

**Laporan Praktikum Pemrograman Web Lanjut**



Disusun oleh :

Octrian Adiluhung Tito Putra

NIM. 2341720078

Dosen Pengampu Mata Kuliah :

Dimas Wahyu Wibowo, ST., MT.

**KELAS 2-A**

**D-4 TEKNIK INFORMATIKA**

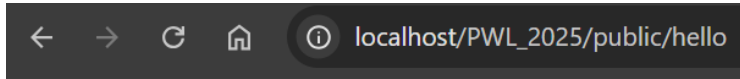
**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2025**

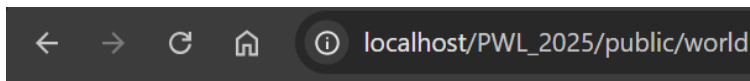
### Langkah Praktikum 1 :

- Buka browser, tuliskan URL untuk memanggil route tersebut: localhost/PWL\_2024/public/hello. Perhatikan halaman yang muncul apakah sudah sesuai dan jelaskan pengamatan Anda.



Hello World

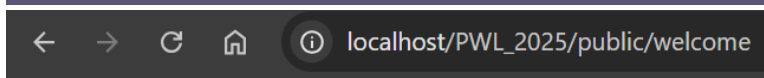
- Jika Pengguna mengakses URL pada /hello, maka fungsi pengembalian akan ditampilkan yang menampilkan pesan "Hello World" pada browser.
- Bukalah pada browser, tuliskan URL untuk memanggil route tersebut: localhost/PWL\_2024/public/world. Perhatikan halaman yang muncul apakah sudah sesuai dan jelaskan pengamatan Anda.



World

- Jika Pengguna mengakses URL pada /world, maka fungsi pengembalian pada route yang baru akan ditampilkan yang menampilkan pesan "World" pada browser.
- Selanjutnya, cobalah membuat route '/' yang menampilkan pesan 'Selamat Datang'.

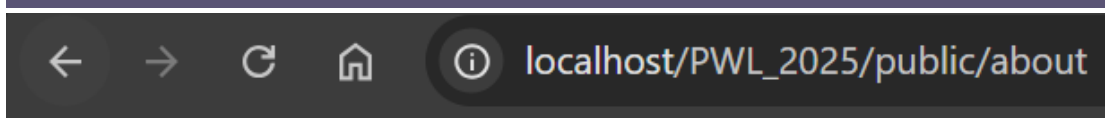
```
Route::get(uri: '/welcome', action: function (): string {  
    return 'Selamat Datang';  
});
```



Selamat Datang

- Kemudian buatlah route '/about' yang akan menampilkan NIM dan nama Anda.

```
Route::get(uri: '/about', action: function (): string {  
    return 'NIM : 2341720078 <br>Nama : Octrian Adiluhung Tito Putra';  
});
```

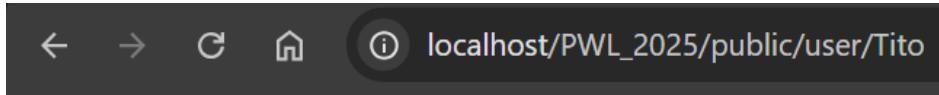


NIM : 2341720078

Nama : Octrian Adiluhung Tito Putra

## Langkah Praktikum 2 :

- Jalankan kode dengan menuliskan URL untuk memanggil route tersebut: localhost/PWL\_2024/public/user>NamaAnda. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.

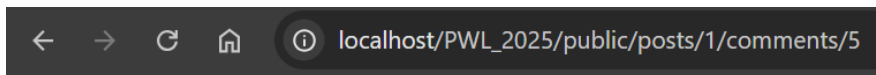


Nama saya Tito

- Nama yang dituliskan pada /public/user/nama muncul pada halaman web browser
- Selanjutnya, coba tuliskan URL: localhost/PWL\_2024/public/user/. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda



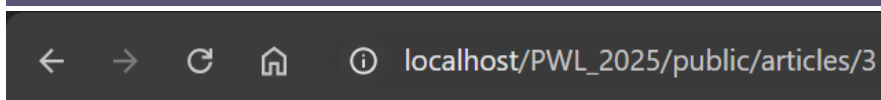
- Akan muncul page not found karena parameter pada /user tidak terisi
- Jalankan kode dengan menuliskan URL untuk memanggil route tersebut: localhost/PWL\_2024/public/posts/1/comments/5. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.



Pos ke-1 Komentar ke-: 5

- Kedua parameter route dari hasil permintaan user ditampilkan di page
- Kemudian buatlah route /articles/{id} yang akan menampilkan output “Halaman Artikel dengan ID {id}”, ganti id sesuai dengan input dari url.

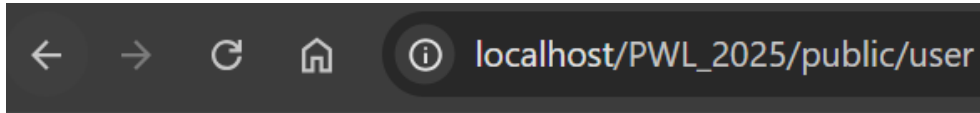
```
41 Route::get(uri: '/articles/{articleId}', action: function  
42 ($articleId): string {  
43     return 'Halaman artikel dengan ID ' . $articleId;  
44 });
```



Halaman artikel dengan ID 3

### Langkah Praktikum 3 :

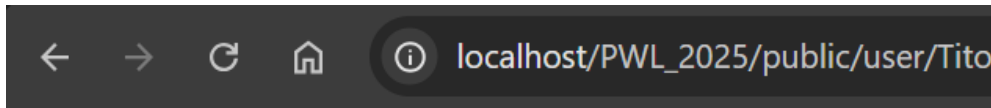
- Jalankan kode dengan menuliskan URL: localhost/PWL\_2024/public/user/. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.



Nama saya

- Nama tidak ditampilkan karena bernilai default null

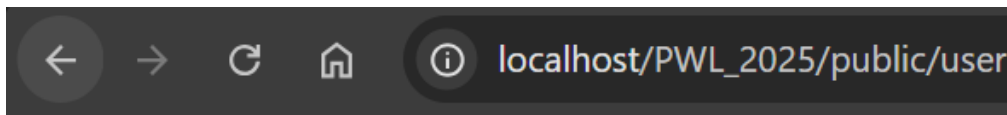
- Selanjutnya tuliskan URL: localhost/PWL\_2024/public/user>NamaAnda. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda



Nama saya Tito

- Nama ditampilkan karena parameter opsional terisi dan tidak bernilai null (default)

- Jalankan kode dengan menuliskan URL: localhost/PWL\_2024/public/user/. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.

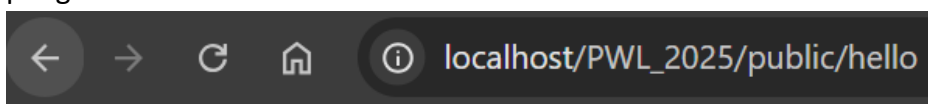


Nama saya John

- Nama yang ditampilkan John sebagai default dari parameter \$name

### Langkah Praktikum 4 :

- Buka browser, tuliskan URL untuk memanggil route tersebut: localhost/PWL\_2024/public/hello. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.



Hello World

- Muncul action dari class WelcomeController yang diakses menggunakan route /hello

- Modifikasi hasil pada praktikum poin 2 (Routing) dengan konsep controller. Pindahkan logika eksekusi ke dalam controller dengan nama PageController.

```
class PageController extends Controller
{
    1 reference | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | Genera
    public function index(): string {
        return 'Selamat Datang';
    }
    1 reference | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | Genera
    public function about(): string {
        return 'Nama : Octrian Adiluhung Tito Putra <br> NIM : 2341720078';
    }
    1 reference | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | Genera
    public function articles($id): string {
        return 'Halaman Artikel dengan ID ' . $id;
    }
}

Route::get(uri: '/hello', action: [WelcomeController::class, 'hello']);
Route::get(uri: '/', action: [PageController::class, 'index']);
Route::get(uri: '/about', action: [PageController::class, 'about']);
Route::get(uri: '/articles/{id}', action: [PageController::class, 'articles']);
```

localhost/pemrograman\_web\_lanjut/Minggu2/PWL\_2025/public/

Selamat Datang

localhost/pemrograman\_web\_lanjut/Minggu2/PWL\_2025/public/about

Nama : Octrian Adiluhung Tito Putra  
NIM : 2341720078

localhost/pemrograman\_web\_lanjut/Minggu2/PWL\_2025/public/articles/3

Halaman Artikel dengan ID 3

- Modifikasi kembali implementasi sebelumnya dengan konsep Single Action Controller. Sehingga untuk hasil akhir yang didapatkan akan ada HomeController, AboutController dan ArticleController. Modifikasi juga route yang digunakan.

```

2 references | 0 implementations | Codeium: Refactor | Explain
class HomeController extends Controller
{
    1 reference | 0 overrides | Tabnine | Edit | Test | Explain
    public function index(): string {
        return 'Selamat Datang';
    }
}

2 references | 0 implementations | Codeium: Refactor | Explain
class AboutController extends Controller
{
    1 reference | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | Generate
    public function about(): string {
        return 'Nama : Octrian Adiluhung Tito Putra <br> NIM : 2341720078';
    }
}

class ArticlesController extends Controller
{
    1 reference | 0 overrides | Tabnine | Edit | Test | Explain | Document |
    public function articles($id): string {
        return 'Halaman Artikel dengan ID ' . $id;
    }
}

Route::get(uri: '/hello', action: [WelcomeController::class,'hello']);
Route::get(uri: '/', action: [HomeController::class,'index']);
Route::get(uri: '/about', action: [AboutController::class,'about']);
Route::get(uri: '/articles/{id}', action: [ArticlesController::class, 'articles']);

```

### Langkah Praktikum 5 :

Membuat Controller yang berisi method method standar CRUD

```

3 references | 0 implementations | Codeium: Refactor | Explain
class PhotoController extends Controller
{
    /**
     * Display a listing of the resource.
     */
    0 references | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | X
    public function index(): void
    {
        //
    }

    /**
     * Show the form for creating a new resource.
     */
    0 references | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | X
    public function create(): void
    {
        //
    }

    /**
     * Store a newly created resource in storage.
     */
    0 references | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | X
    public function store(Request $request): void
    {
        //
    }

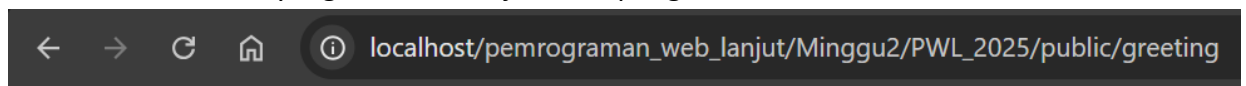
    /**
     * Display the specified resource.
     */
    0 references | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | X
    public function show(string $id): void
    {
        //
    }

    /**
     * Show the form for editing the specified resource.
     */
    0 references | 0 overrides | Tabnine | Edit | Test | Explain | Document | Codeium: Refactor | Explain | X
    public function edit(string $id): void
    {
        //
    }
}

```

## Langkah Praktikum 6 :

- Jalankan code dengan membuka url localhost/PWL\_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.

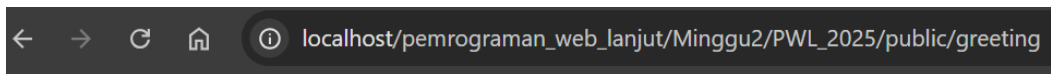


# Hello, Tito

- View akan ditampilkan melalui route URL yang sudah ditentukan

### Langkah Praktikum 7 :

- Jalankan code dengan membuka url localhost/PWL\_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.

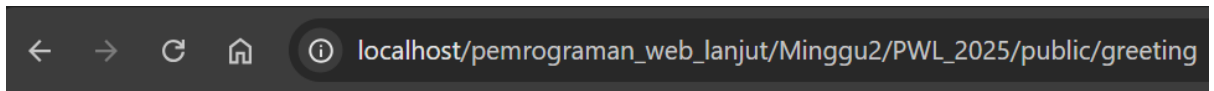


**Hello, Tito**

- Output nya tetap sama, tetapi file hello.blade.php dipanggil dari folder views/blog melalui routing

### Langkah Praktikum 8 :

- Jalankan code dengan membuka url localhost/PWL\_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.

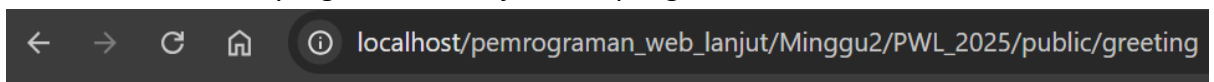


**Hello, Tito**

- Views dipanggil oleh controller, lalu routing akan memanggil controller untuk bisa me return view pada browser

### Langkah Praktikum 9 :

- Jalankan code dengan membuka url localhost/PWL\_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.



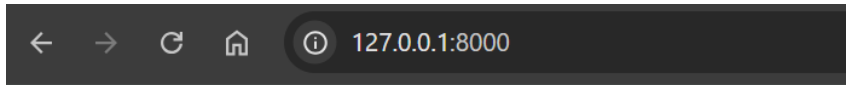
**Hello, Tito**

**You are Astronaut**



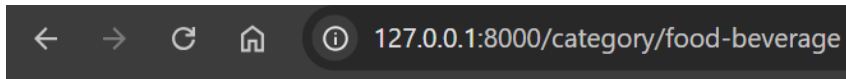
## OUTPUT TUGAS PRAKTIKUM

### 1. Halaman Home



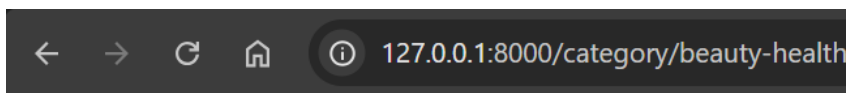
# Selamat Datang di Aplikasi POS

### 2. Halaman Products



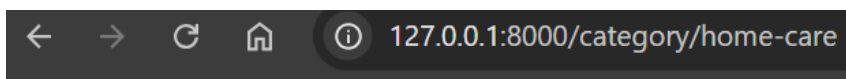
## Kategori Food & Beverage

Daftar produk makanan dan minuman



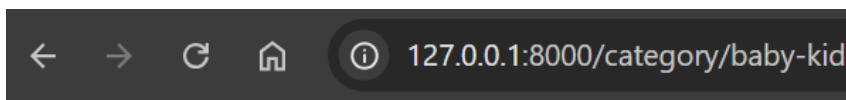
## Kategori Beauty & Health

Produk kecantikan dan kesehatan



## Kategori Home Care

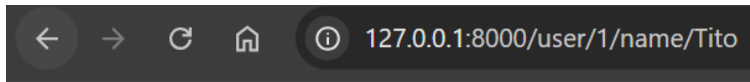
Produk perawatan rumah



## Kategori Baby & Kid

Produk bayi dan anak

3. Halaman User

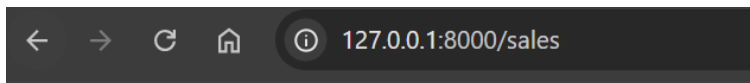


## Profil Pengguna

ID: 1

Nama: Tito

4. Halaman Penjualan



## Halaman Transaksi Penjualan