

# LAPORAN PRAKTIKUM 2

## Mata Kuliah Pemrograman Website Lanjut

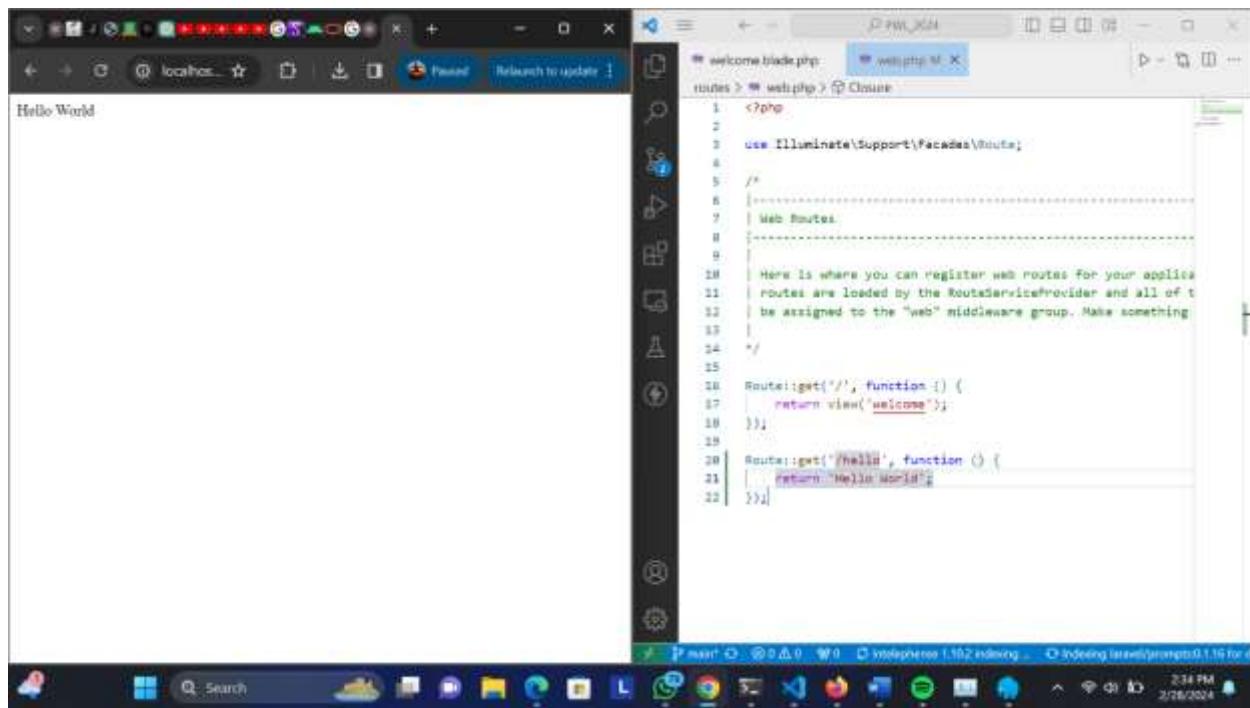


Oleh :

Avicenna Mumtaza  
NIM 2241720112

Teknik Informatika 2H

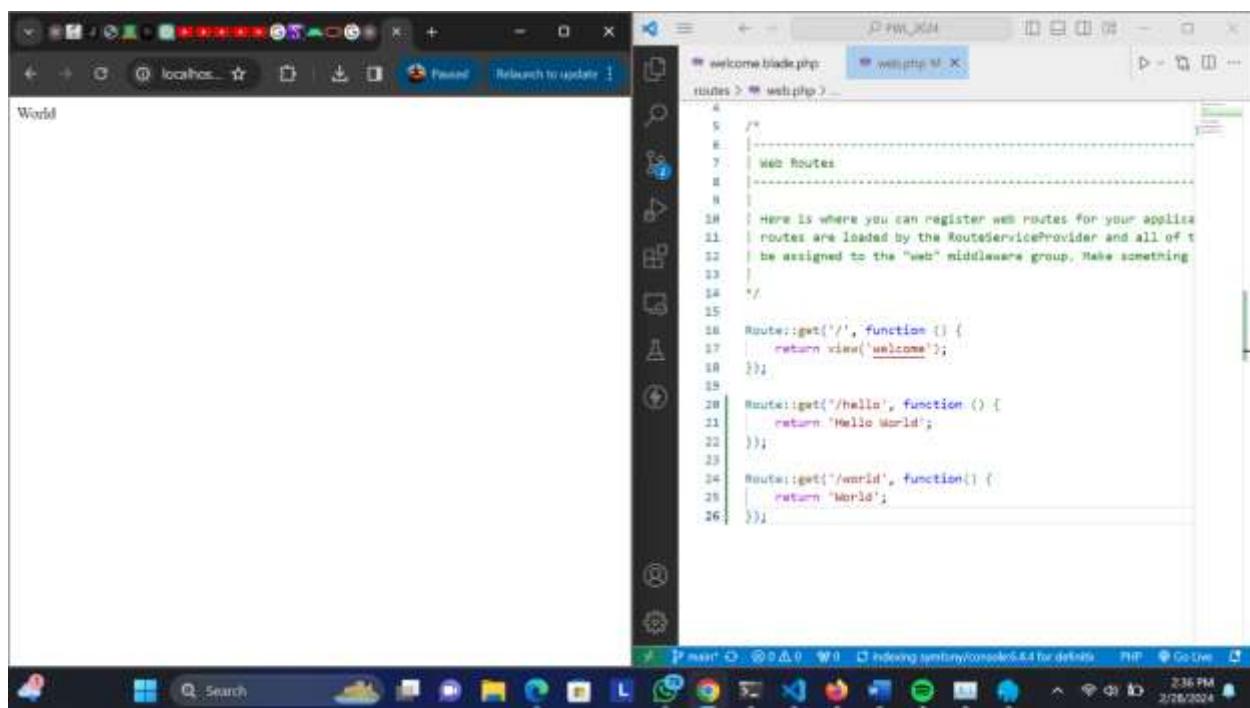
## 1. Routing sederhana untuk menampilkan “Hello World” pada /hello.



A screenshot of a Windows desktop environment. On the left, a browser window shows the text "Hello World". On the right, a code editor displays the contents of the file `routes/web.php`. The code defines two routes: one for the root path '/' returning the view 'welcome', and another for the path '/hello' returning the string 'Hello World'.

```
<?php  
use Illuminate\Support\Facades\Route;  
  
/*  
 | Web Routes:  
 |-----  
 |  
 | Here is where you can register web routes for your application.  
 | Routes are loaded by the RouteServiceProvider and all of them  
 | be assigned to the "web" middleware group. Make something like this:  
 |  
 |>>> Route::get('/', function () {  
 |>>>     return view('welcome');  
 |>>> })->name('home');  
 |  
 |>>> Route::get('/hello', function () {  
 |>>>     return 'Hello World';  
 |>>> })->name('hello');
```

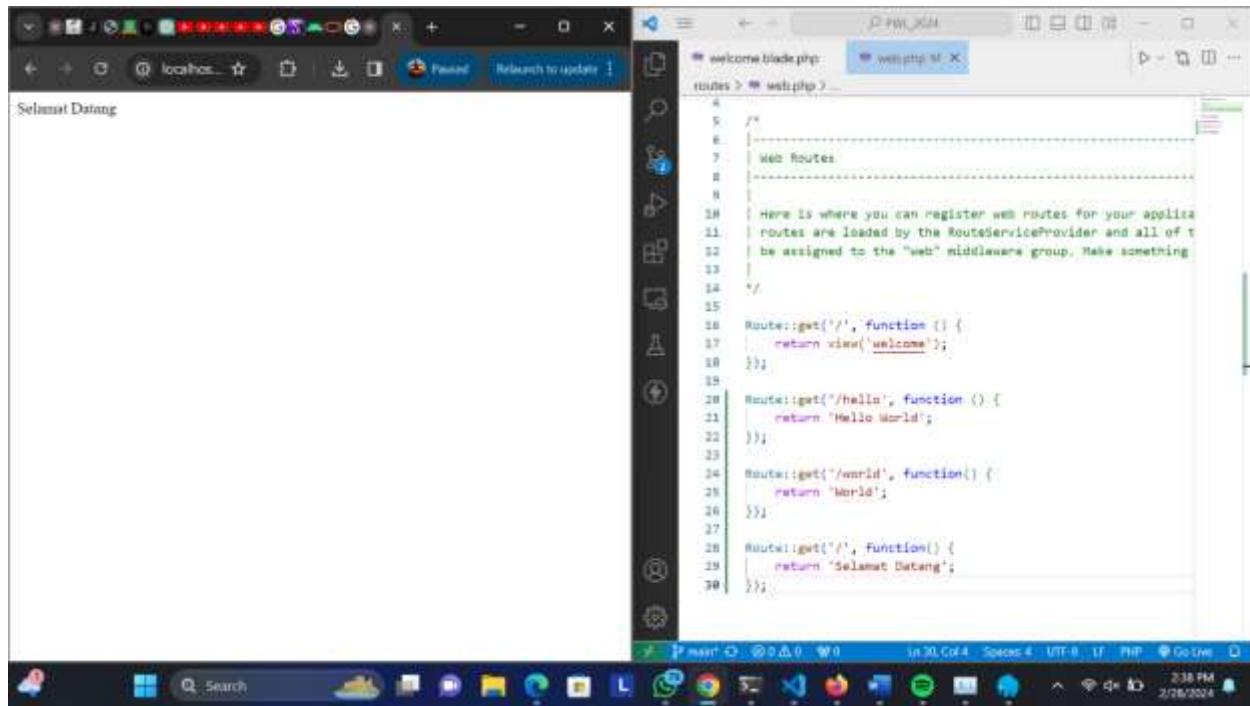
## 2. Routing sederhana untuk menampilkan “World” pada /world.



A screenshot of a Windows desktop environment. On the left, a browser window shows the text "World". On the right, a code editor displays the contents of the file `routes/web.php`. The code defines three routes: one for the root path '/' returning the view 'welcome', one for the path '/hello' returning the string 'Hello World', and one for the path '/world' returning the string 'World'.

```
/*  
 | Web Routes:  
 |-----  
 |  
 | Here is where you can register web routes for your application.  
 | Routes are loaded by the RouteServiceProvider and all of them  
 | be assigned to the "web" middleware group. Make something like this:  
 |  
 |>>> Route::get('/', function () {  
 |>>>     return view('welcome');  
 |>>> })->name('home');  
 |  
 |>>> Route::get('/hello', function () {  
 |>>>     return 'Hello World';  
 |>>> })->name('hello');  
 |  
 |>>> Route::get('/world', function() {  
 |>>>     return 'World';  
 |>>> })->name('world');
```

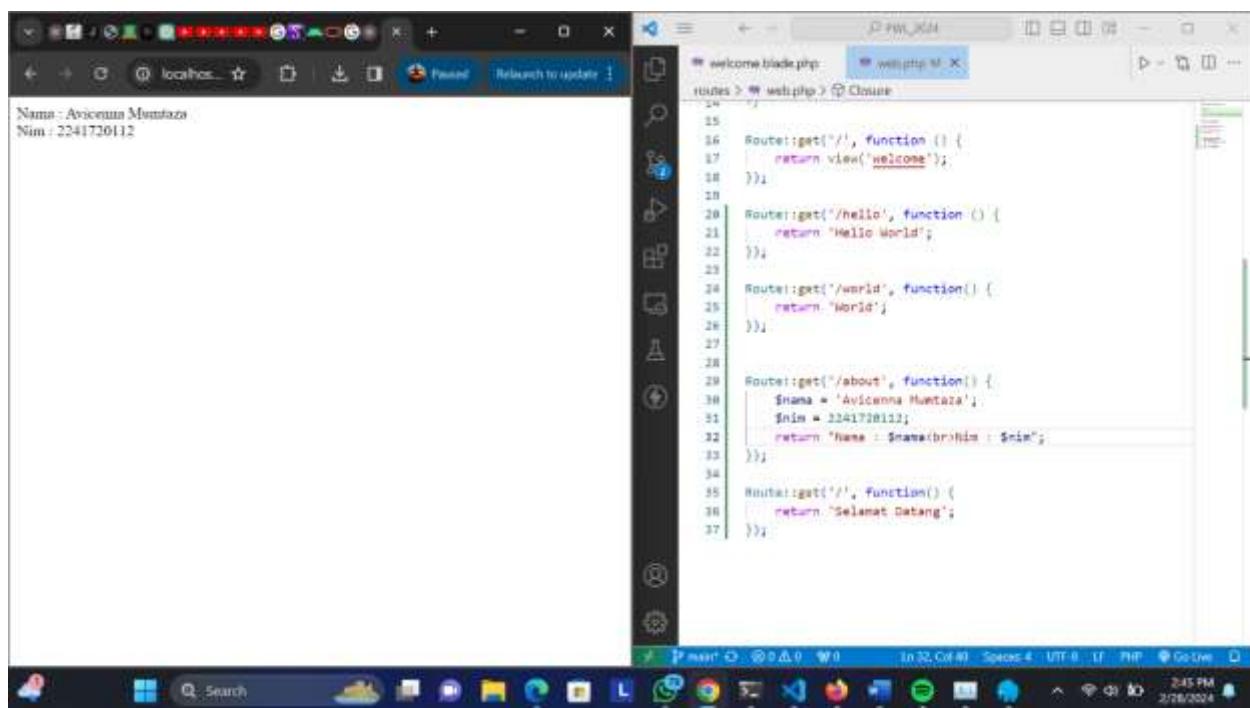
### 3. Menampilkan pesan “Selamat Datang” sebagai route default.



A screenshot of a Windows desktop environment. On the left, a browser window displays the text "Selamat Datang". On the right, a code editor shows the contents of the file `routes/web.php`. The code defines several routes:

```
4
5  /*
6  |--------------------------------------------------------------------------
7   | Web Routes
8   |--------------------------------------------------------------------------
9
10  |
11  | Here is where you can register web routes for your application.
12  | These routes are loaded by the RouteServiceProvider and all of them
13  | will be assigned to the "web" middleware group. Make something
14  | interesting happen!
15  |
16  Route::get('/', function () {
17      return view('welcome');
18  });
19
20  Route::get('/hello', function () {
21      return 'Hello World';
22  });
23
24  Route::get('/world', function() {
25      return 'World';
26  });
27
28  Route::get('/about', function() {
29      return 'Selamat Datang';
30  });
31
32
33
34
35
36
37
```

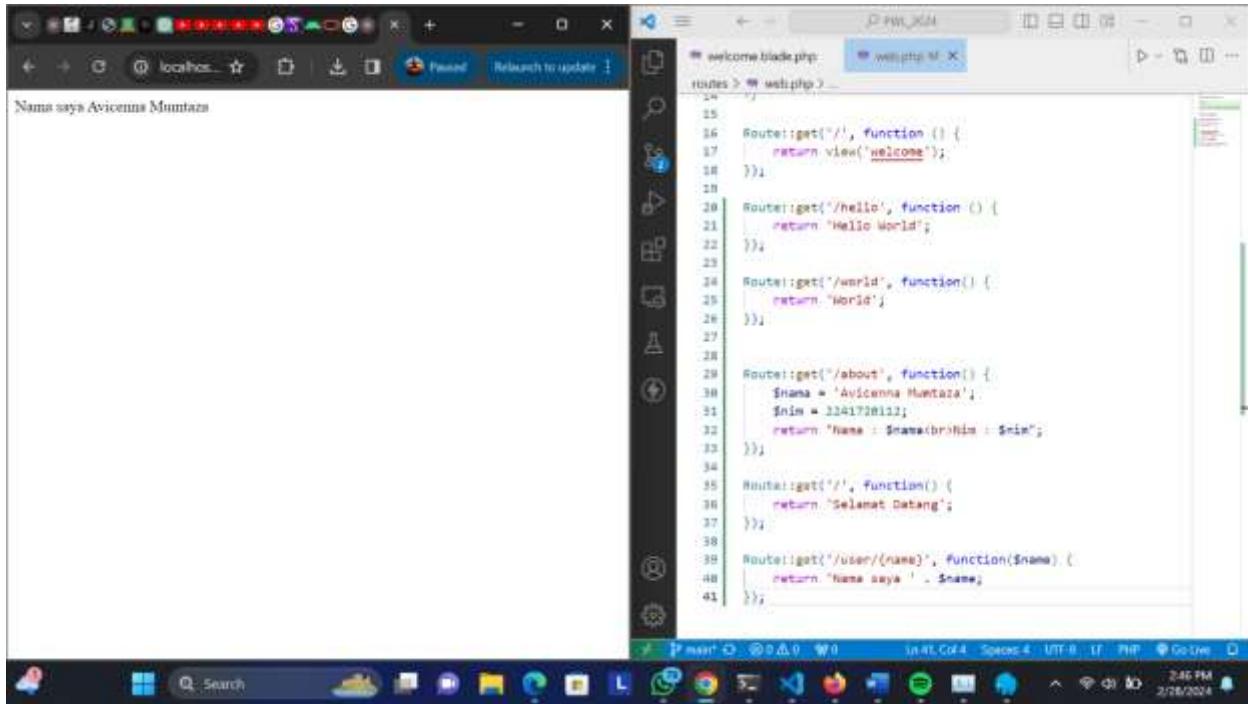
### 4. Menampilkan Nama dan Nim pada /about.



A screenshot of a Windows desktop environment. On the left, a browser window displays the text "Nama : Ayicenna Mumtaza" and "Nim : 2241720112". On the right, a code editor shows the contents of the file `routes/web.php`. The code defines routes for "/about" and "/". The route for "/about" returns a string containing the name and nim. The route for "/" returns the welcome message.

```
14
15
16  Route::get('/', function () {
17      return view('welcome');
18  });
19
20  Route::get('/hello', function () {
21      return 'Hello World';
22  });
23
24  Route::get('/world', function() {
25      return 'World';
26  });
27
28  Route::get('/about', function() {
29      $name = 'Ayicenna Mumtaza';
30      $nim = 2241720112;
31      return "Name : $name(Nim : $nim)";
32  });
33
34  Route::get('/about', function() {
35      return 'Selamat Datang';
36  });
37
```

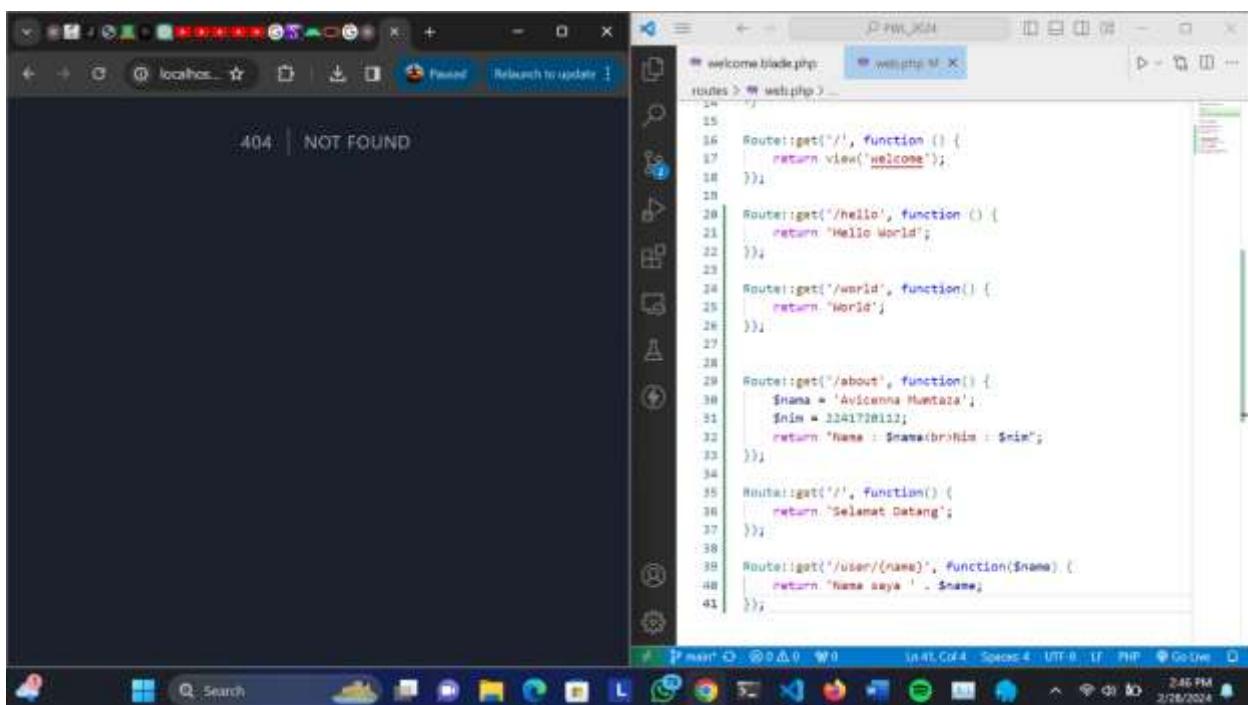
## 5. Menampilkan nama sebagai parameter pada url /user/{name}



A screenshot of a Windows desktop environment. On the left, a browser window shows the URL `localhost:8000` and the page content "Nama saya Avicenna Muntaza". On the right, a code editor window titled "welcome.blade.php" displays PHP code for a Laravel application. The code defines several routes: '/' (returning 'welcome'), '/hello' (returning 'Hello World'), '/world' (returning 'World'), '/about' (returning a string with \$name and \$nim), '/' (returning 'Selamat Datang'), and '/user/{name}' (returning 'Nama saya ' . \$name). The code editor interface includes tabs for "welcome.blade.php" and "web.php", and a status bar at the bottom.

```
routes > web.php:2  
14 //  
15 Route::get('/', function () {  
16     return view('welcome');  
17 });  
18  
19 Route::get('/hello', function () {  
20     return "Hello World";  
21 });  
22  
23 Route::get('/world', function() {  
24     return "World";  
25 });  
26  
27 Route::get('/about', function() {  
28     $name = 'Avicenna Muntaza';  
29     $nim = 2241728112;  
30     return "Name : $name  
Nim : $nim";  
31 });  
32  
33 Route::get('/', function(){  
34     return "Selamat Datang";  
35 });  
36  
37 Route::get('/user/{name}', function($name) {  
38     return "Nama saya ' . $name;  
39 });  
40  
41
```

## 6. Halaman tidak ditemukan karena tidak ada parameter yang dikirim dan fungsi tidak memiliki nilai default pada parameter tersebut.



A screenshot of a Windows desktop environment. On the left, a browser window shows a 404 NOT FOUND error page. On the right, a code editor window titled "welcome.blade.php" displays the same PHP code as the previous screenshot. The code defines the same set of routes, including the '/user/{name}' route which returns 'Nama saya ' . \$name. The code editor interface includes tabs for "welcome.blade.php" and "web.php", and a status bar at the bottom.

```
routes > web.php:2  
14 //  
15 Route::get('/', function () {  
16     return view('welcome');  
17 });  
18  
19 Route::get('/hello', function () {  
20     return "Hello World";  
21 });  
22  
23 Route::get('/world', function() {  
24     return "World";  
25 });  
26  
27 Route::get('/about', function() {  
28     $name = 'Avicenna Muntaza';  
29     $nim = 2241728112;  
30     return "Name : $name  
Nim : $nim";  
31 });  
32  
33 Route::get('/', function(){  
34     return "Selamat Datang";  
35 });  
36  
37 Route::get('/user/{name}', function($name) {  
38     return "Nama saya ' . $name;  
39 });  
40  
41
```

7. Menampilkan pesan di bawah menggunakan 2 parameter pada `/posts/{post}/comments/{comment}`.

```
Pos ke-1 komentar ke-5

Route::get('/hello', function () {
    return 'Hello World';
});

Route::get('/world', function() {
    return 'World';
});

Route::get('/about', function() {
    $name = 'Auliaenna Muettasa';
    $nim = '2241728112';
    return "Nama : $name  
Nim : $nim";
});

Route::get('/', function() {
    return 'Selamat Datang';
});

Route::get('/user/{name}', function($name) {
    return "Nama saya " . $name;
});

Route::get('/posts/{post}/comments/{comment}', function($postId, $commentId) {
    return "Pos ke-". $postId . " komentar ke-$commentId";
});
```

8. Menampilkan informasi di bawah menggunakan 1 parameter.

```
Halaman Artikel dengan ID 20

Route::get('/hello', function () {
    return 'Hello World';
});

Route::get('/about', function() {
    $name = 'Auliaenna Muettasa';
    $nim = '2241728112';
    return "Nama : $name  
Nim : $nim";
});

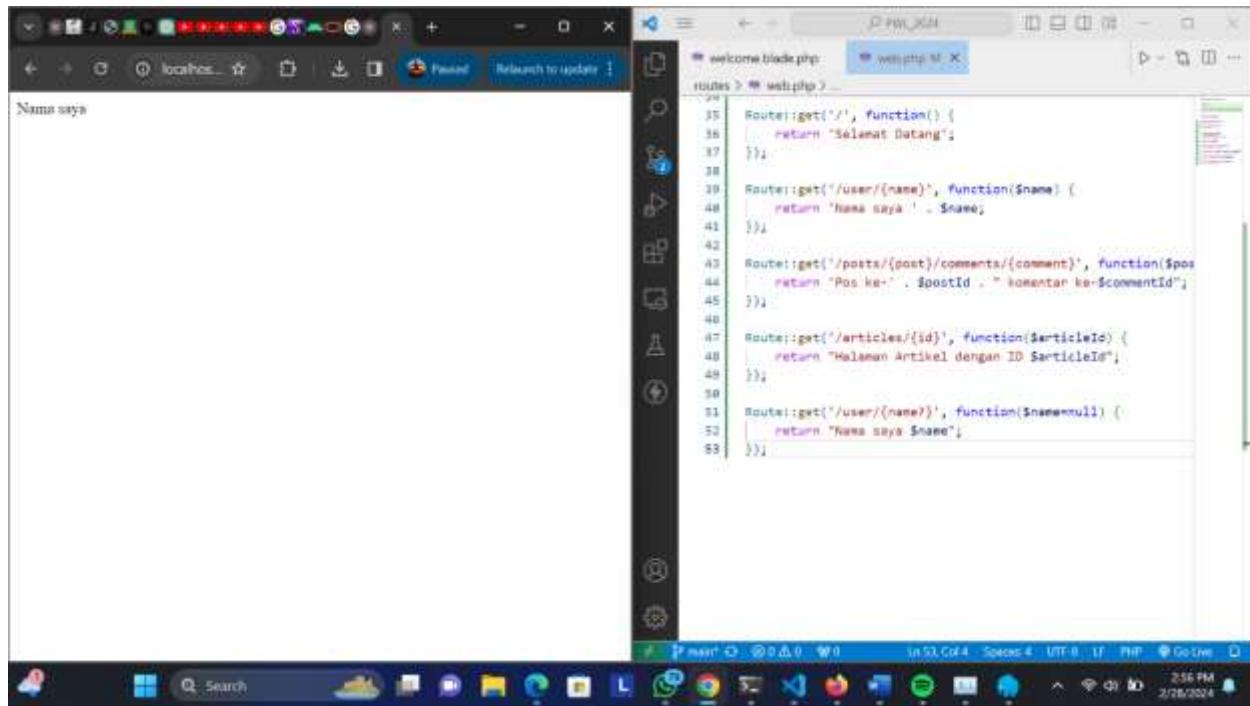
Route::get('/', function() {
    return 'Selamat Datang';
});

Route::get('/user/{name}', function($name) {
    return "Nama saya " . $name;
});

Route::get('/posts/{post}/comments/{comment}', function($postId, $commentId) {
    return "Pos ke-". $postId . " komentar ke-$commentId";
});

Route::get('/articles/{id}', function($articleId) {
    return "Halaman Artikel dengan ID $articleId";
});
```

## 9. Halaman ditampilkan tapi parameter default tidak bernilai (null).

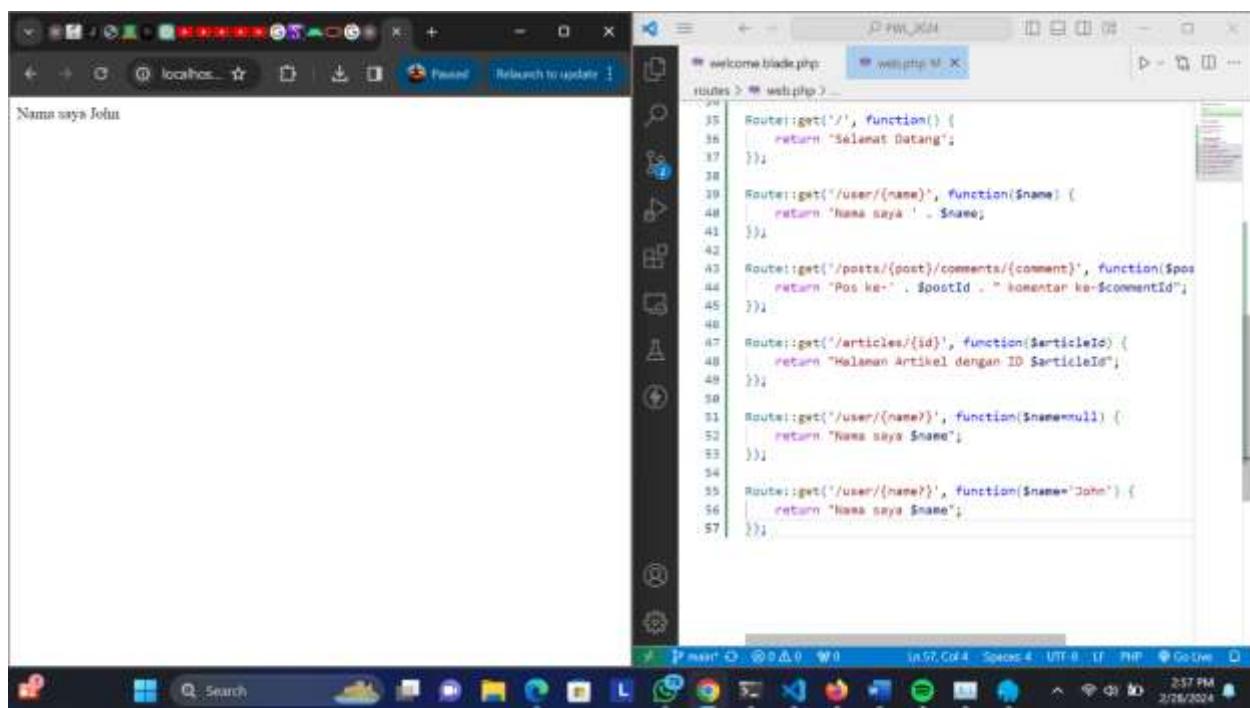


A screenshot of a Windows desktop environment. On the left, a browser window shows the URL `localhost:8000` and displays the text "Nama saya". On the right, a code editor window titled "welcome.blade.php" shows PHP code defining routes. The code includes a route for the root path that returns "Selamat Datang". Another route handles user names, returning "Nama saya \$name". A third route handles post comments, returning "Pos ke-\$postId komentar ke-\$commentId". A fourth route handles article IDs, returning "Halaman Artikel dengan ID \$articleId". A fifth route handles user names again, returning "Nama saya \$name". A sixth route handles user names with a default value of "John", returning "Nama saya \$name".

```
routes > welcome.blade.php
```

```
35 Router::get('/', function() {
36     return "Selamat Datang";
37 });
38
39 Router::get('/user/{name}', function($name) {
40     return "Nama saya " . $name;
41 });
42
43 Route::get('/posts/{post}/comments/{comment}', function($pos
44     return "Pos ke-" . $postId . " komentar ke-$commentId";
45 );
46
47 Route::get('/articles/{id}', function($articleId) {
48     return "Halaman Artikel dengan ID $articleId";
49 );
50
51 Router::get('/user/{name?}', function($name=null) {
52     return "Nama saya $name";
53 });
54
55 Router::get('/user/{name?}', function($name='John') {
56     return "Nama saya $name";
57 });
```

## 10. Halaman ditampilkan menggunakan john sebagai default parameter.

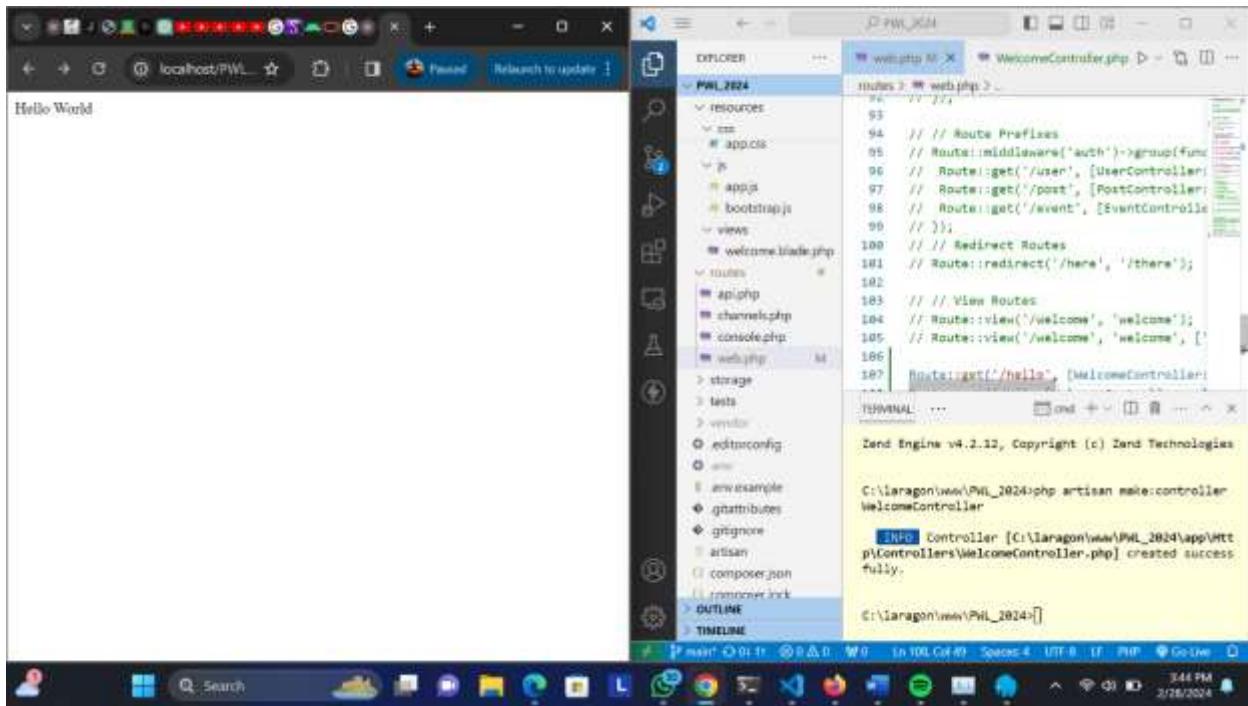


A screenshot of a Windows desktop environment, similar to the previous one. The browser window now displays "Nama saya John". The code editor window titled "welcome.blade.php" shows the same PHP code as before, but the route for user names now uses the default value "John" instead of null. This results in the displayed name being "Nama saya John".

```
routes > welcome.blade.php
```

```
35 Router::get('/', function() {
36     return "Selamat Datang";
37 });
38
39 Router::get('/user/{name}', function($name) {
40     return "Nama saya " . $name;
41 });
42
43 Route::get('/posts/{post}/comments/{comment}', function($pos
44     return "Pos ke-" . $postId . " komentar ke-$commentId";
45 );
46
47 Route::get('/articles/{id}', function($articleId) {
48     return "Halaman Artikel dengan ID $articleId";
49 );
50
51 Router::get('/user/{name?}', function($name=null) {
52     return "Nama saya $name";
53 });
54
55 Router::get('/user/{name?}', function($name='John') {
56     return "Nama saya $name";
57 });
```

## 11. Menampilkan “Hello World” menggunakan fungsi di controller.

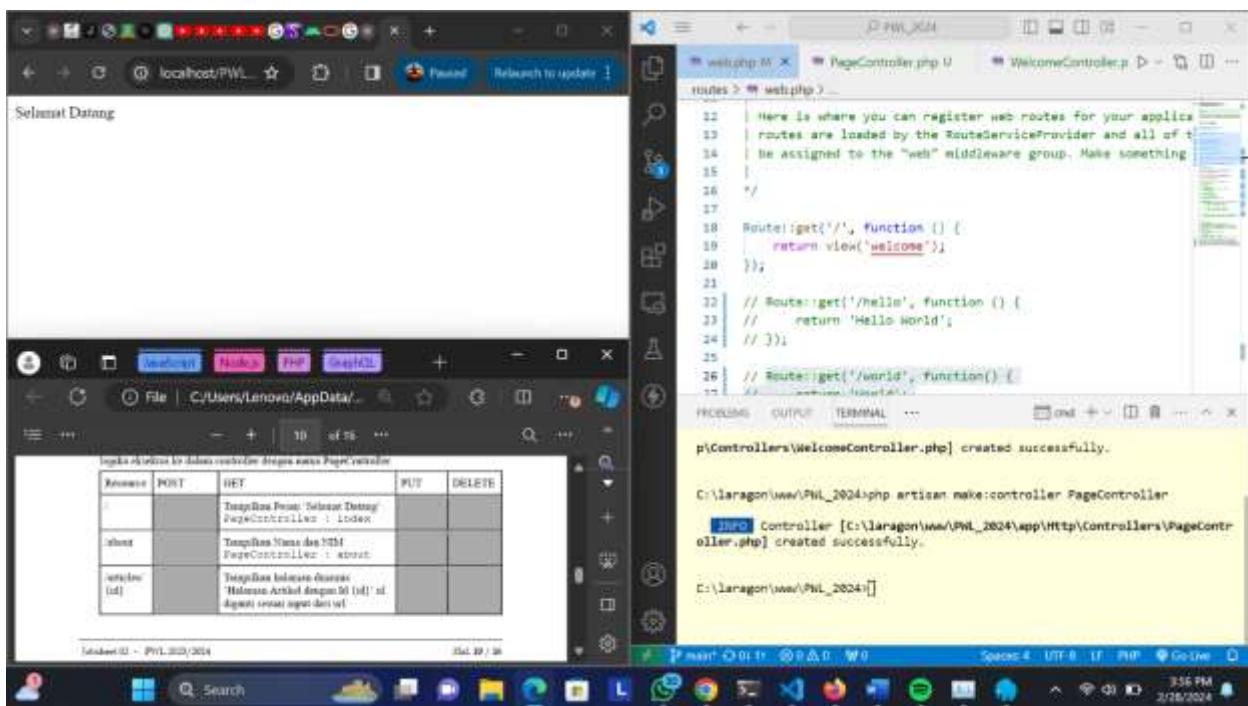


The screenshot shows a Windows desktop environment. On the left, a browser window displays the text "Hello World". On the right, a PHPStorm IDE window is open, showing the routes file (`routes/web.php`). The code defines a route for the URL `/hello` to the `WelcomeController`:

```
Route::get('/hello', [WelcomeController::class, 'index']);
```

Below the routes file, a terminal window shows the command `php artisan make:controller WelcomeController` being run, followed by a success message indicating the controller was created successfully.

## 12. Menampilkan pesan berikut menggunakan fungsi di controller.



The screenshot shows a Windows desktop environment. On the left, a browser window displays the text "Selamat Datang". On the right, a PHPStorm IDE window is open, showing the routes file (`routes/web.php`). The code defines a route for the URL `/` to the `WelcomeController`:

```
Route::get('/', [WelcomeController::class, 'index']);
```

Below the routes file, a terminal window shows the command `php artisan make:controller WelcomeController` being run, followed by a success message indicating the controller was created successfully.

### 13. Menampilkan Nama dan Nim menggunakan fungsi di controller.

The screenshot shows a Windows desktop environment. On the left, a browser window displays the URL `localhost/PNL_2024`. The page content is "Nama : Avicenna Muntaza" and "Nim : 2241720112". On the right, a terminal window titled "PANEL\_XAM" shows the command `php artisan make:controller PageController` being run. The output indicates that the controller was created successfully. Below the terminal is a standard Windows taskbar with various icons.

```
routes > web.php (2)

12 | Here is where you can register web routes for your application.
13 | Routes are loaded by the RouteServiceProvider and all of them
14 | will be assigned to the "web" middleware group. Make something
15 |
16 |
17 | Route::get('/', function () {
18 |     return view('welcome');
19 | });
20 |
21 |
22 | // Route::get('/hello', function () {
23 | //     return 'Hello World!';
24 | // });
25 |
26 | // Route::get('/world', function() {
27 | //     return 'Hello World!';
28 | });

PROBLEMS OUTPUT TERMINAL ...
p\Controllers\WelcomeController.php] created successfully.

C:\laragon\www\PNL_2024>php artisan make:controller PageController
[INFO] Controller [C:\laragon\www\PNL_2024\app\Http\Controllers\PageController.php] created successfully.

E:\laragon\www\PNL_2024>[]
```

### 14. Menampilkan informasi di bawah menggunakan fungsi di controller.

The screenshot shows a Windows desktop environment. On the left, a browser window displays the URL `localhost/PNL_2024`. The page content is "Halaman article dengan id 99". On the right, a terminal window titled "PANEL\_XAM" shows the command `php artisan make:controller PageController` being run. The output indicates that the controller was created successfully. Below the terminal is a standard Windows taskbar with various icons.

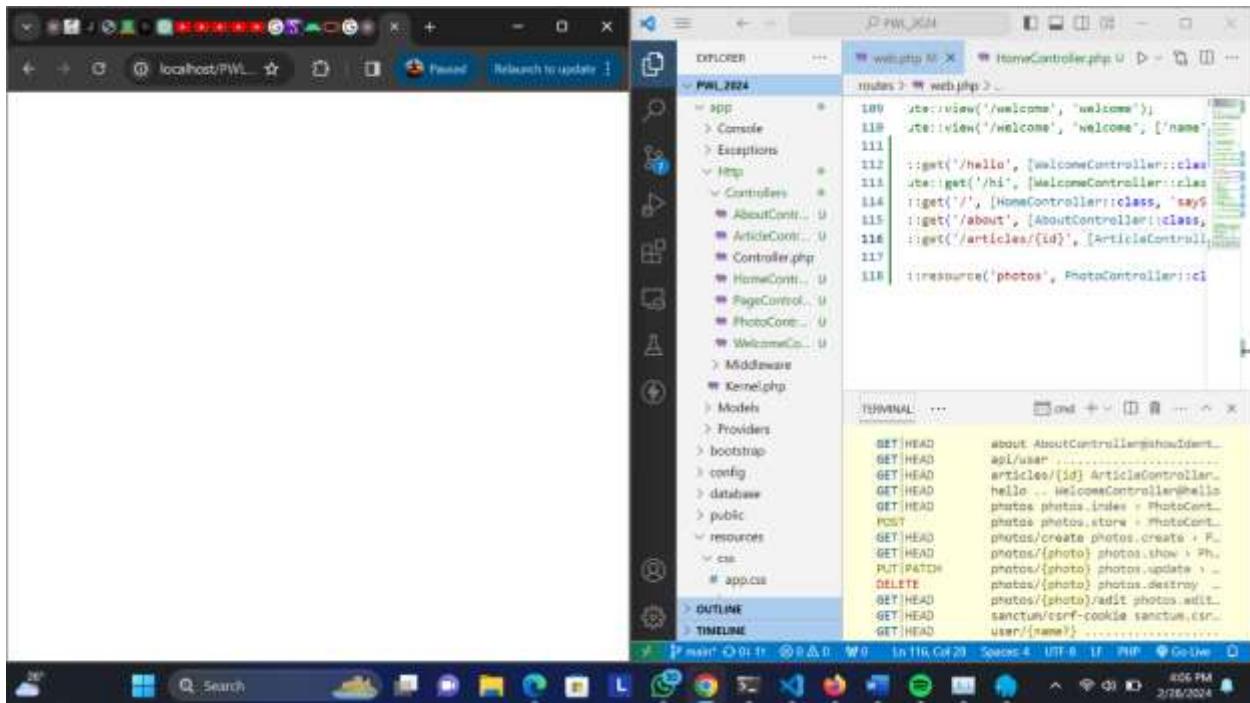
```
php artisan make:controller PageController
[INFO] Controller [C:\laragon\www\PNL_2024\app\Http\Controllers\PageController.php] created successfully.

p\Controllers\WelcomeController.php] created successfully.

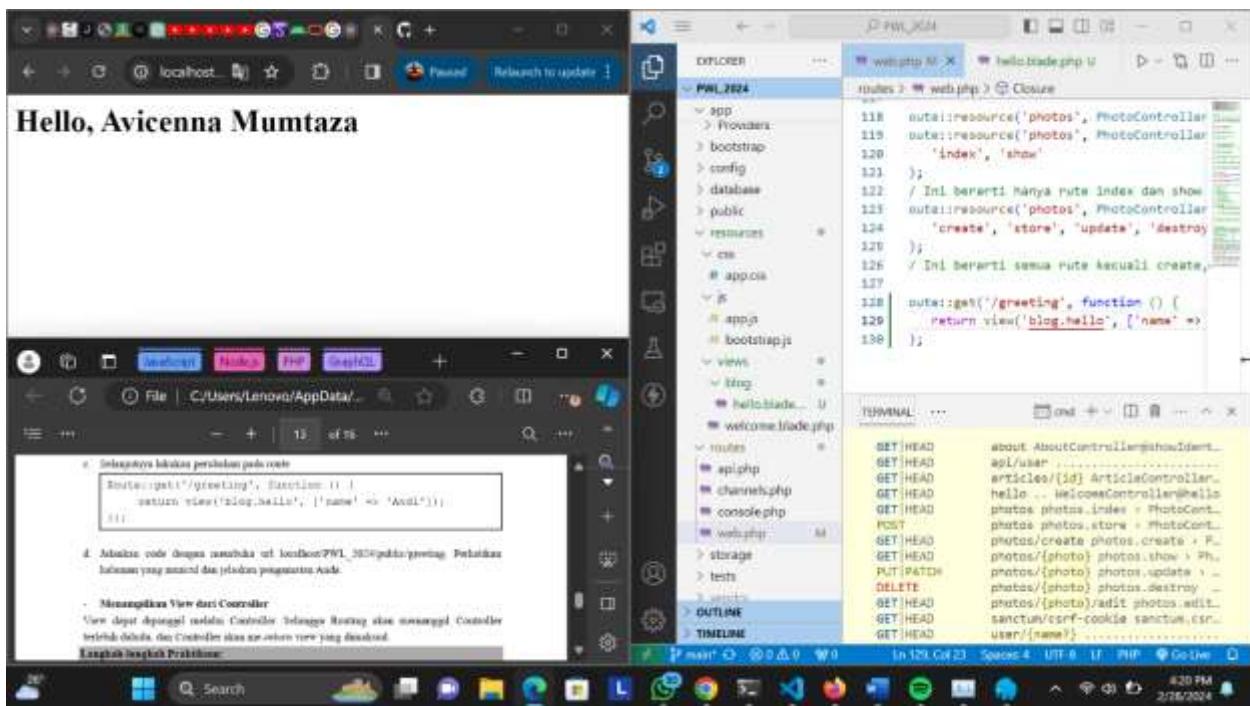
C:\laragon\www\PNL_2024>php artisan make:controller PageController
[INFO] Controller [C:\laragon\www\PNL_2024\app\Http\Controllers\PageController.php] created successfully.

E:\laragon\www\PNL_2024>[]
```

## 15. Memisahkan implementasi sebelumnya dengan konsep *Single Action Controller*.



## 16. Only dan except sebagai filter method yang akan digunakan.



## 17. Membuat view sederhana sebagai return value dari routing.

The screenshot shows a Windows desktop environment. In the top-left corner, there's a browser window displaying the text "Hello, Avis". Below it, a code editor window shows PHP code for a routes file:

```
resources > web.php >
120     'index', 'show'
121   );
122   // Ini berarti hanya route index dan show dari controller Photo
123   Route::resource('photos', PhotoController::class)->except([
124     'create', 'store', 'update', 'destroy'
125   ]);
126   // Ini berarti semua route kecuali create, store, update, dan
127   // destroy
128   // Route::get('/greeting', function () {
129   //     return view('blog.hello', ['name' => 'Avicenna Rautaw']);
130   // });
131   Route::get('/greeting', [WelcomeController::class, 'greeting']
```

Below the code editor is a terminal window showing a list of routes:

```
POST      photos/photos.store = PhotoController@store
GET|HEAD  photos/photos.create = PhotoController@create
GET|HEAD  photos/(:photo) photos.show = PhotoController@show
PUT|PATCH photos/(:photo) photos.update = PhotoController@update
DELETE    photos/(:photo) photos.destroy = PhotoController@destroy
GET|HEAD  photos/(:photo)/edit = PhotoController@edit
GET|HEAD  sanctum/csrf-cookie = sanctum/csrf-cookie
GET|HEAD  user/{name?} ...
```

The terminal also shows the command "Shading [17] routes".

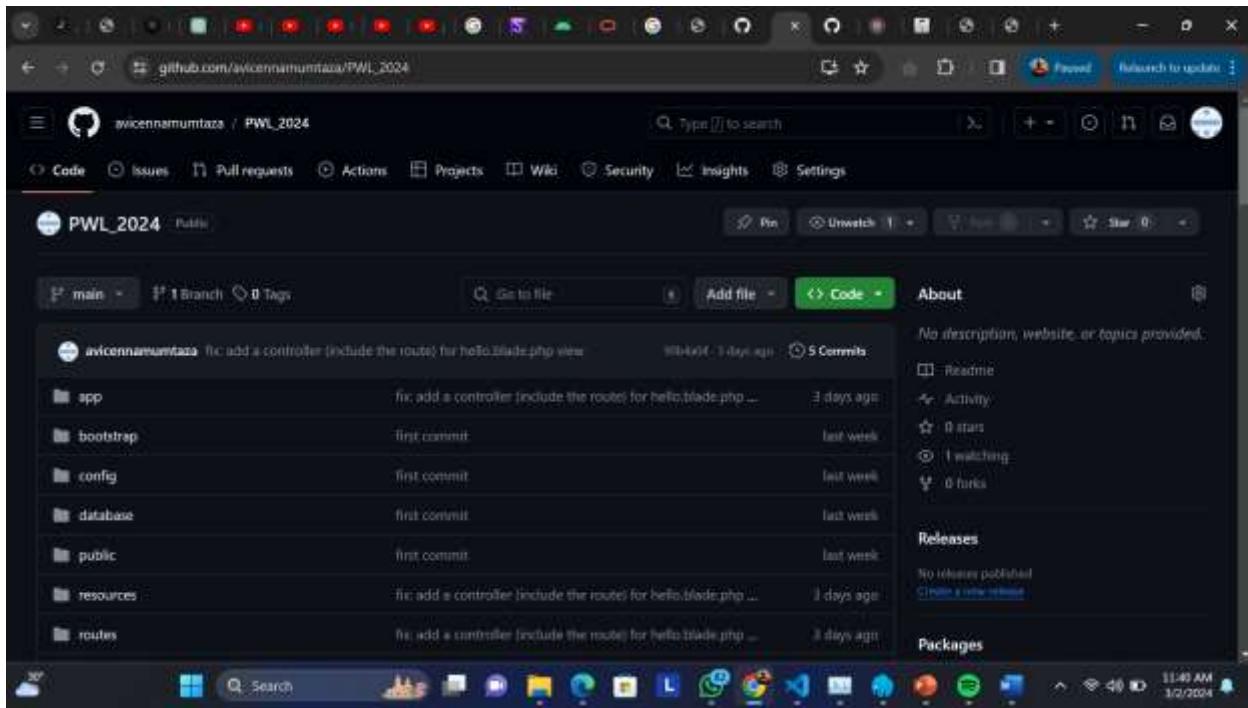
## 18. Controller mengirim nilai \$name dan \$occupation ke sebuah view untuk ditampilkan di halaman html.

The screenshot shows a Windows desktop environment. In the top-left corner, there's a browser window displaying the text "Hello, Avis" and "You are Student". Below it, a code editor window shows the contents of a view file:

```
<!-- View pada resources/views/hello.blade.php -->
1 <html>
2   <body>
3     <!-- <h1>Hello, {{ $name }}</h1> -->
4     <h1>Hello, <?php echo $name; ?></h1>
5     <!-- <h1>You are {{ $occupation }}</h1> -->
6     <h1>You are <?php echo $occupation; ?></h1>
7   </body>
8 </html>
```

Below the code editor is a terminal window showing a list of routes, identical to the one in the previous screenshot.

## 19. Sinkronisasi perubahan pada github.



===== terima kasih 😊 =====