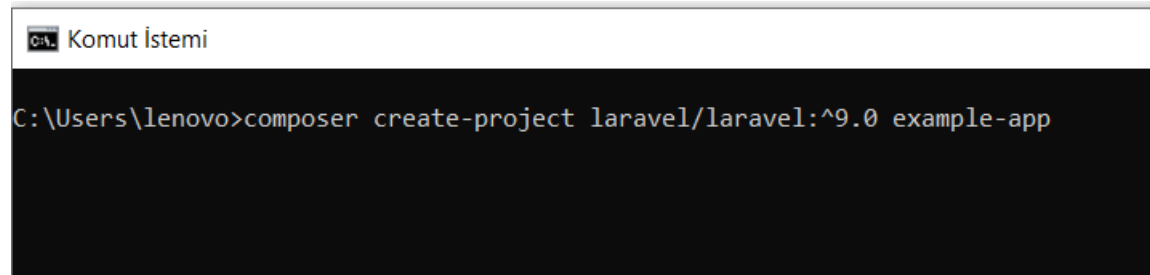


# Laravel Route

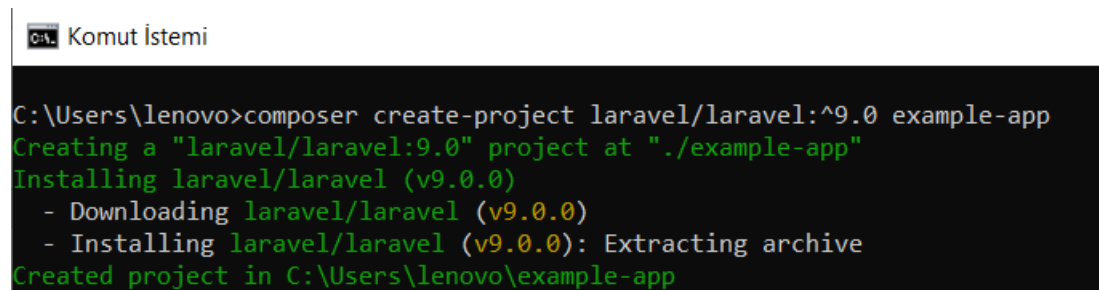
# Laravel 9 Kurulumu ve Test Edilmesi

- cmd açtıktan sonra,
  - `composer create-project laravel/laravel:^9.0 example-app`  
Komutu çalıştırılarak example-app adında yeni bir laravel projesi oluşturulur.



```
C:\Users\lenovo>composer create-project laravel/laravel:^9.0 example-app
```

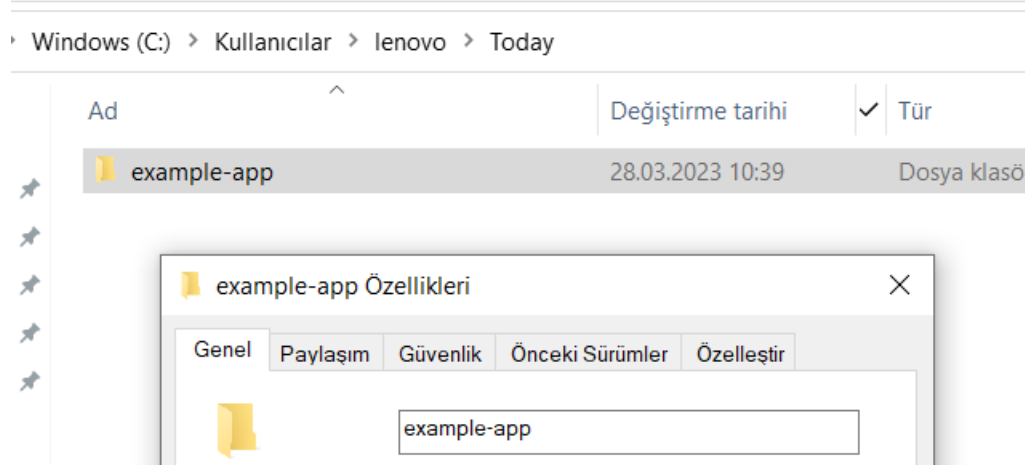
- kurulum başladığında laravel versiyonu ve nereye kurulduğu bilgisi veriliyor.



```
C:\Users\lenovo>composer create-project laravel/laravel:^9.0 example-app
Creating a "laravel/laravel:9.0" project at "./example-app"
Installing laravel/laravel (v9.0.0)
- Downloading laravel/laravel (v9.0.0)
- Installing laravel/laravel (v9.0.0): Extracting archive
Created project in C:\Users\lenovo\example-app
```

# Laravel 9 Kurulumu ve Test Edilmesi

- Kurulduktan sonra C:\Users\lenovo\example-app dizinine gidip klasörü görebiliriz.
- example-app klasöründe laravel dosyaları yüklenmiştir.



Windows (C:) > Kullanıcılar > lenovo > example-app

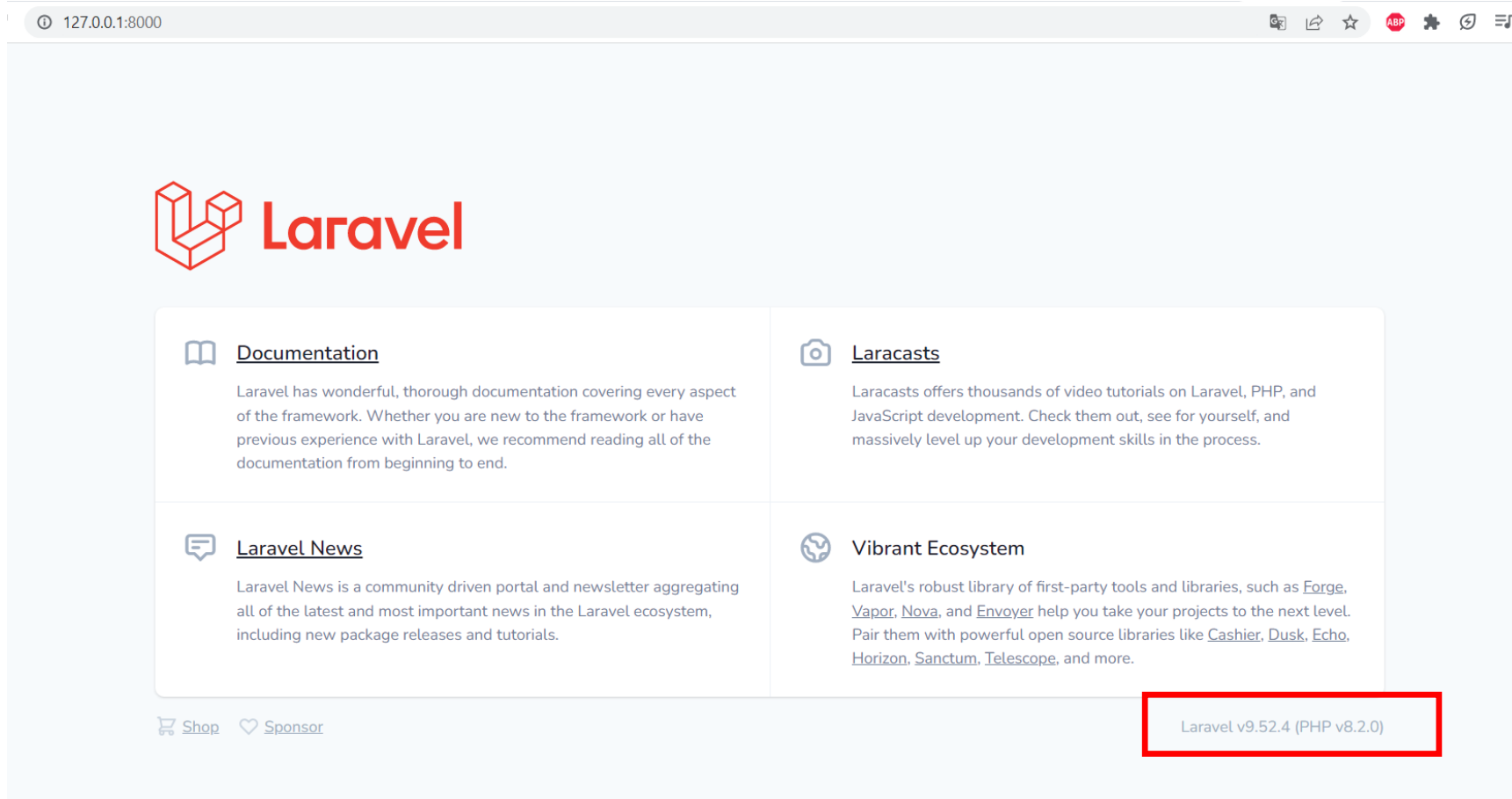
Ad	Değiştirme tarihi	Tür	Boyut
app	28.03.2023 10:39	Dosya klasörü	
bootstrap	28.03.2023 10:39	Dosya klasörü	
config	28.03.2023 10:39	Dosya klasörü	
database	28.03.2023 10:39	Dosya klasörü	
lang	28.03.2023 10:39	Dosya klasörü	
public	28.03.2023 10:39	Dosya klasörü	
resources	28.03.2023 10:39	Dosya klasörü	
routes	28.03.2023 10:39	Dosya klasörü	
storage	28.03.2023 10:39	Dosya klasörü	
tests	28.03.2023 10:39	Dosya klasörü	
vendor	28.03.2023 10:41	Dosya klasörü	
.editorconfig	8.02.2022 07:52	EDITORCONFIG D...	1 KB
.env	28.03.2023 10:41	ENV Dosyası	1 KB
.env.example	8.02.2022 07:52	EXAMPLE Dosyası	1 KB
.gitattributes	8.02.2022 07:52	Metin Belgesi	1 KB
.gitignore	8.02.2022 07:52	Metin Belgesi	1 KB
.styleci.yml	8.02.2022 07:52	YML Dosyası	1 KB
artisan	8.02.2022 07:52	Dosya	2 KB
composer.json	8.02.2022 07:52	JSON Dosyası	2 KB
composer.lock	28.03.2023 10:39	LOCK Dosyası	287 KB
package.json	8.02.2022 07:52	JSON Dosyası	1 KB
phpunit.xml	8.02.2022 07:52	XML Belgesi	2 KB
README.md	8.02.2022 07:52	MD Dosyası	4 KB
webpack.mix.js	8.02.2022 07:52	JavaScript Dosyası	1 KB

# Laravel 9 Kurulumu ve Test Edilmesi

- `cd example-app`  
komutu ile example-app klasörü açılır.
- `php artisan serve`  
komutu ile localhost adresine erişilir.


```
C:\Users\lenovo>cd example-app  
C:\Users\lenovo\example-app>php artisan serve  
INFO Server running on [http://127.0.0.1:8000].  
Press Ctrl+C to stop the server
```


Localhost adresinden Laravel ve PHP versiyonlarını görebiliriz.




The screenshot shows the Laravel website interface. At the top, the browser address bar displays '127.0.0.1:8000'. The main content area features the Laravel logo and four links: Documentation, Laracasts, Laravel News, and Vibrant Ecosystem. At the bottom left, there are links for 'Shop' and 'Sponsor'. At the bottom right, a red box highlights the text 'Laravel v9.52.4 (PHP v8.2.0)'.

127.0.0.1:8000


 **Laravel**

 [Documentation](#)


Laravel has wonderful, thorough documentation covering every aspect of the framework. Whether you are new to the framework or have previous experience with Laravel, we recommend reading all of the documentation from beginning to end.

 [Laracasts](#)



Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.

 [Laravel News](#)

Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.

 [Vibrant Ecosystem](#)

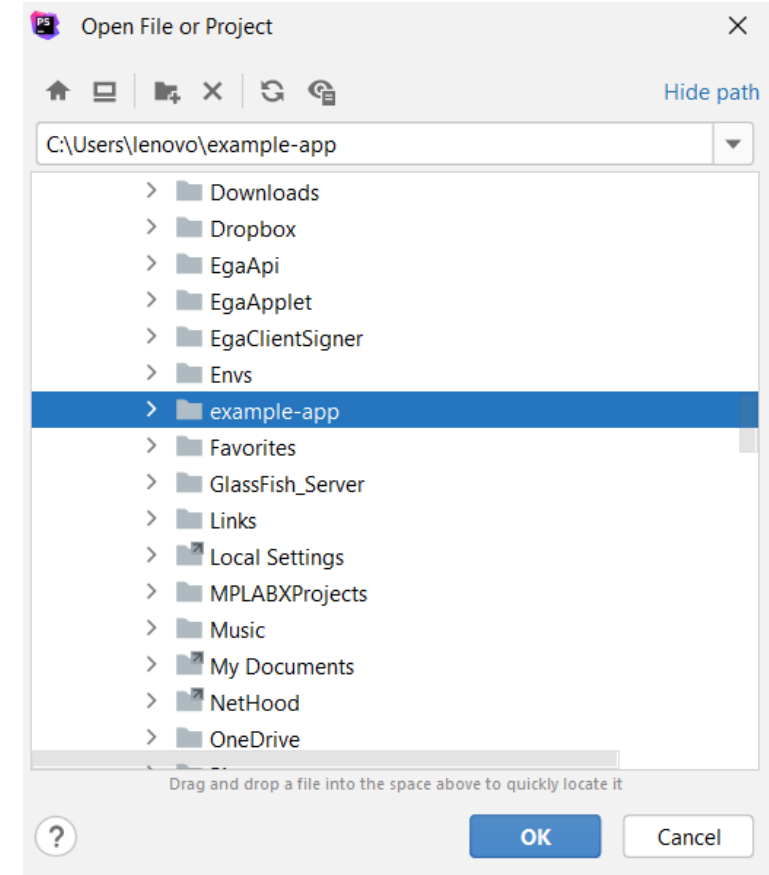
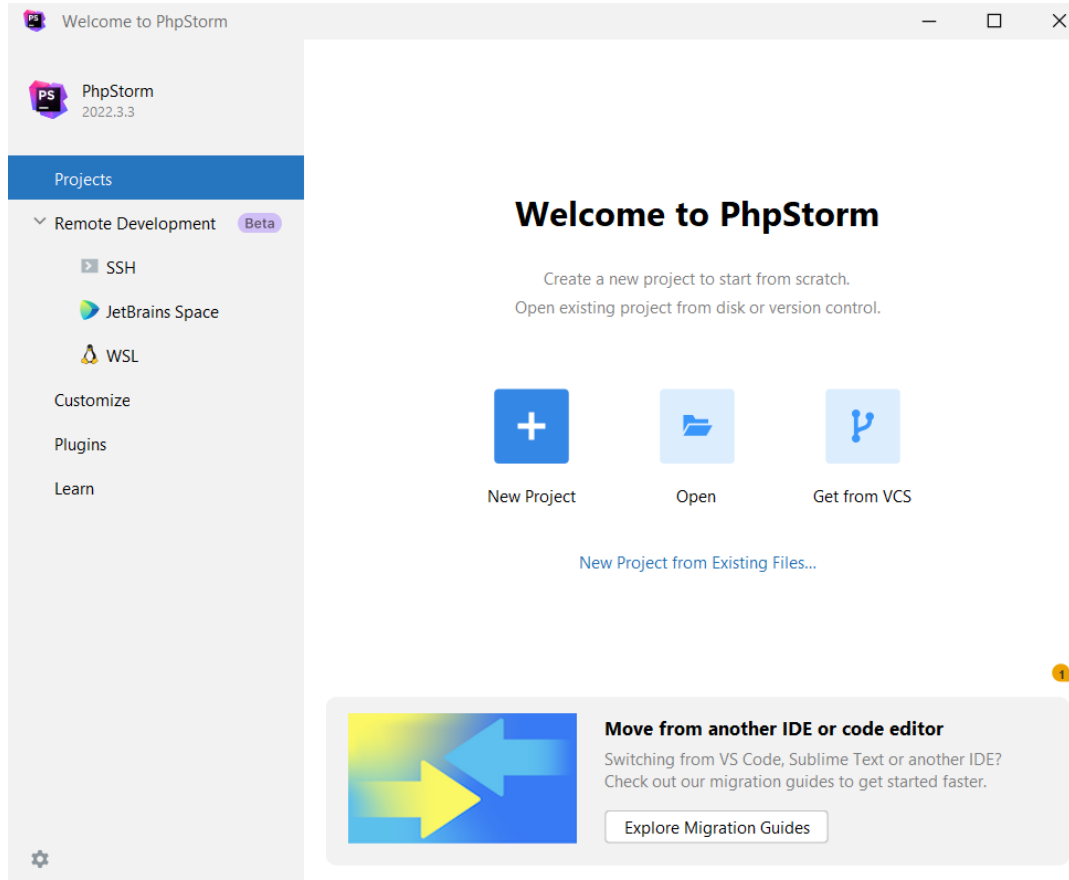
Laravel's robust library of first-party tools and libraries, such as [Forge](#), [Vapor](#), [Nova](#), and [Envoyer](#) help you take your projects to the next level. Pair them with powerful open source libraries like [Cashier](#), [Dusk](#), [Echo](#), [Horizon](#), [Sanctum](#), [Telescope](#), and more.

 [Shop](#)  [Sponsor](#)

Laravel v9.52.4 (PHP v8.2.0)

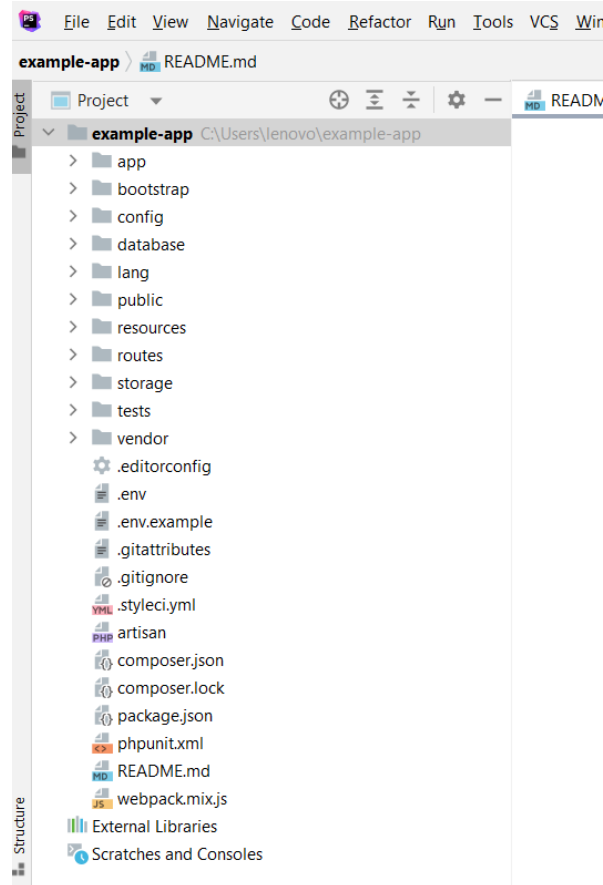
# Laravel 9 Kurulumu ve Test Edilmesi

- PHPStorm'dan projeyi açmak için open'a basıp proje dosyasının olduğu dizine gidiyoruz.



# Laravel 9 Kurulumu ve Test Edilmesi

- Laravel dosyaları PhpStorm'da açıldı.

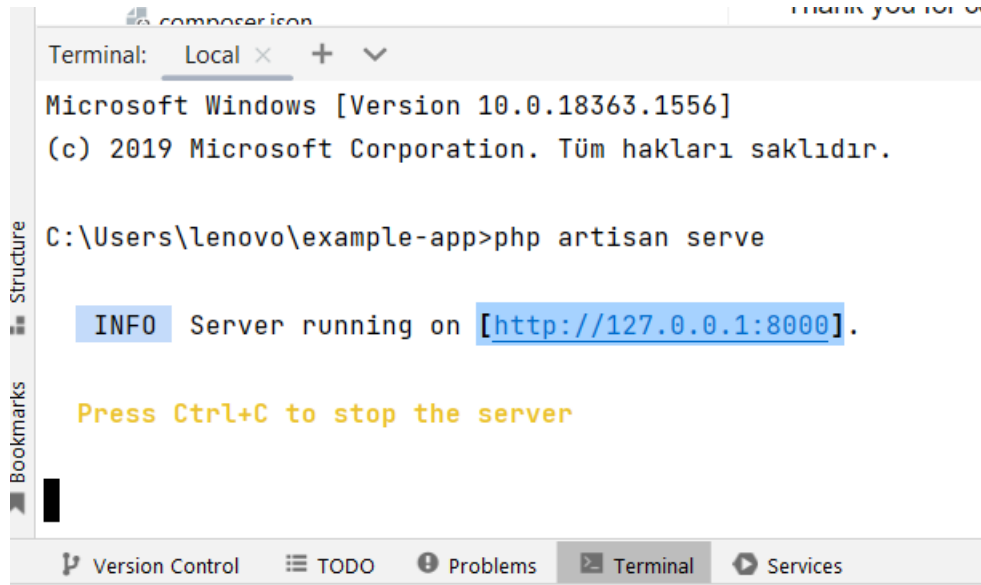


# Laravel 9 Kurulumu ve Test Edilmesi

- Burda terminali açarak,
  - php artisan serve

Komutunu yazıyoruz. php artisan laravelin bir paketidir. Bu komut projemizi localhost'ta çalıştırır.

Komut çalıştıktan sonra localhosta verilen ip adresinde çalışır.

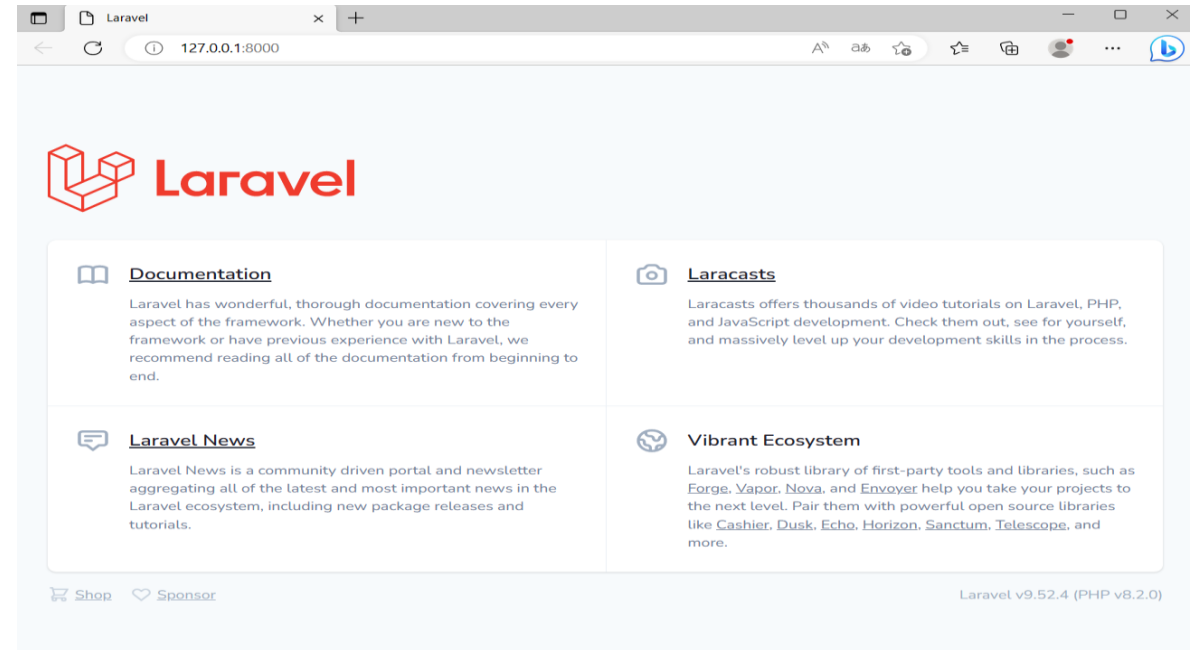


```
Terminal: Local x + v
Microsoft Windows [Version 10.0.18363.1556]
(c) 2019 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\lenovo\example-app>php artisan serve

INFO Server running on [http://127.0.0.1:8000].

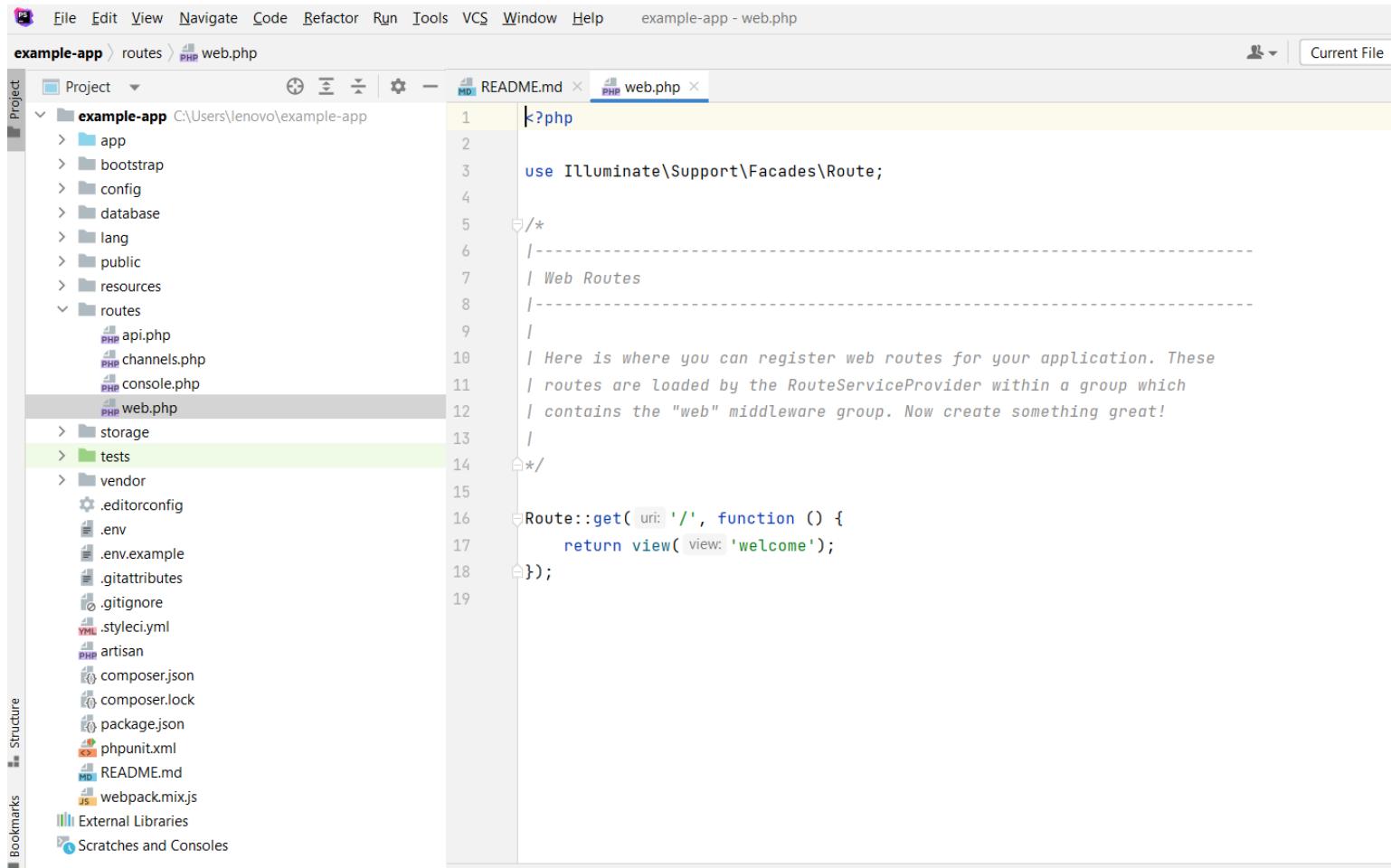
Press Ctrl+C to stop the server
```





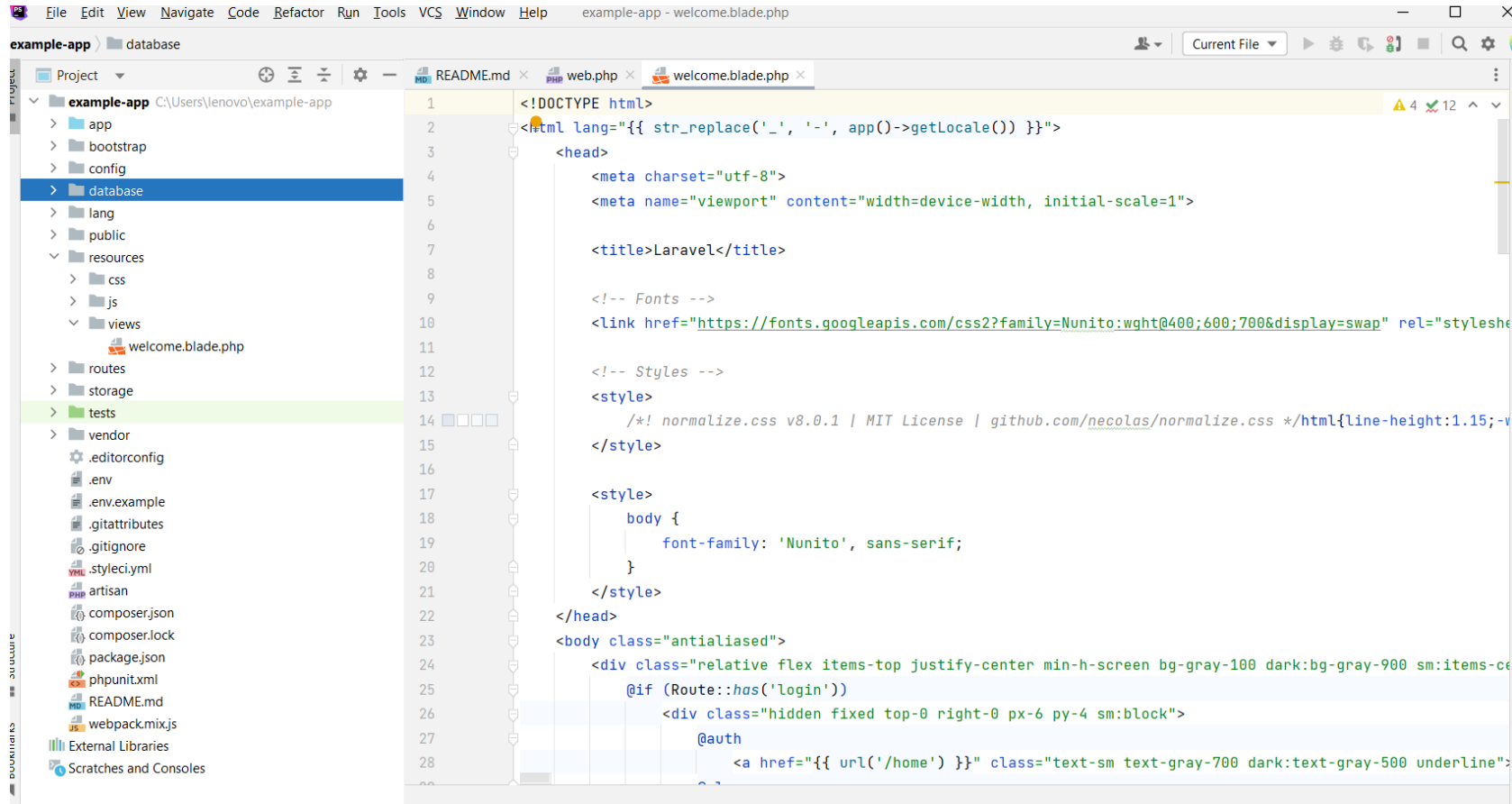
# Standart Route Oluşturma ve Kullanımı

- PhpStrom'da routes klasöründeki web.php dosyası açılır.



# Standart Route Oluřturma ve Kullanımı

- Route bařlangıçta view dizini altındaki welcome.blade.php dosyasını alıřtırıyor.



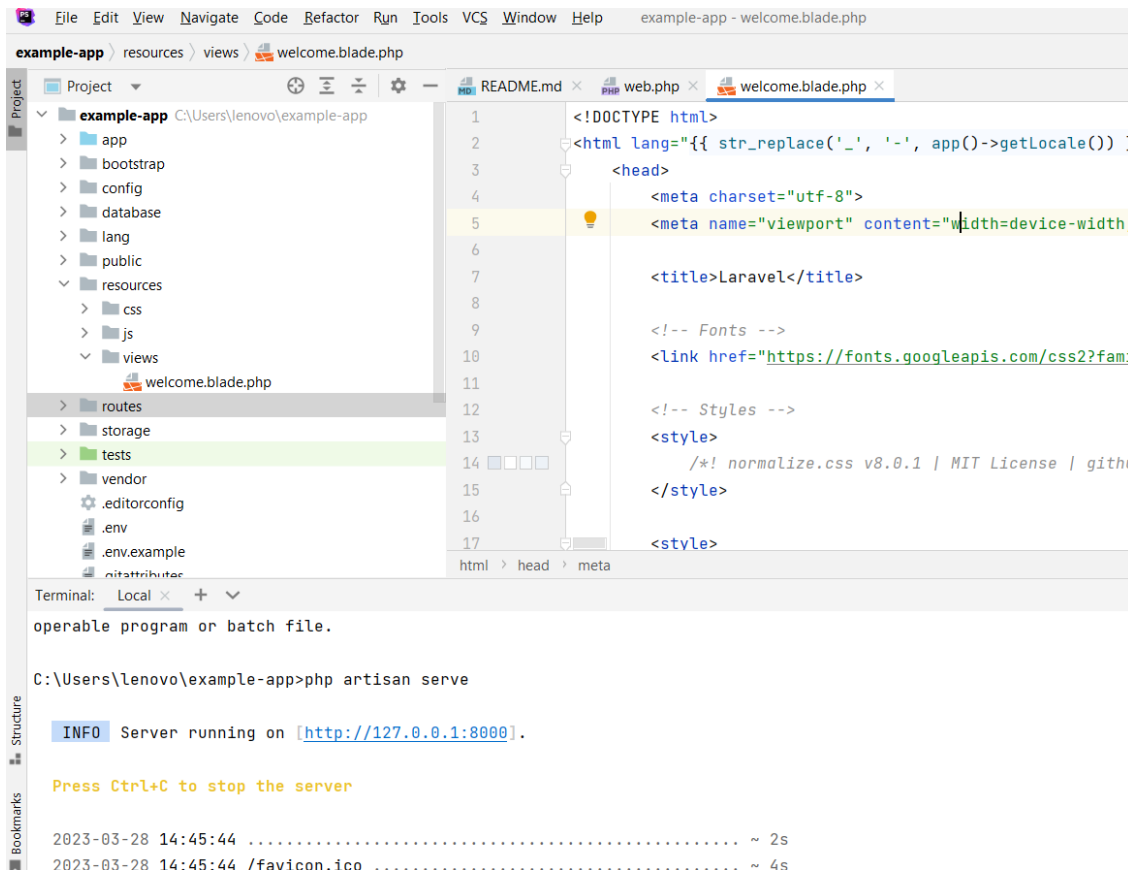
The screenshot shows an IDE window with the following components:

- Project Explorer (Left):** Displays the project structure. The 'views' directory is expanded, showing 'welcome.blade.php'.
- Code Editor (Right):** Displays the content of 'welcome.blade.php'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
3
4 <head>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7
8     <title>Laravel</title>
9
10    <!-- Fonts -->
11    <link href="https://fonts.googleapis.com/css2?family=Nunito:wght@400;600;700&display=swap" rel="stylesheet">
12
13    <!-- Styles -->
14    <style>
15        /* normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */html{line-height:1.15;-v
16    </style>
17
18    <style>
19        body {
20            font-family: 'Nunito', sans-serif;
21        }
22    </style>
23
24    </head>
25    <body class="antialiased">
26        <div class="relative flex items-top justify-center min-h-screen bg-gray-100 dark:bg-gray-900 sm:items-ce
27            @if (Route::has('login'))
28                <div class="hidden fixed top-0 right-0 px-6 py-4 sm:block">
29                    @auth
30                        <a href="{{ url('/home') }}" class="text-sm text-gray-700 dark:text-gray-500 underline">
```

# Standart Route Oluşturma ve Kullanımı

- Terminalde php artisan serve yazdığımızda localhost adresini bize geri döndürecektir.



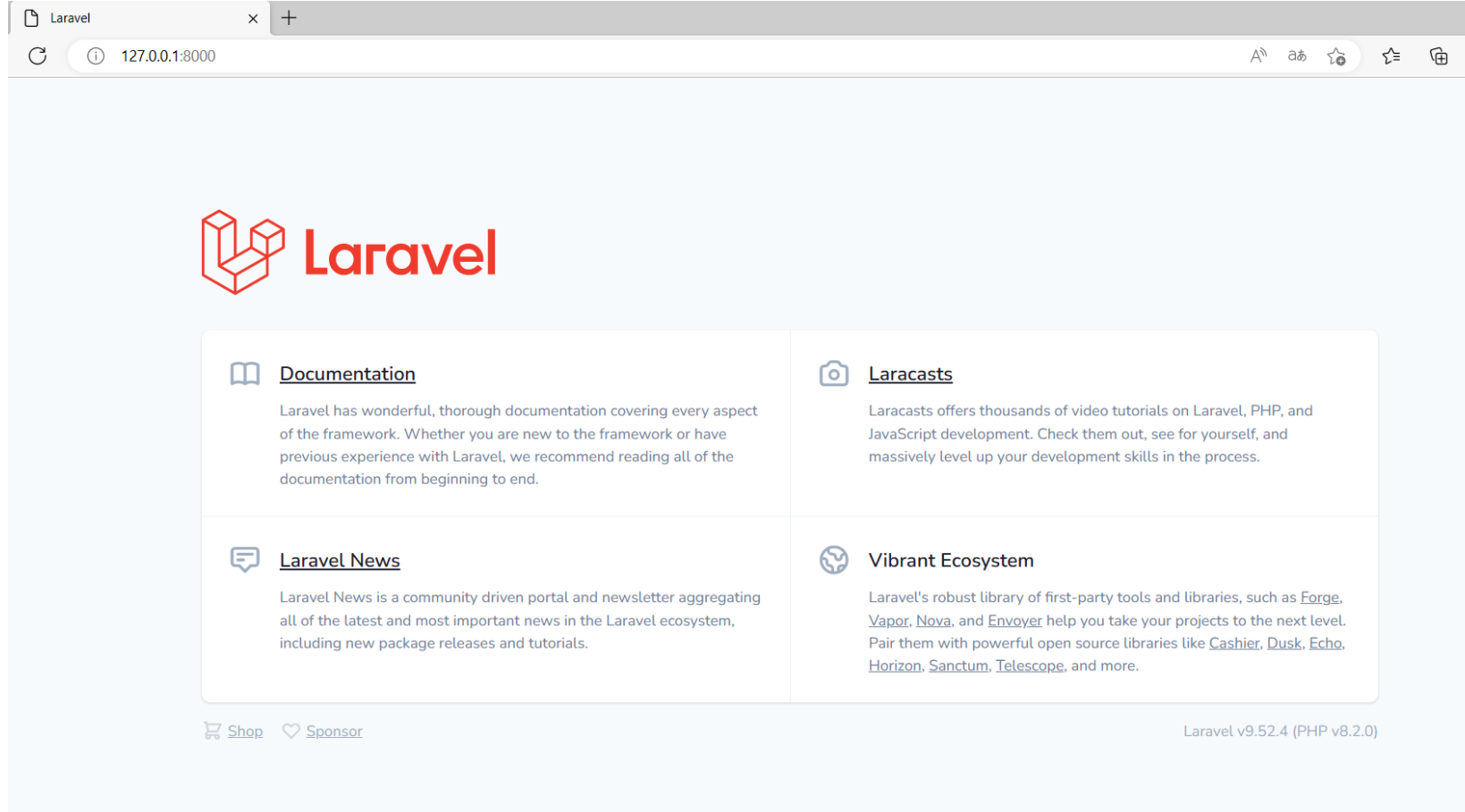
The screenshot shows an IDE window with the following components:

- Project Explorer:** Displays the project structure of 'example-app'. The 'views' directory is expanded, showing 'welcome.blade.php'.
- Code Editor:** Displays the content of 'welcome.blade.php'. The code is as follows:

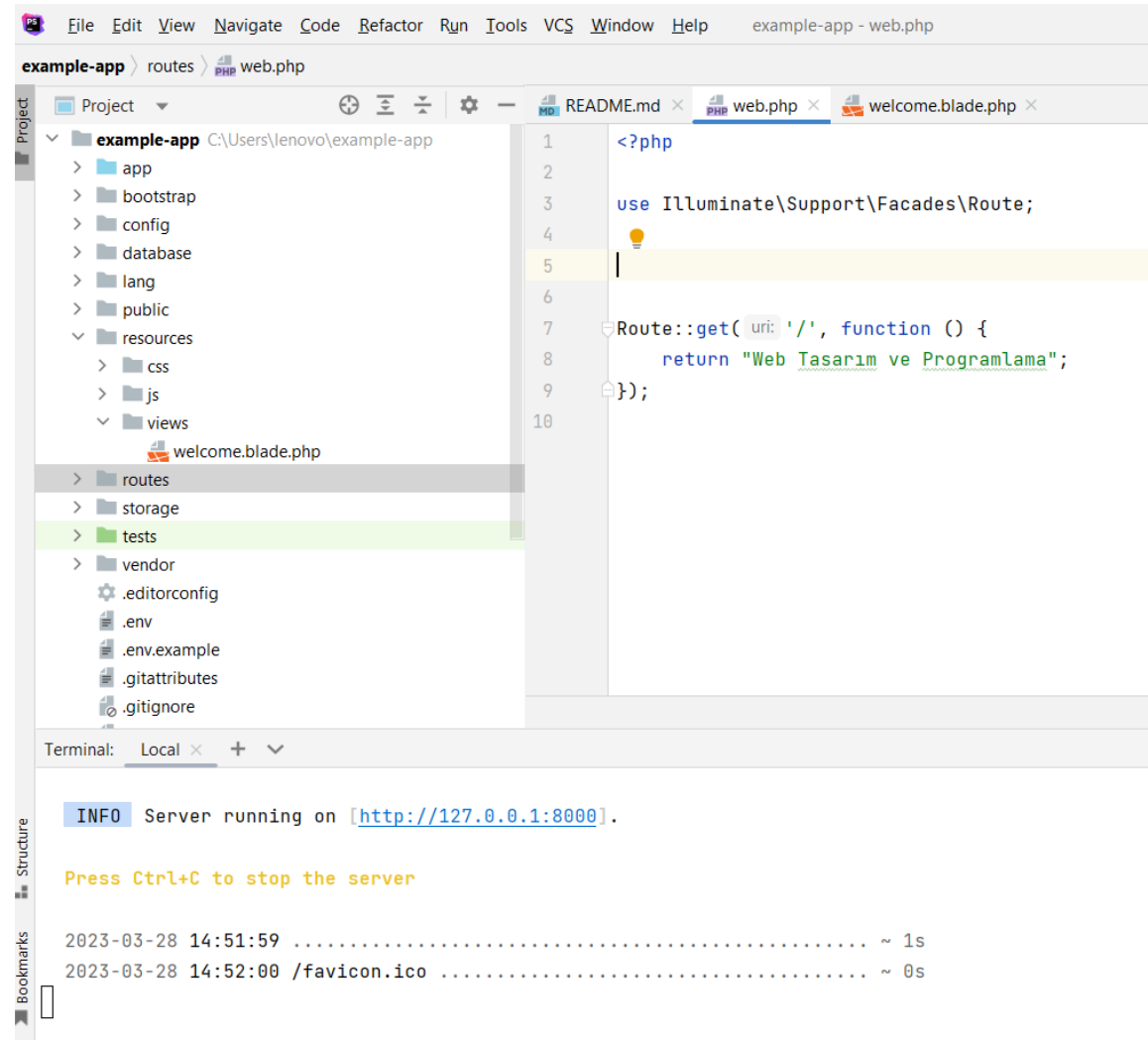
```
1 <!DOCTYPE html>
2 <html lang="{ str_replace('_', '-', app()->getLocale()) }">
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width,
6   </head>
7   <title>Laravel</title>
8
9   <!-- Fonts -->
10  <link href="https://fonts.googleapis.com/css2?family=
11
12  <!-- Styles -->
13  <style>
14    /*! normalize.css v8.0.1 | MIT License | githu
15  </style>
16
17  <style>
```
- Terminal:** Shows the output of the 'php artisan serve' command. The output is:

```
C:\Users\lenovo\example-app>php artisan serve
INFO Server running on [http://127.0.0.1:8000].
Press Ctrl+C to stop the server
2023-03-28 14:45:44 ..... ~ 2s
2023-03-28 14:45:44 /favicon.ico ..... ~ 4s
```

Bu sayfa welcome.blade.php dosyasıdır.



web.php dosyasındaki Route komutunu aşağıdaki gibi değiştirip, Ctrl+C ile server durdurulup php artisan serve komutunu tekrar çalıştırdığımızda



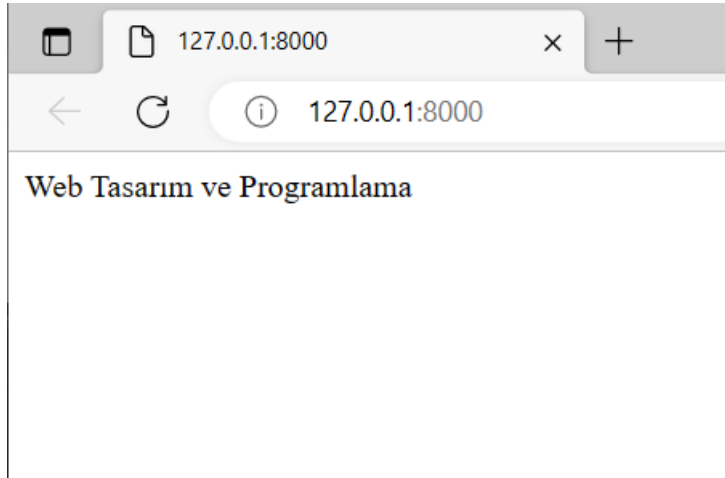
The screenshot shows an IDE interface with the following components:

- Project Explorer:** Displays the project structure of 'example-app'. The 'routes' folder is selected.
- Code Editor:** Shows the 'web.php' file with the following code:

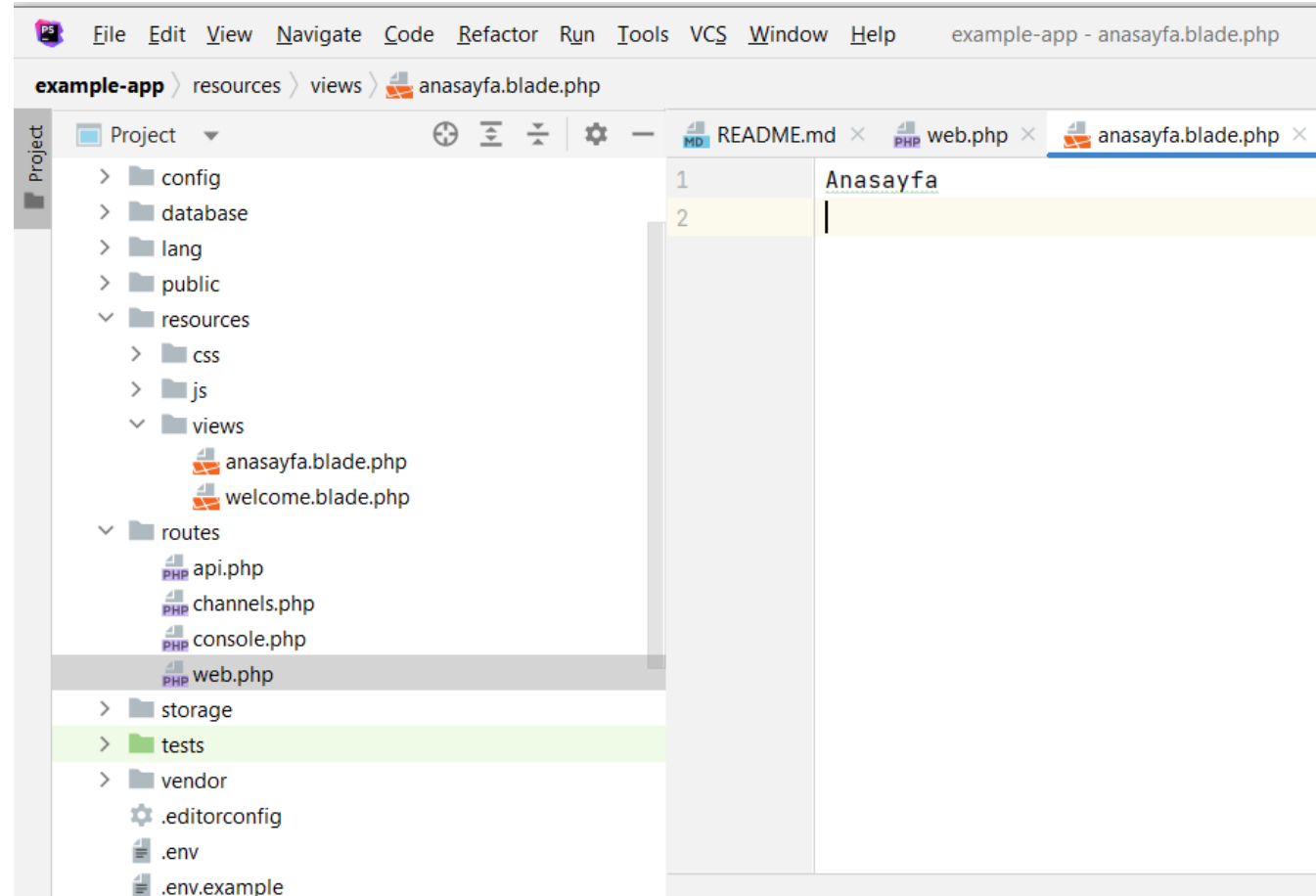
```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5
6
7 Route::get( uri: '/', function () {
8     return "Web Tasarım ve Programlama";
9 });
10
```
- Terminal:** Shows the output of the server running on `http://127.0.0.1:8000`. It includes a message to press Ctrl+C to stop the server and a list of recent requests:

```
INFO Server running on [http://127.0.0.1:8000].
Press Ctrl+C to stop the server
2023-03-28 14:51:59 ..... ~ 1s
2023-03-28 14:52:00 /favicon.ico ..... ~ 0s
```

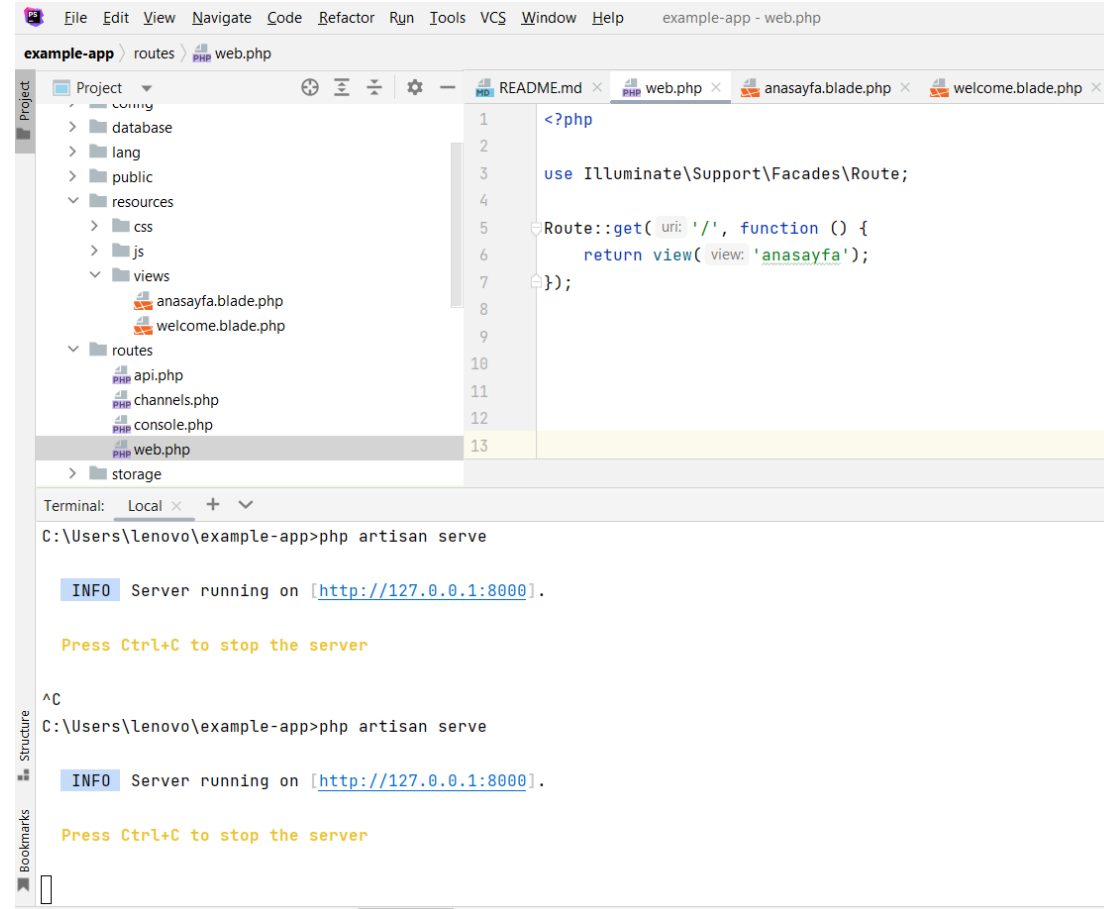
Giriş sayfasında Web Tasarım ve Programlama yazısı yer alacaktır.



Resources klasörünün altındaki views klasörüne anasayfa.blade.php dosyası oluşturulup içine Anasayfa yazısı eklendi.



web.php dosyasındaki Route komutunu aşağıdaki gibi değiştirip, Ctrl+C ile server durdurulup php artisan serve komutunu tekrar çalıştırdığımızda



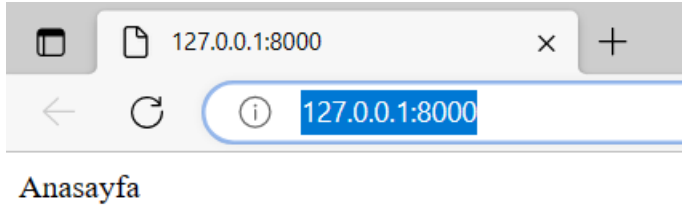
The screenshot shows an IDE window titled 'example-app - web.php'. The left sidebar displays the project structure with the 'routes' folder selected. The main editor shows the content of 'web.php' with the following code:

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::get( uri: '/', function () {
6     return view( view: 'anasayfa' );
7 });
8
9
10
11
12
13
```

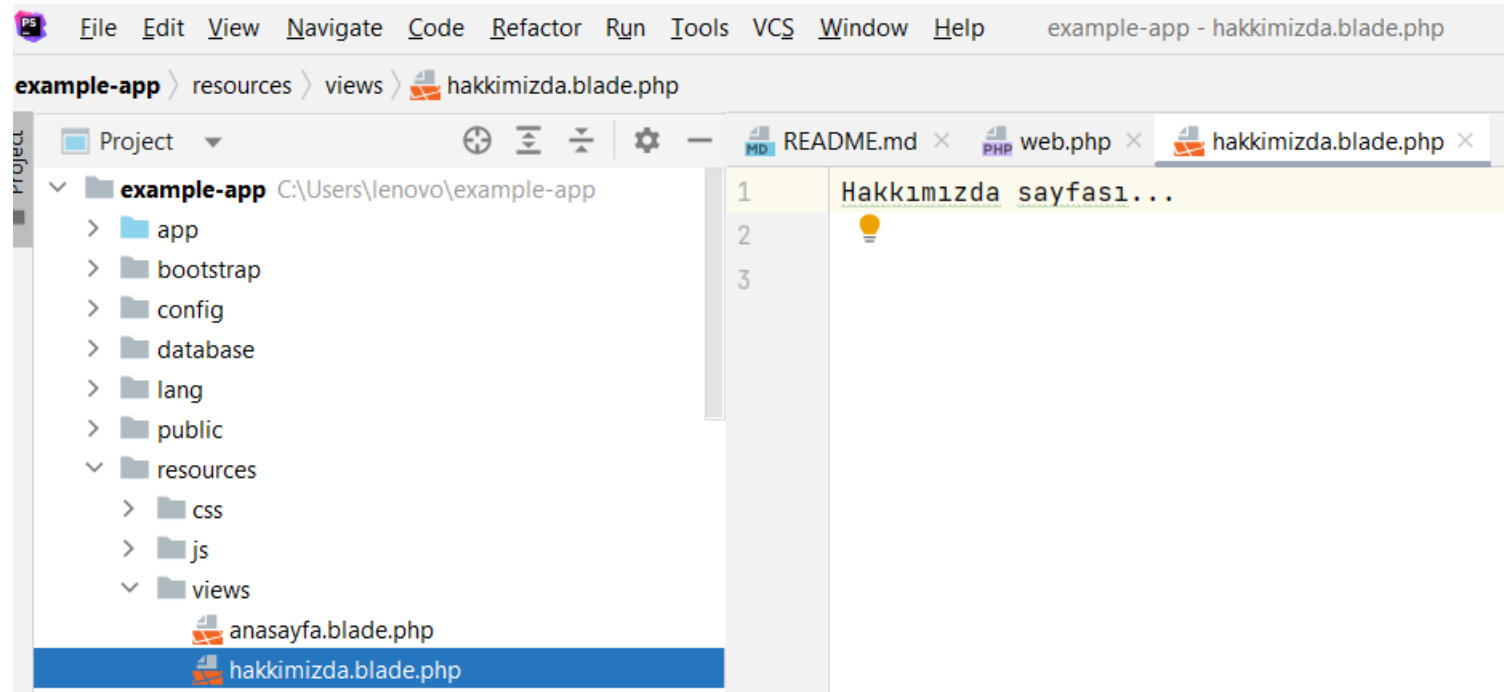
The bottom terminal window shows the command 'C:\Users\lenovo\example-app>php artisan serve' being executed. The output indicates the server is running on 'http://127.0.0.1:8000'. The user then presses Ctrl+C, which is shown as '^C' in the terminal, and the command is re-executed, resulting in the same output.



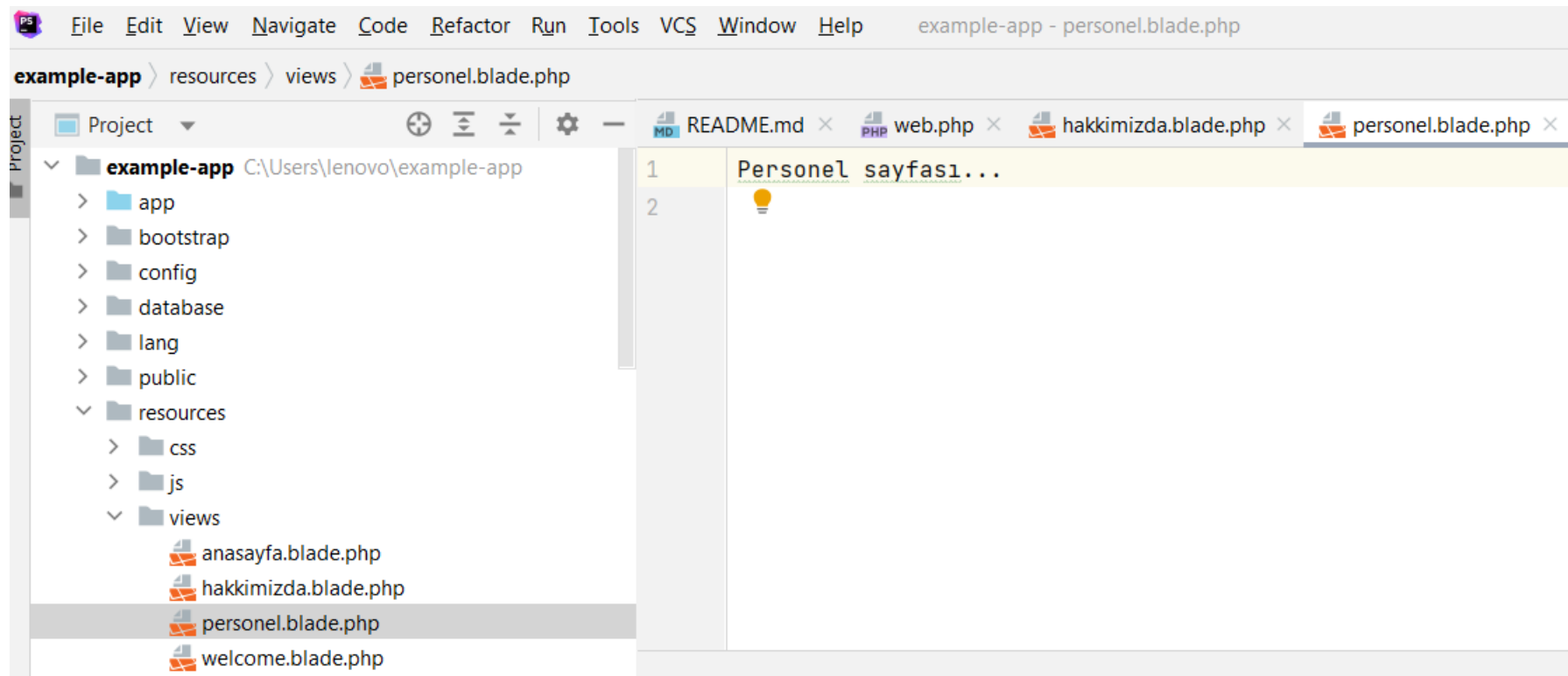
Giriş sayfasında Anasayfa yazısı yer alacaktır.



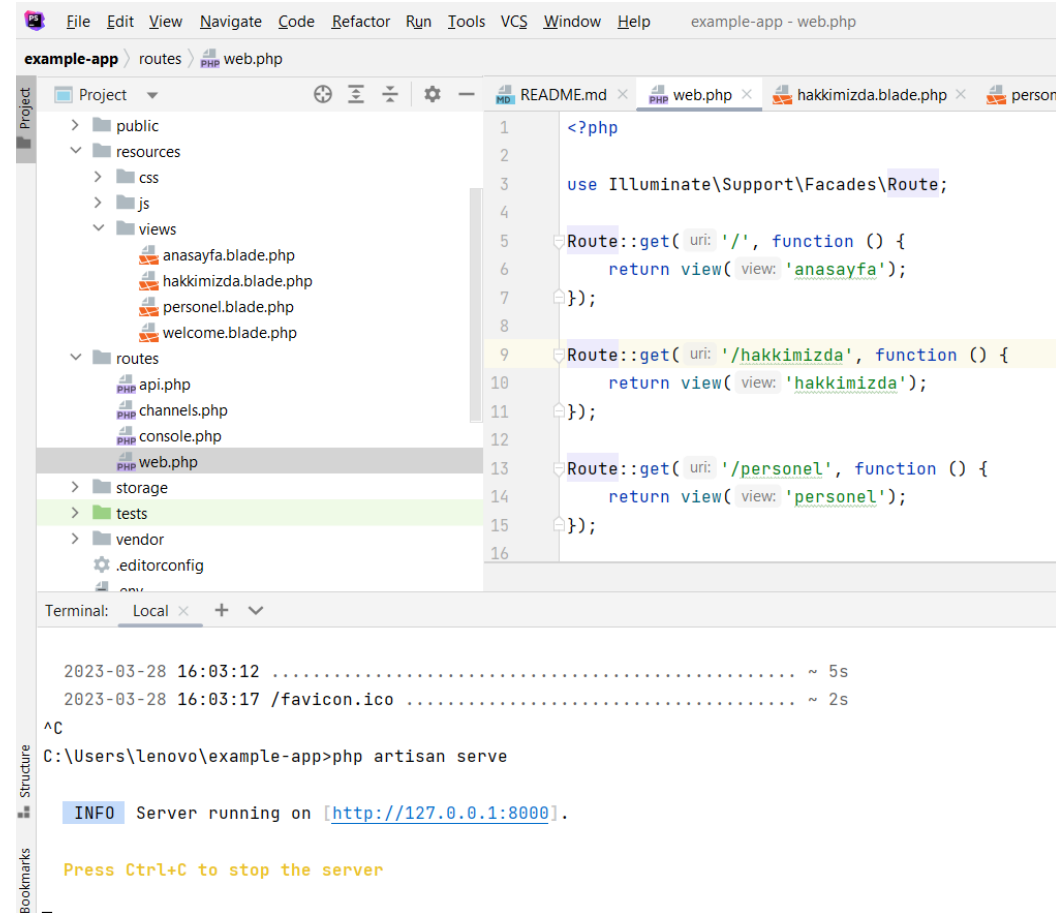
Resources klasörünün altındaki views klasörüne hakkimizda.blade.php dosyası oluşturulup içine «Hakkımızda sayfası...» yazısı eklendi.



Resourcers klasörünün altındaki views klasörüne personel.blade.php dosyası oluşturulup içine «Personel sayfası...» yazısı eklendi.



web.php dosyasındaki Route komutunu aşağıdaki kodlar eklenip, Ctrl+C ile server durdurulup php artisan serve komutunu tekrar çalıştırdığımızda



The screenshot shows an IDE window titled 'example-app - web.php'. The left sidebar displays the project structure with the 'routes' folder expanded, showing 'web.php' selected. The main editor area displays the contents of 'web.php' with line numbers 1 through 16. The code defines three GET routes: a root route pointing to 'anasayfa', a route for '/hakkimizda' pointing to 'hakkimizda', and a route for '/personel' pointing to 'personel'. The bottom terminal window shows the command 'C:\Users\lenovo\example-app>php artisan serve' and its output: 'INFO Server running on [http://127.0.0.1:8000]. Press Ctrl+C to stop the server'.

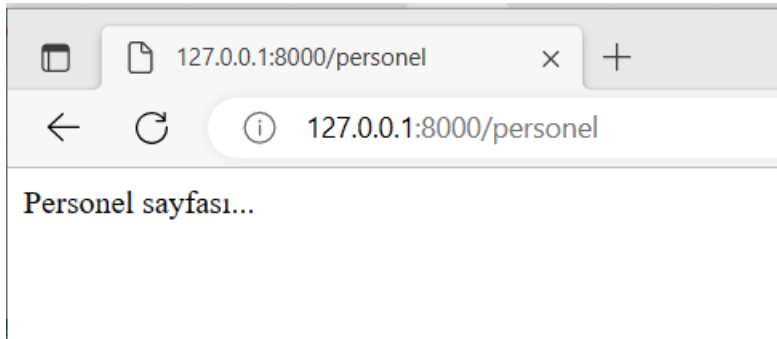
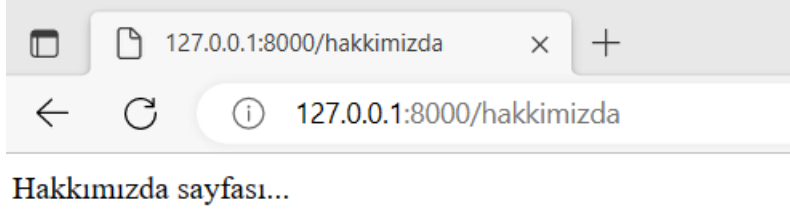
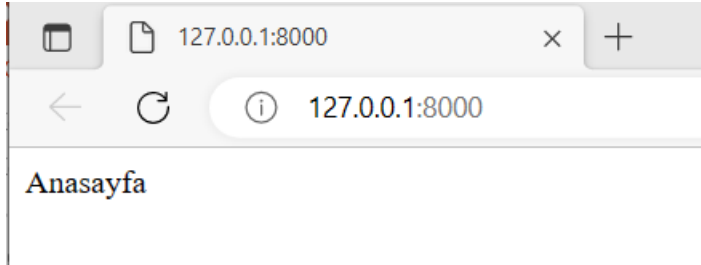
```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::get( uri: '/', function () {
6     return view( view: 'anasayfa' );
7 });
8
9 Route::get( uri: '/hakkimizda', function () {
10     return view( view: 'hakkimizda' );
11 });
12
13 Route::get( uri: '/personel', function () {
14     return view( view: 'personel' );
15 });
16
```

Terminal: Local x + v

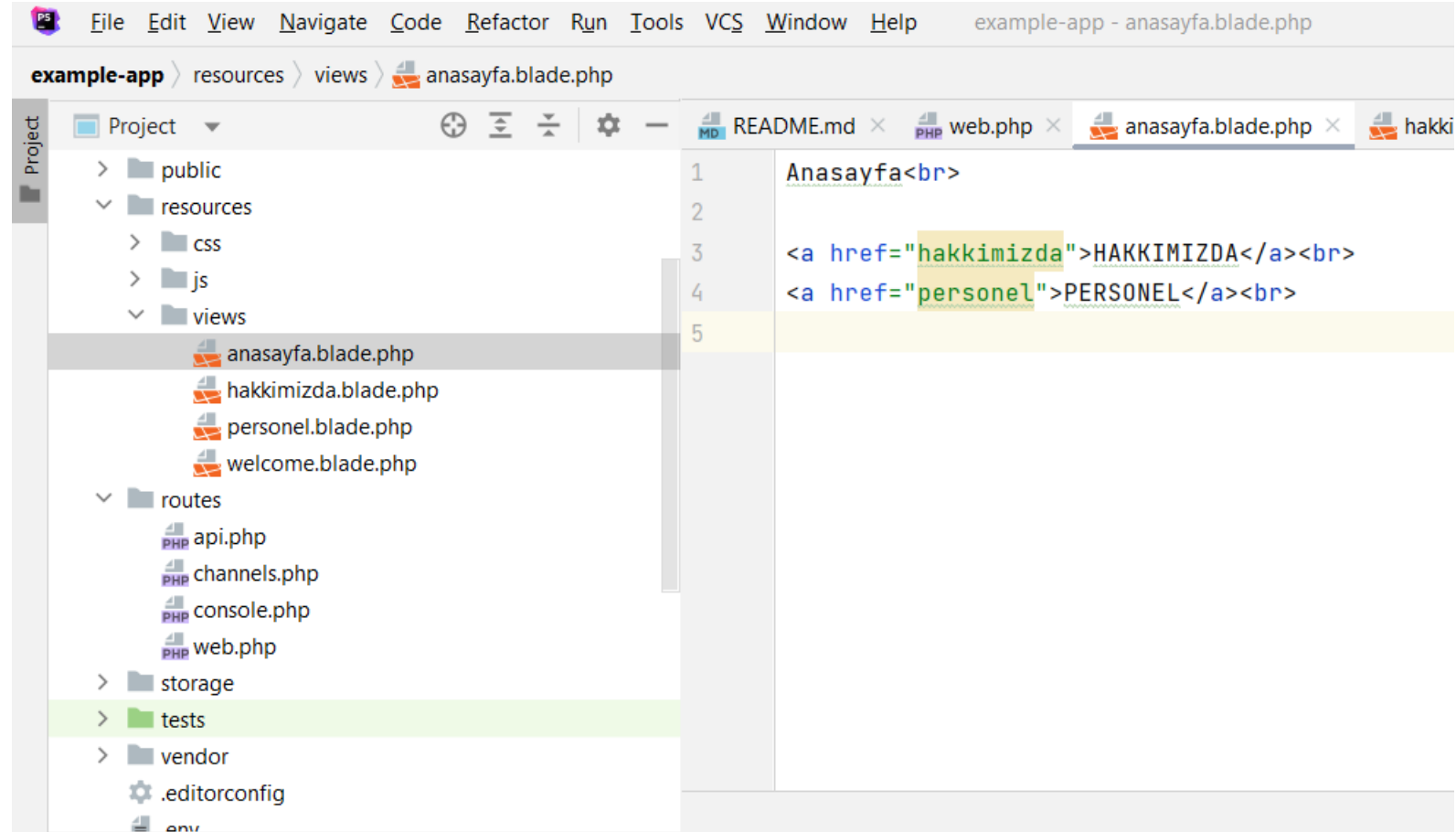
```
2023-03-28 16:03:12 ..... ~ 5s
2023-03-28 16:03:17 /favicon.ico ..... ~ 2s
^C
C:\Users\lenovo\example-app>php artisan serve

INFO Server running on [http://127.0.0.1:8000].

Press Ctrl+C to stop the server
```



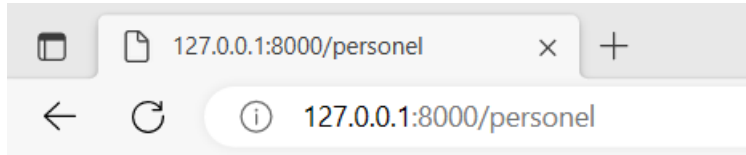
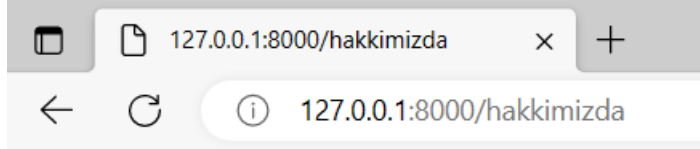
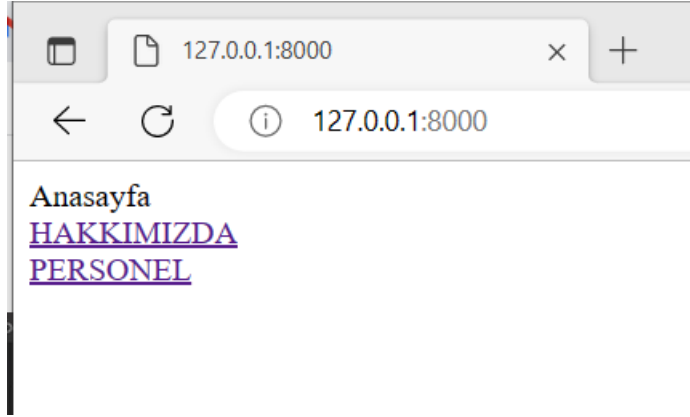
Anasayfada bu sayfaları link olarak vermek için anasayfa.blade.php dosyasına aşağıdaki kodlar eklenir.



The screenshot shows an IDE interface with a project structure on the left and a code editor on the right. The project structure includes folders like public, resources, css, js, views, routes, storage, tests, and vendor. The views folder is expanded, showing anasayfa.blade.php, hakkimizda.blade.php, personel.blade.php, and welcome.blade.php. The code editor displays the content of anasayfa.blade.php, which includes a title 'Anasayfa' and two links: 'HAKKIMIZDA' and 'PERSONEL'.

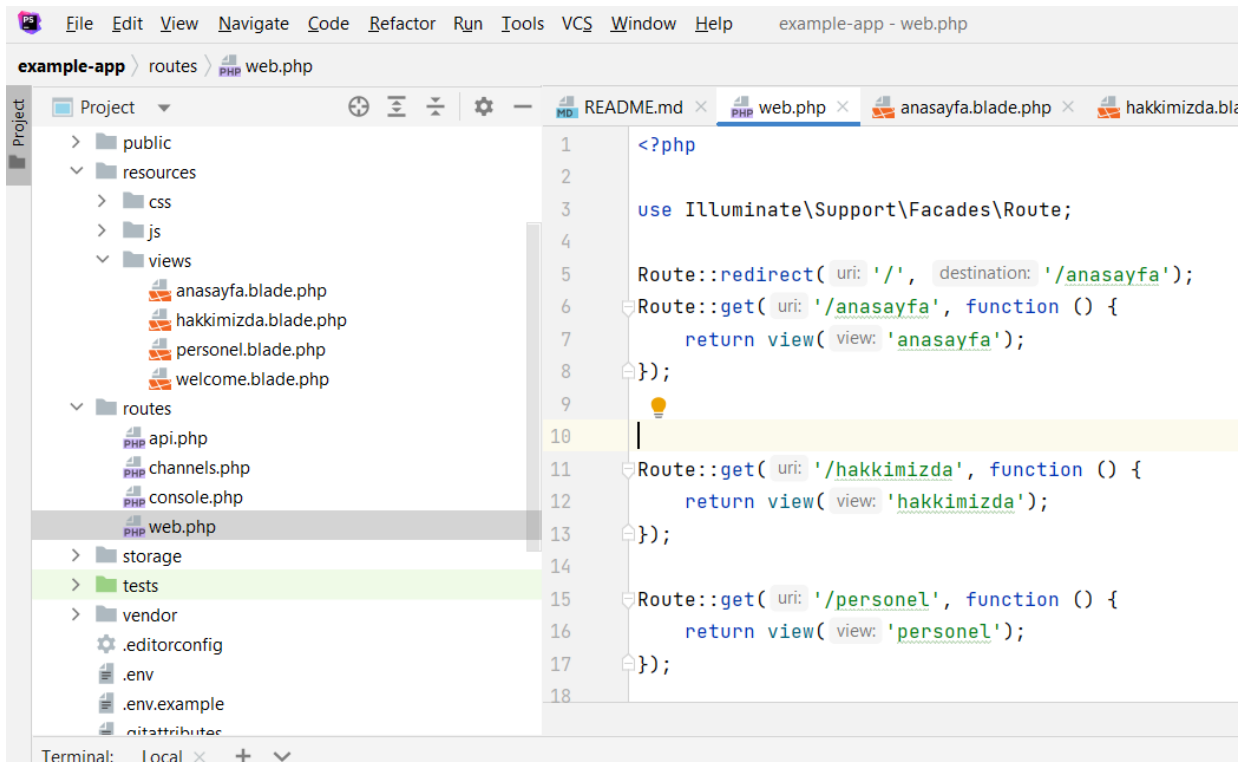
```
1 Anasayfa<br>
2
3 <a href="hakkimizda">HAKKIMIZDA</a><br>
4 <a href="personel">PERSONEL</a><br>
5
```

Sayfadaki linklere tıkladığımızda yönlendirmeler gerçekleştirilir.



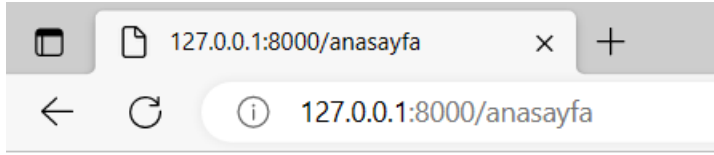
# Redirect Route Oluşturma ve Kullanımı

- `Route::redirect('/', '/anasayfa');`
  - Komutu ile localhost adresine [http://127.0.0.1:8000] girildiğinde anasayfa sekmesine yönlendirir.



```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::redirect(uri: '/', destination: '/anasayfa');
6 Route::get(uri: '/anasayfa', function () {
7     return view(view: 'anasayfa');
8 });
9
10
11 Route::get(uri: '/hakkimizda', function () {
12     return view(view: 'hakkimizda');
13 });
14
15 Route::get(uri: '/personel', function () {
16     return view(view: 'personel');
17 });
18
```





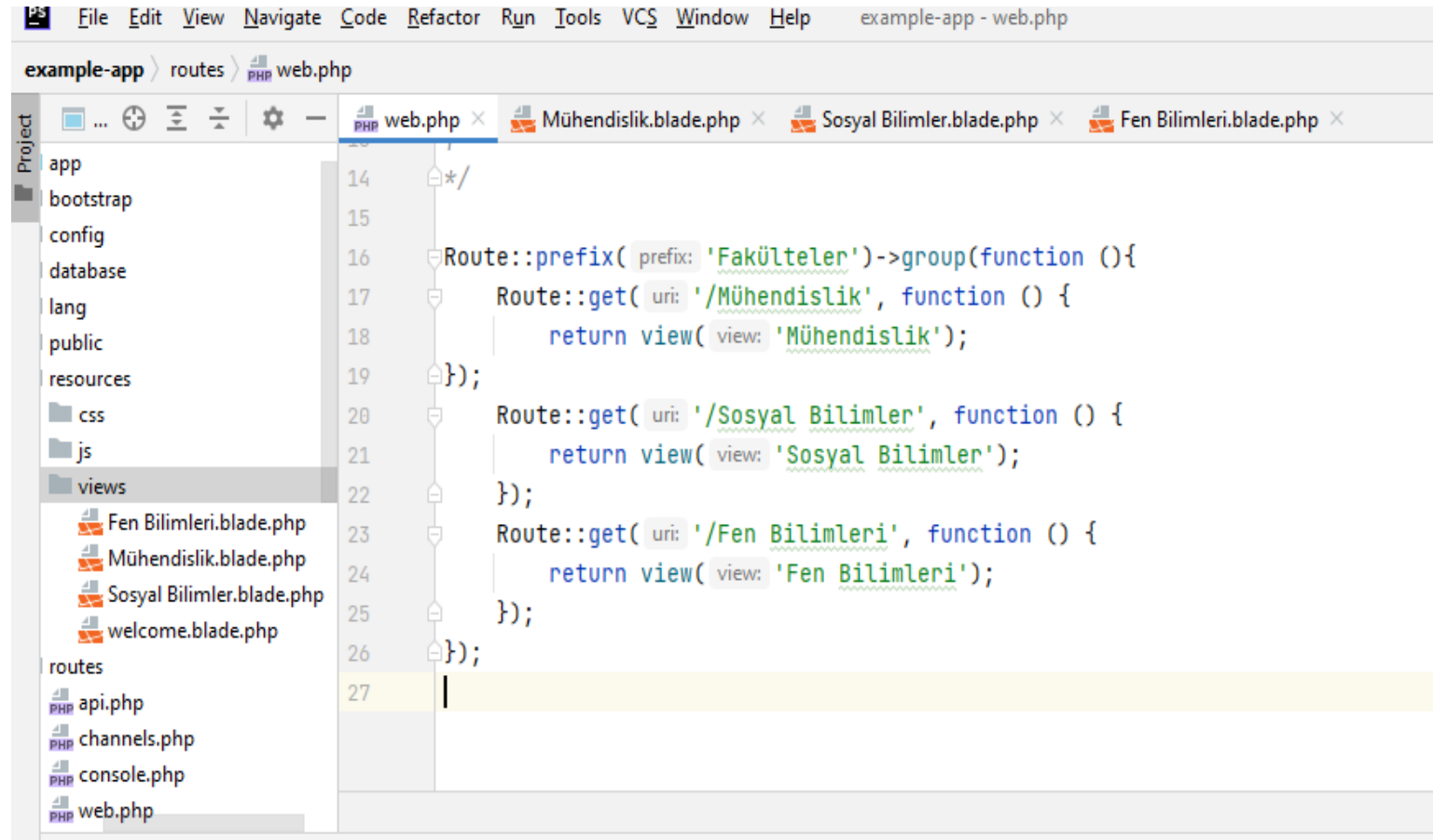
Anasayfa

[HAKKIMIZDA](#)

[PERSONEL](#)

# Laravel 9 Prefix Route Group Oluşturma ve Kullanımı

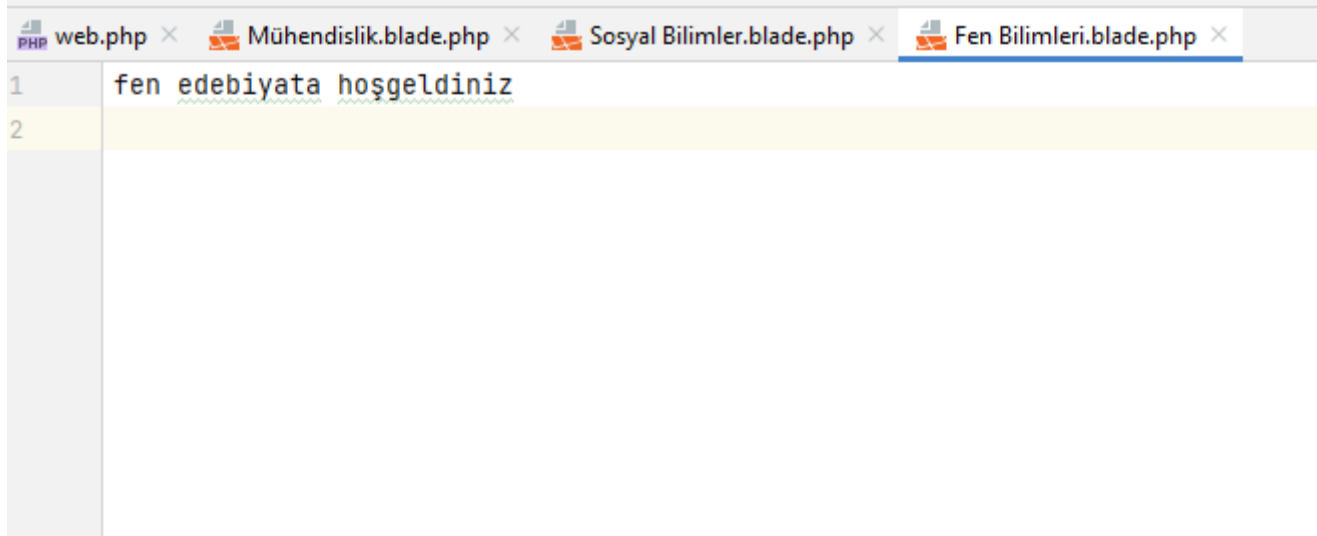
- Rota grubunun oluşturulması
- Her bir route ayrı ayrı tanımlanmak yerine klasör mantığı ile bir grup luşturmak için kullanılır. Aksi hale Route tanımlanırken / olan tüm sayfa isimlerini tek tek girmek gerekecektir.
- Prefix yanda verildiği şekilde oluşturulur.



```
example-app > routes > web.php

14 */
15
16 Route::prefix('Fakülteler')->group(function () {
17     Route::get('/Mühendislik', function () {
18         return view('Mühendislik');
19     });
20     Route::get('/Sosyal Bilimler', function () {
21         return view('Sosyal Bilimler');
22     });
23     Route::get('/Fen Bilimleri', function () {
24         return view('Fen Bilimleri');
25     });
26 });
27
```

- Fen Edebiyat sayfasının içerisini doldurulur



The screenshot shows a web browser with four tabs: 'web.php', 'Mühendislik.blade.php', 'Sosyal Bilimler.blade.php', and 'Fen Bilimleri.blade.php'. The 'Fen Bilimleri.blade.php' tab is active. The browser displays the content 'fen edebiyata hoşgeldiniz' on a yellow background. The text is underlined and has a small red squiggly line under the 'n' in 'hoşgeldiniz'.

```
1 fen edebiyata hoşgeldiniz
2
```

# Laravel 9 Prefix Route Group Oluşturma ve Kullanımı

- Localsayfasında url bölümüne /Fakülteler/Fen Bilimleri yazarsak istediğimiz sayfa açılacaktır.

A screenshot of a web browser's address bar. It features navigation icons (back, forward, refresh) on the left. The address bar itself is a light gray rounded rectangle containing a globe icon and the text "127.0.0.1:8000/Fakülteler/Fen Bilimleri".

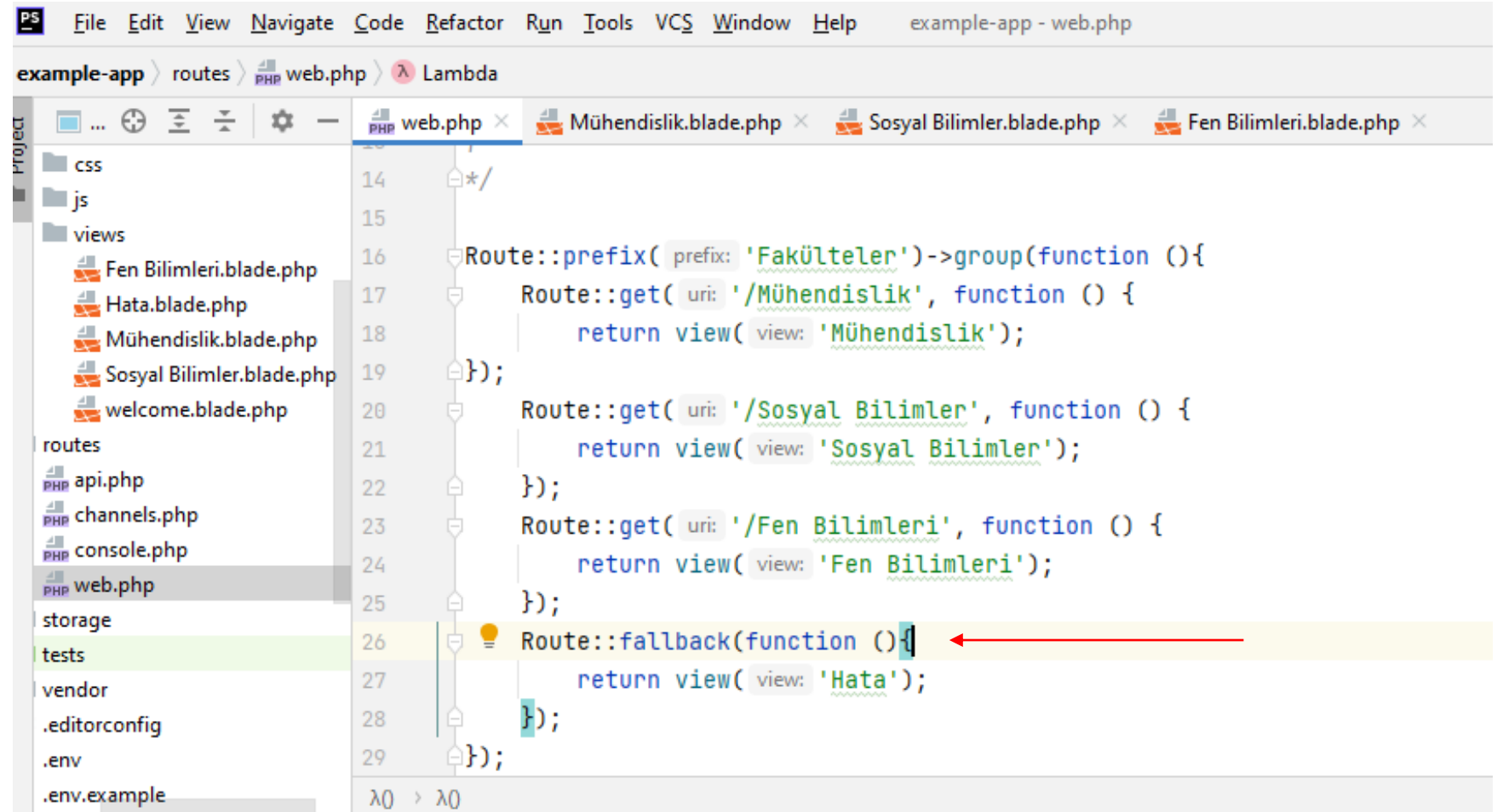
← → ↻ 127.0.0.1:8000/Fakülteler/Fen Bilimleri

---

fen edebiyata hoşgeldiniz

# Laravel 9 Route Fallback Kullanımı

- Eğer link bölümünde yanlış yazım veya farklı bir terim girilirse gitmesini istediğimiz sayfa anlamına gelmektedir. Eğer Fakülteler Route grubunun içerisinde olmasını istiyorsak kod bloğu aşağıdaki gibi yerleştirilir.

The image shows a code editor window for a Laravel application. The left sidebar displays the project structure with folders like css, js, views, routes, storage, tests, and vendor. The 'routes' folder is expanded, showing files like api.php, channels.php, console.php, web.php, and welcome.blade.php. The main editor area shows the 'web.php' file with the following code:

```
14 */
15
16 Route::prefix('Fakülteler')->group(function () {
17     Route::get('/Mühendislik', function () {
18         return view('Mühendislik');
19     });
20     Route::get('/Sosyal Bilimler', function () {
21         return view('Sosyal Bilimler');
22     });
23     Route::get('/Fen Bilimleri', function () {
24         return view('Fen Bilimleri');
25     });
26     Route::fallback(function () {
27         return view('Hata');
28     });
29 });
```

A red arrow points to the 'Route::fallback' line (line 26), highlighting its use as a catch-all route for unmatched paths. The status bar at the bottom shows 'λ0 > λ0'.

# Laravel 9 Route Fallback Kullanımı

- Fakültelerin içerisinde Deneme sayfası olmadığından hata sayfasına yönlendirecektir.

← → ↻ ⓘ 127.0.0.1:8000/Fakülteler/deneme 📄 ☆

ARADIĞINIZ SAYFAYA ULAŞILAMIYOR

# Laravel 9 Route Fallback Kullanımı

- Eğer tüm sayfalarınızda aynı işlem olsun istiyorsanız kod bloğu aşağıdaki gibi yerleştirilir.



The screenshot shows a code editor with four tabs: 'web.php', 'Mühendislik.blade.php', 'Sosyal Bilimler.blade.php', and 'Fen Bilimleri.b'. The 'web.php' tab is active, displaying PHP code for route registration. A red arrow points to the `Route::fallback()` function on line 15, which is used to handle all non-matching routes by returning a 'Hata' view. Below it, a route group is defined with the prefix 'Fakülteler' (line 18). Inside this group, a GET route is registered for '/Mühendislik' (line 19), which returns the 'Mühendislik' view (line 20). The code continues with another GET route for '/Sosyal Bilimler' (line 22) returning the 'Sosyal Bilimler' view (line 23).

```
8 | -----
9 |
10 | / Here is where you can register web routes for your applicat
11 | / routes are loaded by the RouteServiceProvider within a grou
12 | / contains the "web" middleware group. Now create something g
13 | /
14 | */
15 | Route::fallback(function () {
16 |     return view('Hata');
17 | });
18 | Route::prefix('Fakülteler')->group(function () {
19 |     Route::get('/Mühendislik', function () {
20 |         return view('Mühendislik');
21 |     });
22 |     Route::get('/Sosyal Bilimler', function () {
23 |         return view('Sosyal Bilimler');
```

# Laravel 9 Route Fallback Kullanımı

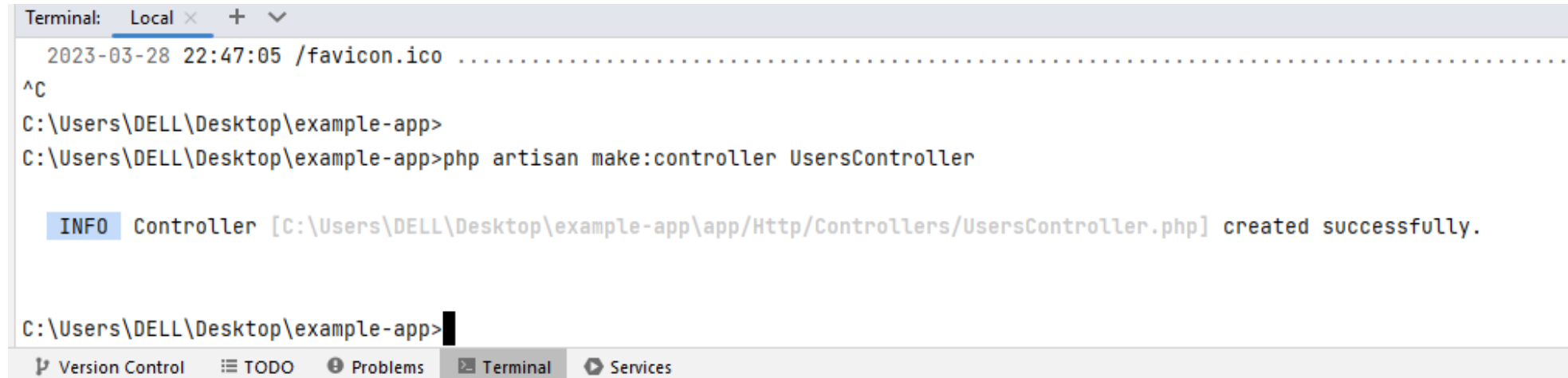
- Deneme sayfası olmadığından Hata sayfasına yönlendirilmiştir.





# Laravel 9 Route Rource Kullanımı

- Öncelikle bir Controller oluşturmamız gerekiyor.
- Bunun için Terminale «php artisan make:controller UsersController» komutu girilir.



```
Terminal: Local x + v
2023-03-28 22:47:05 /favicon.ico .....
^C
C:\Users\DELL\Desktop\example-app>
C:\Users\DELL\Desktop\example-app>php artisan make:controller UsersController

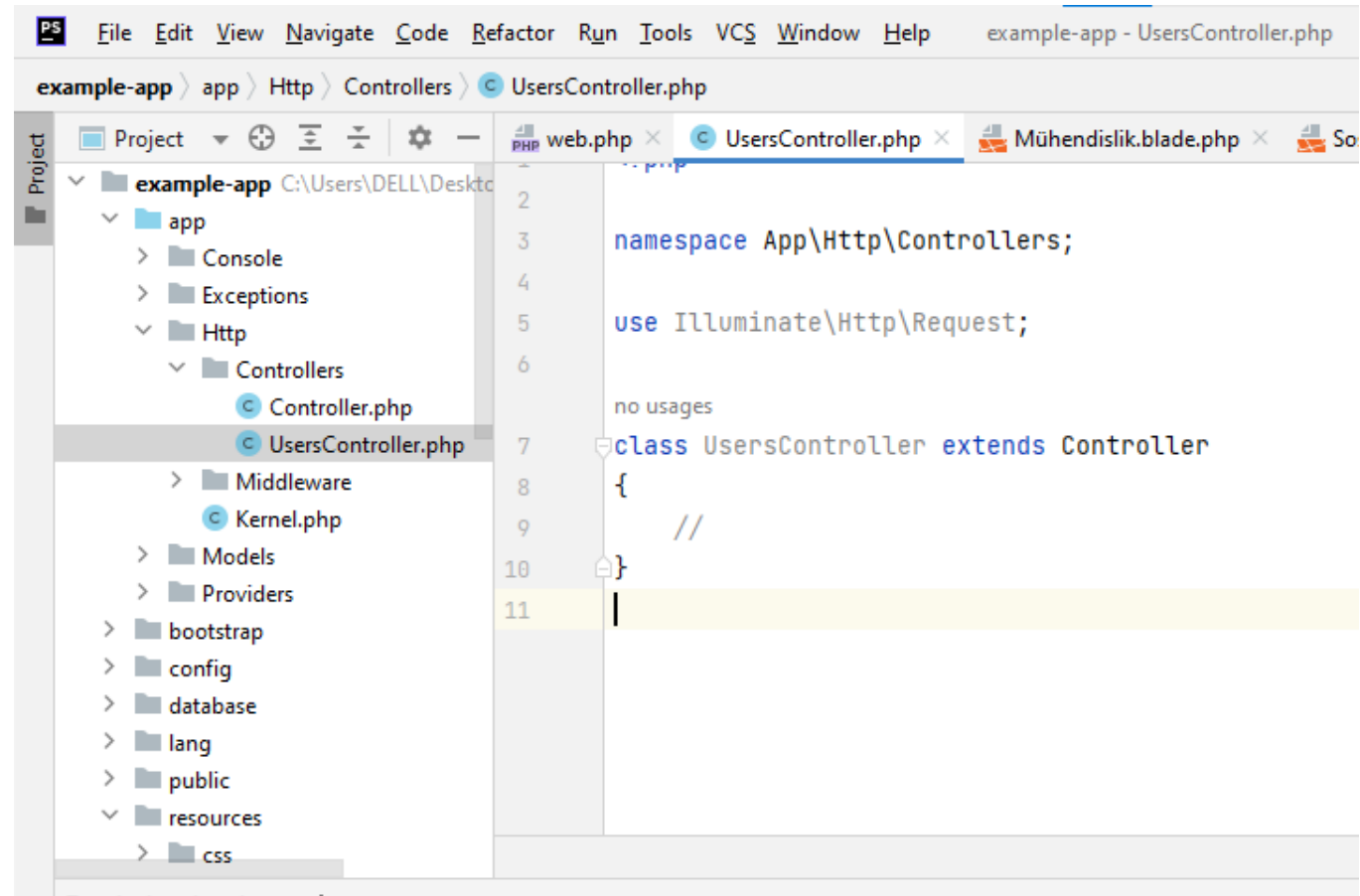
INFO Controller [C:\Users\DELL\Desktop\example-app\app\Http/Controllers/UsersController.php] created successfully.

C:\Users\DELL\Desktop\example-app>
```

The screenshot shows a terminal window with the command `php artisan make:controller UsersController` being executed. The output is an informational message: `INFO Controller [C:\Users\DELL\Desktop\example-app\app\Http/Controllers/UsersController.php] created successfully.` The terminal is part of an IDE interface, with tabs for Version Control, TODO, Problems, Terminal, and Services visible at the bottom.

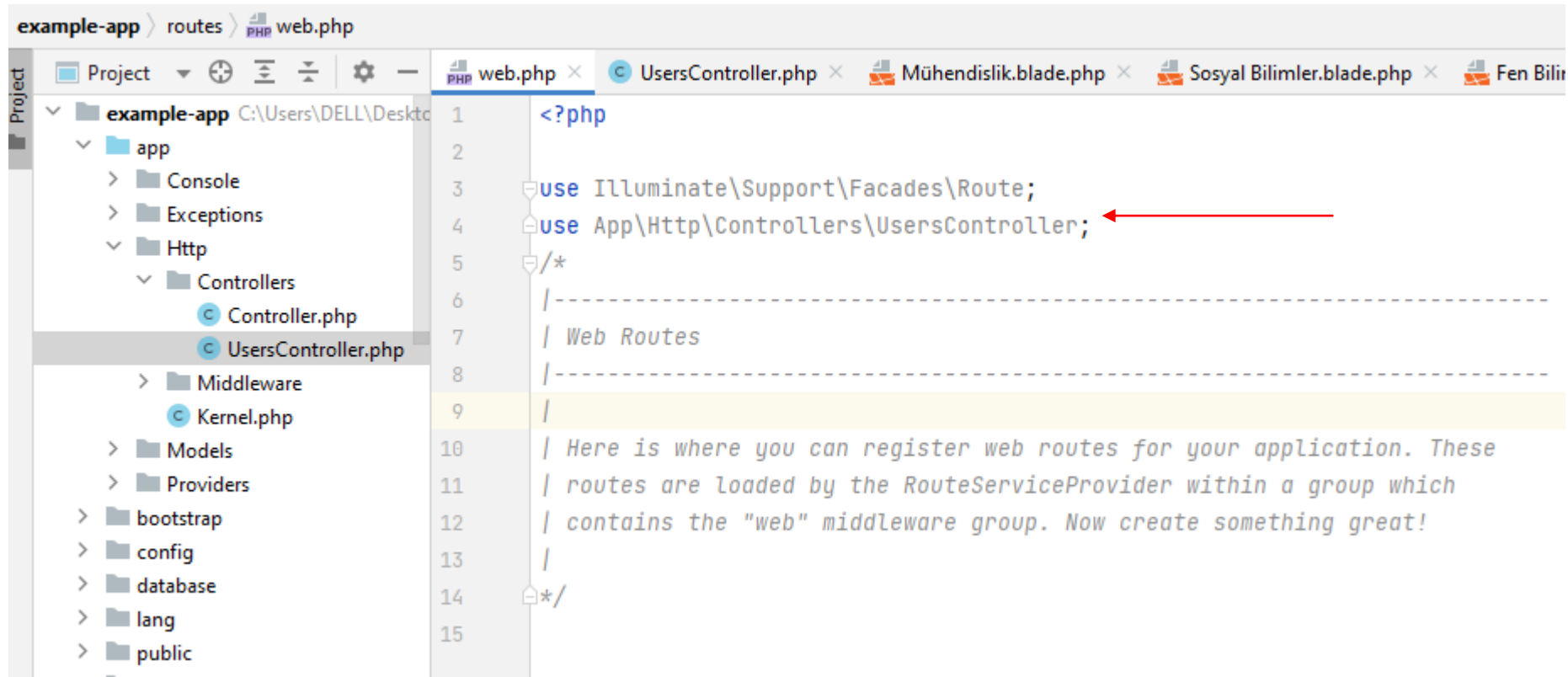
# Laravel 9 Route Rource Kullanımı

- App içerisinde Http klasöründeki Controller içerisinde UsersController oluşturuldu.



# Laravel 9 Route Rource Kullanımı

- Controller Classındaki fonksiyonlara erişebilmemiz için öncelikle kullanmaya izin vermemiz gerekiyor. Bunun için web'e aşağıdaki komut yazılır

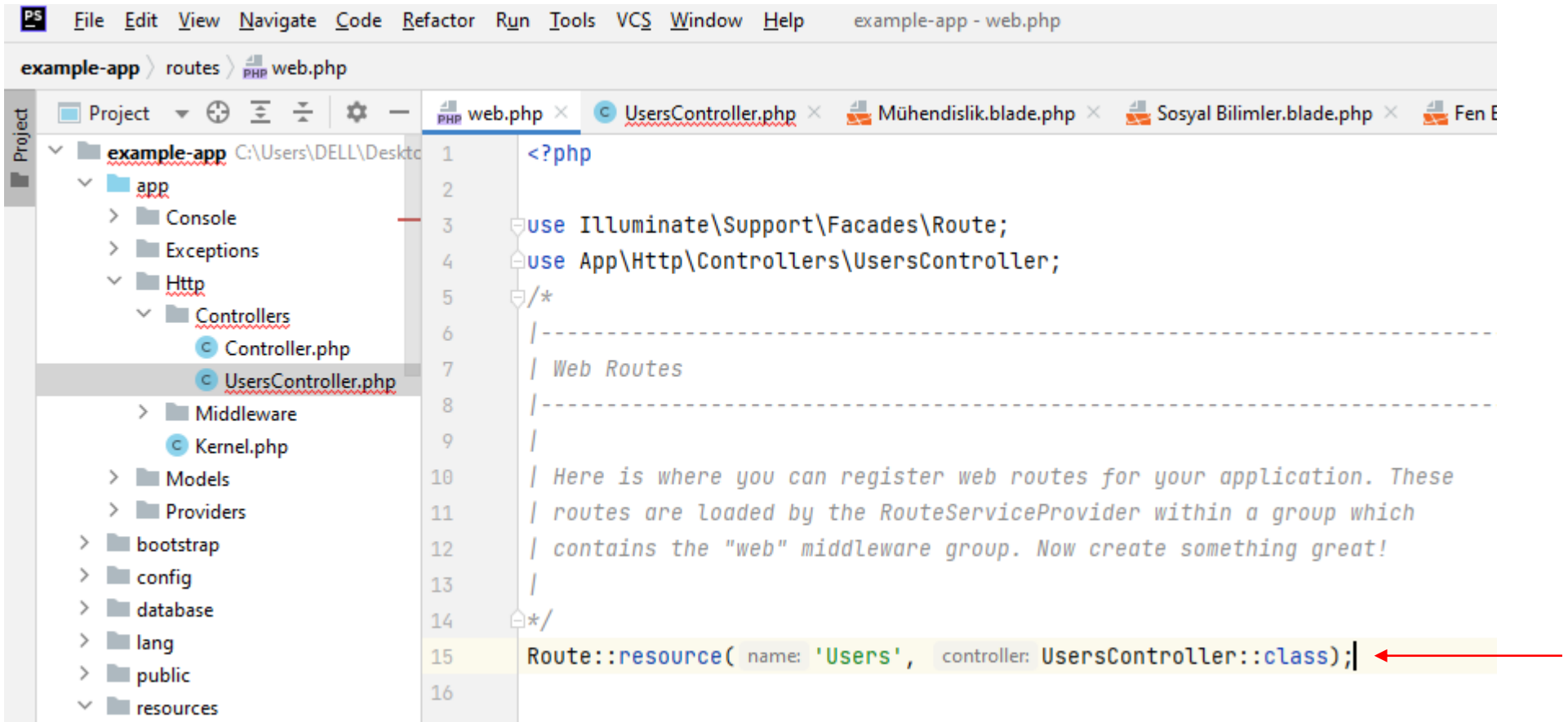


```
example-app > routes > web.php
Project
  example-app C:\Users\DELL\Desktop
    app
      Console
      Exceptions
      Http
        Controllers
          Controller.php
          UsersController.php
        Middleware
          Kernel.php
      Models
      Providers
      bootstrap
      config
      database
      lang
      public

1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\UsersController;
5  /*
6  |-----
7  | Web Routes
8  |-----
9  |
10 | Here is where you can register web routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | contains the "web" middleware group. Now create something great!
13 |
14 */
15
```

# Laravel 9 Route Rource Kullanımı

- UsersController içerisindeki Classlara erişim izninden sonra fonksiyonları kullanmak için aşağıdaki komut yazılır.



```
<?php

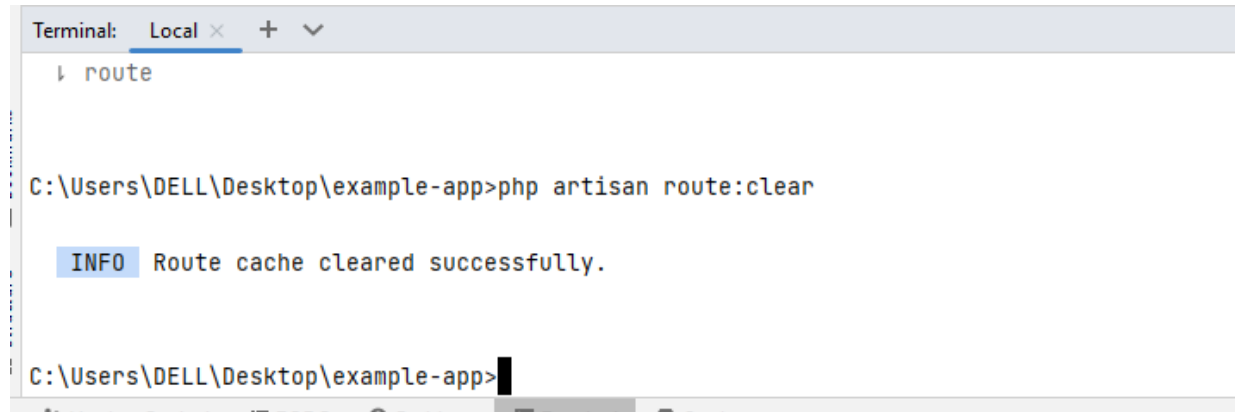
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\UsersController;

/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Route::resource( name: 'Users', controller: UsersController::class);
```

# Laravel 9 Route Rource Kullanımı

- Rotalarda değişiklik yapıldığı için terminale «php artisan route:clear» komutu girilerek temizlenir.



```
Terminal: Local x + v
└─ route

C:\Users\DELL\Desktop\example-app>php artisan route:clear

INFO Route cache cleared successfully.

C:\Users\DELL\Desktop\example-app>
```

- Daha sonra proje «php artisan optimize» komutu ile render edilip
- Proje «php artisan serve» komutu ile tekrar başlatılır

# Laravel 9 Route Rource Kullanımı

- UsersController'da istediğimiz fonksiyonu yazarsak Route ile o fonksiyonu çağırabiliriz.



```
example-app - UsersController.php  
example-app > app > Http > Controllers > UsersController.php > UsersController  
Project | web.php x | UsersController.php x | Mühendislik.blade.php x | Sosyal Bilimler.blade.p  
5 use Illuminate\Http\Request;  
6  
7 2 usages  
8 class UsersController extends Controller  
9 {  
10     no usages  
11     public function index(){  
12         return "index sayfasında bulunuyorsunuz";  
13     }  
14 }
```

# Laravel 9 Route Rource Kullanımı

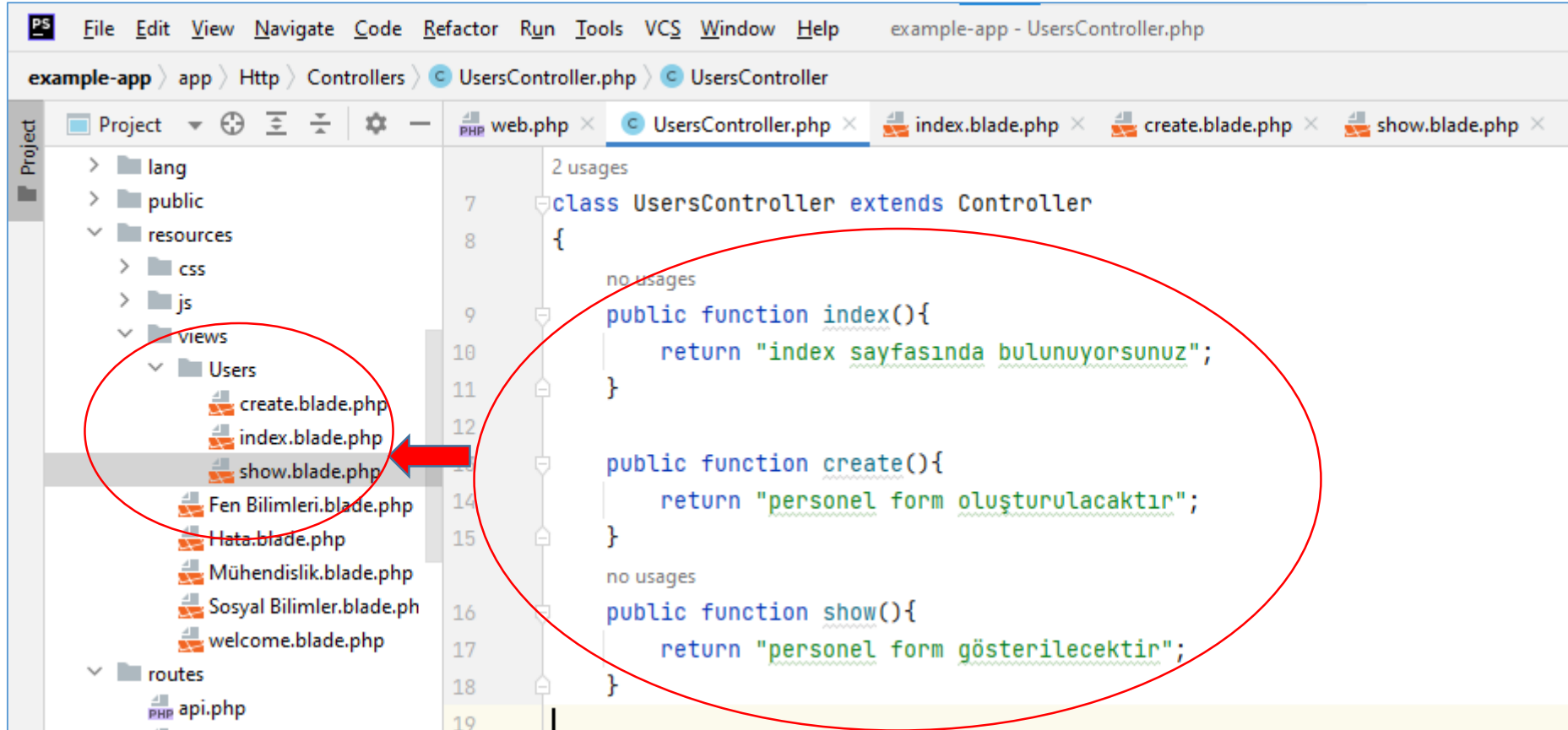
- Kullanıcı sayfamda Users sayfasına gidince ilk olarak index sayfası görülecektir.

A screenshot of a web browser's address bar. It features navigation icons on the left: a back arrow, a forward arrow, and a circular refresh icon. To the right of these icons is a light gray rounded rectangle containing an information icon (a lowercase 'i' inside a circle) and the text '127.0.0.1:8000/Users'.

index sayfasında bulunuyorsunuz

# Laravel 9 Route Rource Kullanımı

- Daha sonra Users içerisinde farklı fonksiyonlar oluşturuyoruz ve Views içerisinde Users klasörü oluşturup her bir fonksiyonun blade dosyasını oluşturuyoruz.





# Laravel 9 Route Rource Kullanımı

- Her bir sayfanın içerisi doldurulur.

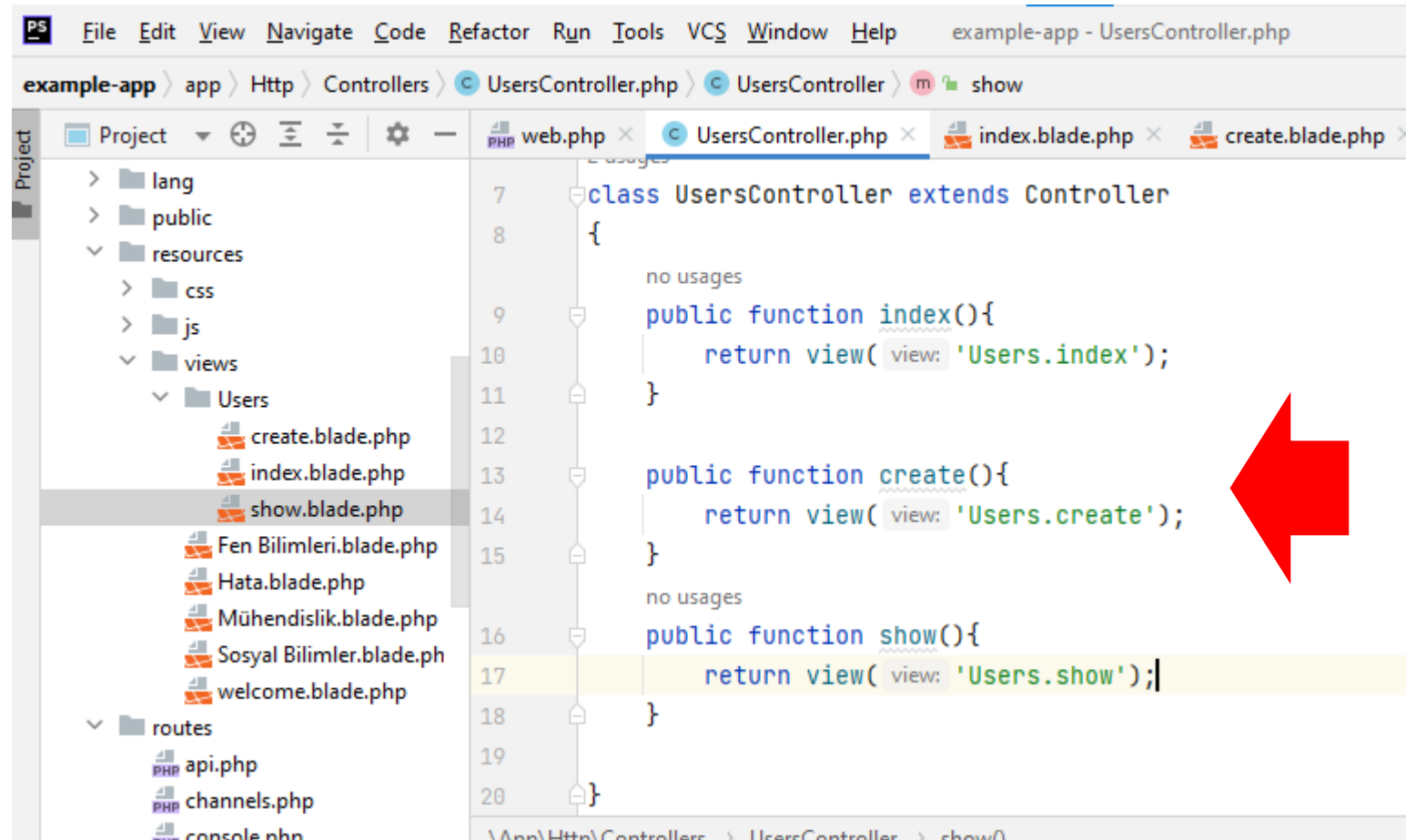
```
web.php x UsersController.php x index.blade.php x
1 Giriş Sayfası
2 |
```

```
web.php x UsersController.php x index.blade.php x create.blade.php x
1 Personel Formu Oluşturulacak
2 ⚡
```

```
web.php x UsersController.php x index.blade.php x create.blade.php x show.blade.php x
1 Personel bilgisi gösterilecek
2 ⚡
```

# Laravel 9 Route Rource Kullanımı

- Fonksiyonları view dosyasındaki sayfalara yönlendirebilmek için fonksiyonlardaki kod bloklarını aşağıdaki gibi düzenleriz.



```
7 class UsersController extends Controller
8 {
9     public function index(){
10         return view( view: 'Users.index');
11     }
12
13     public function create(){
14         return view( view: 'Users.create');
15     }
16
17     public function show(){
18         return view( view: 'Users.show');
19     }
20 }
```

# Laravel 9 Route Rource Kullanımı

- Tekrar terminalden proje temizlenip yeniden başlatılır. Proje çalıştırıldığında sayfalar aşağıdaki gibi oluşturulmuş olacaktır.

