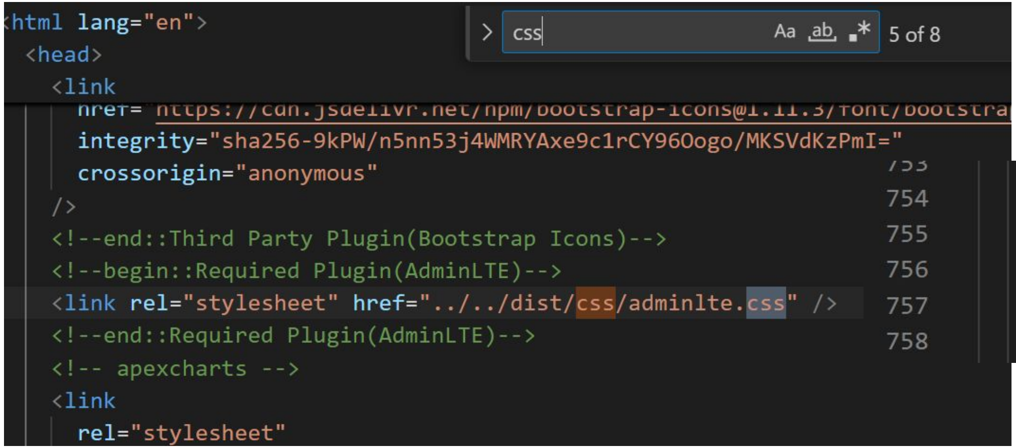
Name	Status	Type	Initiator	Size	Ti...	Waterfall
fullcalendar.css	404	styles...	termahal	241 B	1....	
jqvmap.css	404	styles...	termahal	241 B	1....	
style-conquer.css	404	styles...	termahal	241 B	1....	
style.css	404	styles...	termahal	241 B	1....	
style-responsive.css	404	styles...	termahal	241 B	1....	
plugins.css	404	styles...	termahal	241 B	2....	
tasks.css	404	styles...	termahal	241 B	2....	
default.css	404	styles...	termahal	241 B	2....	
custom.css	404	styles...	termahal	241 B	2....	
logo.png	404	text/...	termahal	1.6 kB	1....	
menu-toggler.png	404	text/...	termahal	1.6 kB	1....	
avatar2.jpg	404	text/...	termahal	1.6 kB	2....	
avatar3.jpg	404	text/...	termahal	1.6 kB	1....	
avatar1.jpg	404	text/...	termahal	1.6 kB	1....	
avatar3_small.jpg	404	text/...	termahal	1.6 kB	1....	
jquery-1.11.0.min.js	404	script	termahal	241 B	2....	
jquery-migrate-1.2.1.min.js	404	script	termahal	241 B	1.	

52 requests | 52.1 kB transferred | 50.4 kB resources | Finish: 9.74 s | DOMContentLoaded: 9.55

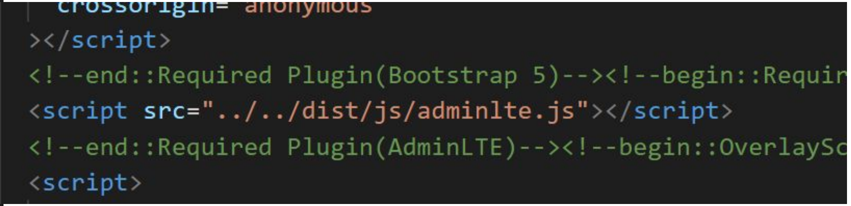
SOLUTION: Install Asset JS/CSS template with @asset syntax

The solution is type “asset()” function before and after `all import js/css/image` scripts in layouts template.

Usually import js/css scripts are located at the beginning (header) of html and footer of html

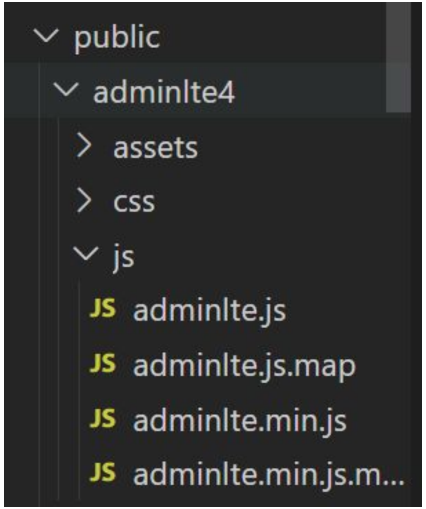


```
<html lang="en">
  <head>
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstra
      integrity="sha256-9kPW/n5nn53j4WMRYAxe9c1rCY96Oogo/MKSVdKzPmI="
      crossorigin="anonymous"
    />
    <!--end::Third Party Plugin(Bootstrap Icons)-->
    <!--begin::Required Plugin(AdminLTE)-->
    <link rel="stylesheet" href="../../dist/css/adminlte.css" />
    <!--end::Required Plugin(AdminLTE)-->
    <!-- apexcharts -->
    <link
      rel="stylesheet"
```

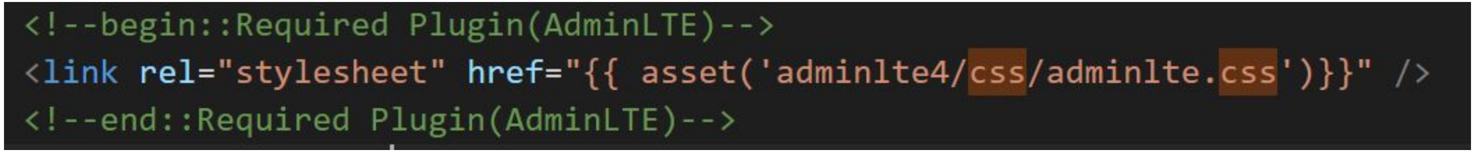


```
  crossorigin="anonymous"
></script>
<!--end::Required Plugin(Bootstrap 5)--><!--begin::Requir
<script src="../../dist/js/adminlte.js"></script>
<!--end::Required Plugin(AdminLTE)--><!--begin::OverlaySc
<script>
```

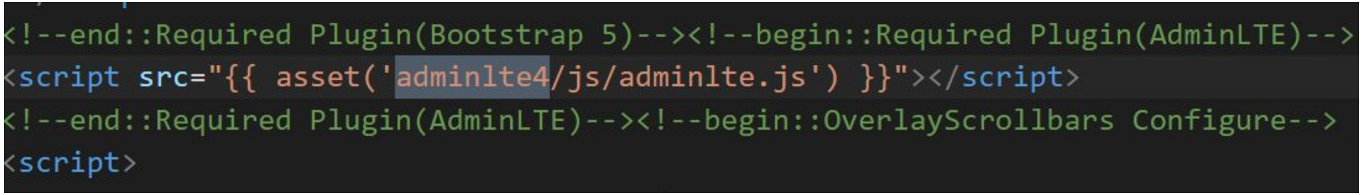
The Result is



```
public
  adminlte4
    assets
    css
    js
      adminlte.js
      adminlte.js.map
      adminlte.min.js
      adminlte.min.js.m...
```



```
<!--begin::Required Plugin(AdminLTE)-->
<link rel="stylesheet" href="{{ asset('adminlte4/css/adminlte.css')}}" />
<!--end::Required Plugin(AdminLTE)-->
```



```
<!--end::Required Plugin(Bootstrap 5)--><!--begin::Required Plugin(AdminLTE)-->
<script src="{{ asset('adminlte4/js/adminlte.js') }}"></script>
<!--end::Required Plugin(AdminLTE)--><!--begin::OverlayScrollbars Configure-->
<script>
```

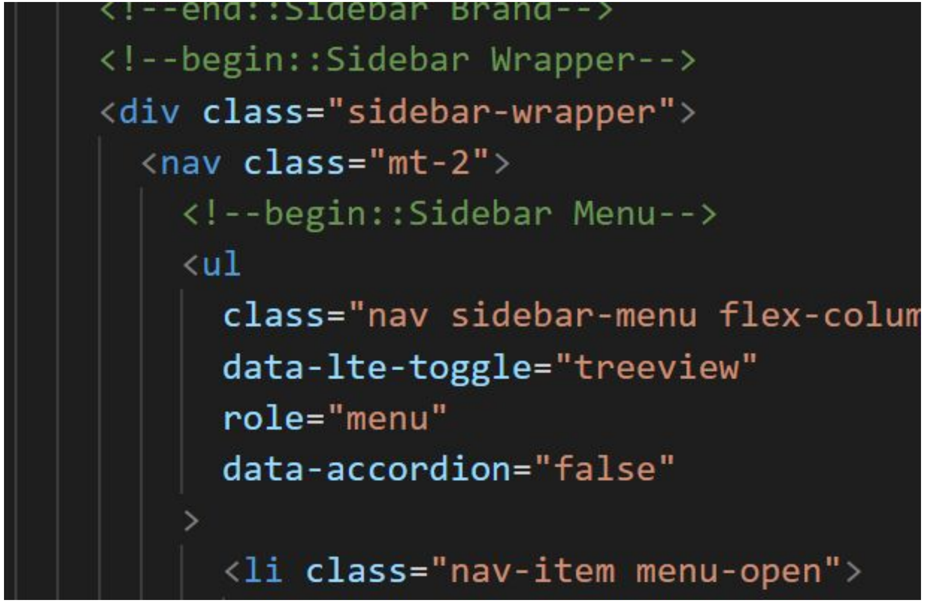
Practice #3: Updating the menu for links to other tables

Workflow Development

1. Go to the script file that implements left–menu
2. Determine what links will appear in the menu
3. Find the URI from the menu in web.php or use the artisan syntax “php artisan route:list”
4. Use the syntax @route('route_name') or @url('path_url') where it goes

1. Go to the script file that implements left-menu

In the above case we go to layouts/adminlte4.blade.php on the <div class="sidebar-wrapper">



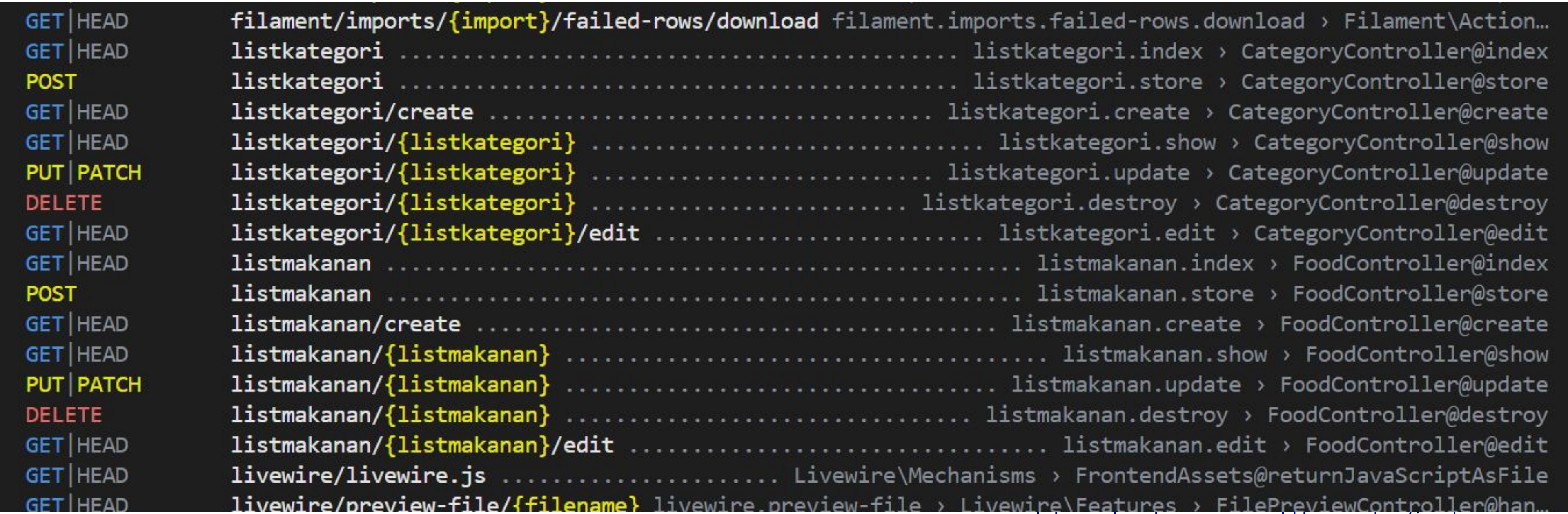
```
<!--end::Sidebar Brand-->
<!--begin::Sidebar Wrapper-->
<div class="sidebar-wrapper">
  <nav class="mt-2">
    <!--begin::Sidebar Menu-->
    <ul
      class="nav sidebar-menu flex-colum
      data-lte-toggle="treeview"
      role="menu"
      data-accordion="false"
    >
      <li class="nav-item menu-open">
```

2. Determine what links will appear in the menu

- Modify the coding in this case by providing 2 menus, namely: Food, Category menus
- In addition to the menu above, we will delete it.

3. Find your URI from route:list

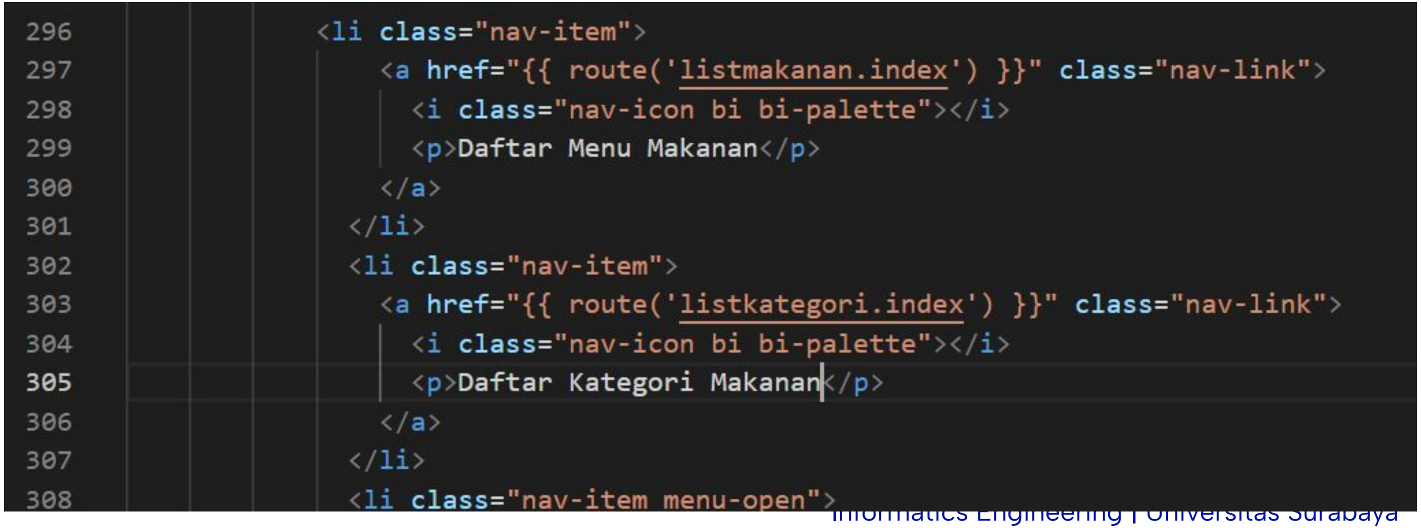
With **php artisan route:list** we are helped to find the link url or route name of the action to be displayed



```
GET|HEAD        filament/imports/{import}/failed-rows/download filament.imports.failed-rows.download › Filament\Action…
GET|HEAD        listkategori ................................... listkategori.index › CategoryController@index
POST            listkategori ................................... listkategori.store › CategoryController@store
GET|HEAD        listkategori/create ............................ listkategori.create › CategoryController@create
GET|HEAD        listkategori/{listkategori} ........................ listkategori.show › CategoryController@show
PUT|PATCH       listkategori/{listkategori} ...................... listkategori.update › CategoryController@update
DELETE          listkategori/{listkategori} ..................... listkategori.destroy › CategoryController@destroy
GET|HEAD        listkategori/{listkategori}/edit ................... listkategori.edit › CategoryController@edit
GET|HEAD        listmakanan ........................................ listmakanan.index › FoodController@index
POST            listmakanan ........................................ listmakanan.store › FoodController@store
GET|HEAD        listmakanan/create ................................ listmakanan.create › FoodController@create
GET|HEAD        listmakanan/{listmakanan} ............................ listmakanan.show › FoodController@show
PUT|PATCH       listmakanan/{listmakanan} ........................ listmakanan.update › FoodController@update
DELETE          listmakanan/{listmakanan} ....................... listmakanan.destroy › FoodController@destroy
GET|HEAD        listmakanan/{listmakanan}/edit ....................... listmakanan.edit › FoodController@edit
GET|HEAD        livewire/livewire.js ................ Livewire\Mechanisms › FrontendAssets@returnJavaScriptAsFile
GET|HEAD        livewire/preview-file/{filename} livewire.preview-file › Livewire\Features › FilePreviewController@han…
```

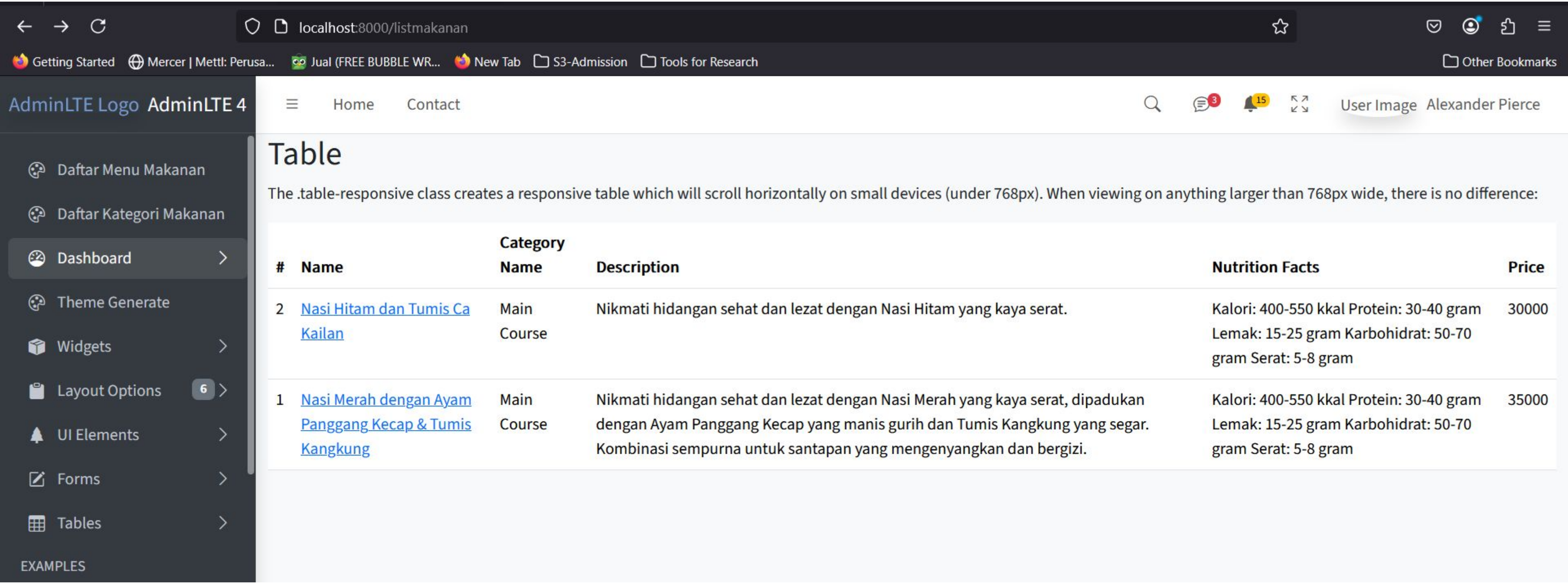
Add script -> route()

Remember @route and @url is syntax blade then must given {{ … }}



```
<li class="nav-item">
    <a href="{{ route('listmakanan.index') }}" class="nav-link">
      <i class="nav-icon bi bi-palette"></i>
      <p>Daftar Menu Makanan</p>
    </a>
  </li>
  <li class="nav-item">
    <a href="{{ route('listkategori.index') }}" class="nav-link">
      <i class="nav-icon bi bi-palette"></i>
      <p>Daftar Kategori Makanan</p>
    </a>
  </li>
  <li class="nav-item menu-open">
```

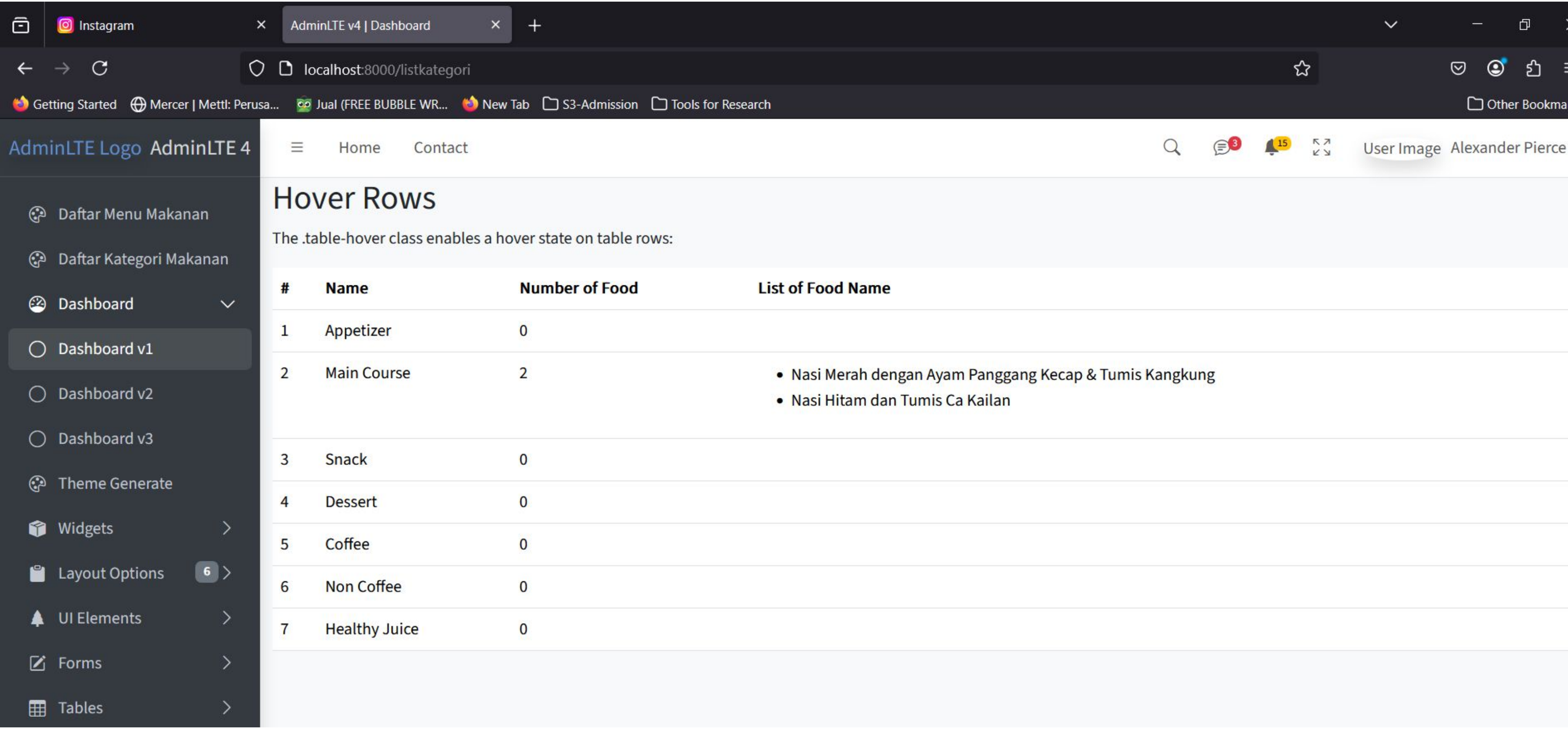
Result



Table

The .table-responsive class creates a responsive table which will scroll horizontally on small devices (under 768px). When viewing on anything larger than 768px wide, there is no difference:

#	Name	Category Name	Description	Nutrition Facts	Price
2	Nasi Hitam dan Tumis Ca Kailan	Main Course	Nikmati hidangan sehat dan lezat dengan Nasi Hitam yang kaya serat.	Kalori: 400-550 kkal Protein: 30-40 gram Lemak: 15-25 gram Karbohidrat: 50-70 gram Serat: 5-8 gram	30000
1	Nasi Merah dengan Ayam Panggang Kecap & Tumis Kangkung	Main Course	Nikmati hidangan sehat dan lezat dengan Nasi Merah yang kaya serat, dipadukan dengan Ayam Panggang Kecap yang manis gurih dan Tumis Kangkung yang segar. Kombinasi sempurna untuk santapan yang mengenyangkan dan bergizi.	Kalori: 400-550 kkal Protein: 30-40 gram Lemak: 15-25 gram Karbohidrat: 50-70 gram Serat: 5-8 gram	35000



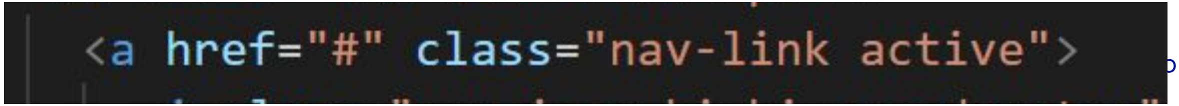
Hover Rows

The .table-hover class enables a hover state on table rows:

#	Name	Number of Food	List of Food Name
1	Appetizer	0	
2	Main Course	2	• Nasi Merah dengan Ayam Panggang Kecap & Tumis Kangkung • Nasi Hitam dan Tumis Ca Kailan
3	Snack	0	
4	Dessert	0	
5	Coffee	0	
6	Non Coffee	0	
7	Healthy Juice	0	

Your Task

- #1. Modify all page of views and reports that you have created in previous tasks with new templates.
    - Food/Category/Customer/Order
    - All your Report
- #2. Add menus in the sidebar to access those pages.
- #3. Change each of title homepage with your current page (clue: add @yield(‘title’) at your layouts)
- #4. Change active menu from your side-bar menu. (clue: add @yield(‘side-bar’) at your layouts and add “active” in side class)



```
<a href="#" class="nav-link active">
```

Thank You