

# **LAPORAN PRAKTIKUM Pemrograman Web Lanjut**

## **PRAKTIKUM 9**

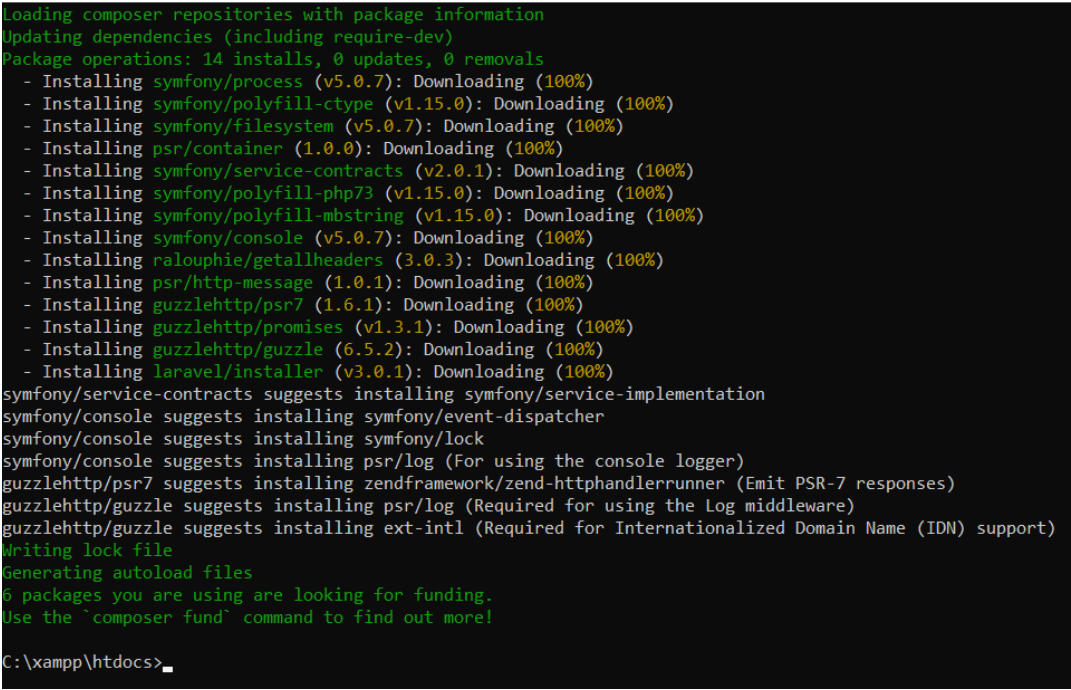


**Oleh :**

**MARTIN AMANU KHUSNA / NIM : 1841720138  
KELAS : TI-2B**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG  
2020**

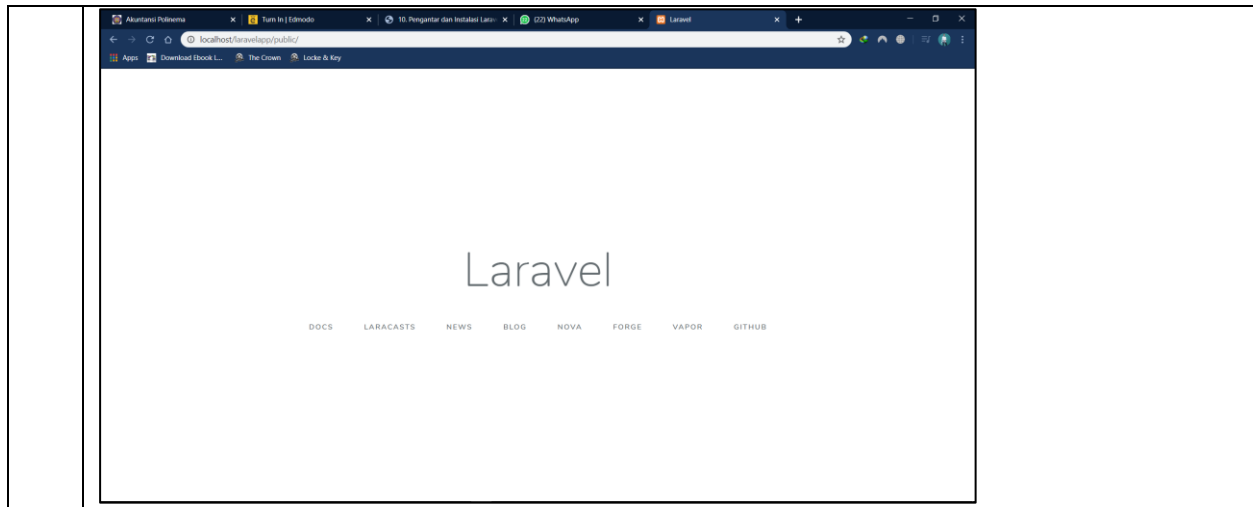
## Praktikum – Bagian 1: Instalasi Laravel di Windows

NO	STEP
1.	<p data-bbox="280 323 448 348">Command Prompt</p>  <pre data-bbox="280 352 1344 1037">Loading composer repositories with package information Updating dependencies (including require-dev) Package operations: 14 installs, 0 updates, 0 removals - Installing symfony/process (v5.0.7): Downloading (100%) - Installing symfony/polyfill-ctype (v1.15.0): Downloading (100%) - Installing symfony/filesystem (v5.0.7): Downloading (100%) - Installing psr/container (1.0.0): Downloading (100%) - Installing symfony/service-contracts (v2.0.1): Downloading (100%) - Installing symfony/polyfill-php73 (v1.15.0): Downloading (100%) - Installing symfony/polyfill-mbstring (v1.15.0): Downloading (100%) - Installing symfony/console (v5.0.7): Downloading (100%) - Installing ralouphie/getallheaders (3.0.3): Downloading (100%) - Installing psr/http-message (1.0.1): Downloading (100%) - Installing guzzlehttp/psr7 (1.6.1): Downloading (100%) - Installing guzzlehttp/promises (v1.3.1): Downloading (100%) - Installing guzzlehttp/guzzle (6.5.2): Downloading (100%) - Installing laravel/installer (v3.0.1): Downloading (100%) symfony/service-contracts suggests installing symfony/service-implementation symfony/console suggests installing symfony/event-dispatcher symfony/console suggests installing symfony/lock symfony/console suggests installing psr/log (For using the console logger) guzzlehttp/psr7 suggests installing zendframework/zend-httphandler (Emit PSR-7 responses) guzzlehttp/guzzle suggests installing psr/log (Required for using the Log middleware) guzzlehttp/guzzle suggests installing ext-intl (Required for Internationalized Domain Name (IDN) support) Writing lock file Generating autoload files 6 packages you are using are looking for funding. Use the `composer fund` command to find out more!  C:\xampp\htdocs&gt;</pre>
2.	<p data-bbox="280 1052 1360 1121">Tunggulah hingga proses penginstalan selesai. Setelah itu, Kita bisa mulai membuat project laravel dengan mengetikkan perintah: <b>laravel new laravelapp</b></p>

Command Prompt

```
C:\xampp\htdocs>laravel new laravelapp
Crafting application...
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
Package operations: 94 installs, 0 updates, 0 removals
- Installing doctrine/inflector (1.3.1): Downloading (100%)
- Installing doctrine/lexer (1.2.0): Downloading (100%)
- Installing dragonmantank/cron-expression (v2.3.0): Downloading (100%)
- Installing voku/portable-ascii (1.4.10): Downloading (100%)
- Installing symfony/polyfill-ctype (v1.15.0): Loading from cache
- Installing phpoption/phpoption (1.7.3): Downloading (100%)
- Installing vlucas/phpdotenv (v4.1.3): Downloading (100%)
- Installing symfony/css-selector (v5.0.7): Downloading (100%)
- Installing tijsverkoyen/css-to-inline-styles (2.2.2): Downloading (connecting..Downloading (100%)
- Installing symfony/polyfill-mbstring (v1.15.0): Loading from cache
- Installing symfony/var-dumper (v5.0.7): Downloading (100%)
- Installing symfony/routing (v5.0.7): Downloading (100%)
- Installing symfony/process (v5.0.7): Loading from cache
- Installing symfony/polyfill-php72 (v1.15.0): Downloading (100%)
- Installing symfony/polyfill-intl-idn (v1.15.0): Downloading (100%)
- Installing symfony/mime (v5.0.7): Downloading (100%)
- Installing symfony/polyfill-php73 (v1.15.0): Loading from cache
- Installing symfony/http-foundation (v5.0.7): Downloading (100%)
- Installing psr/event-dispatcher (1.0.0): Downloading (100%)
- Installing symfony/event-dispatcher-contracts (v2.0.1): Downloading (connectingDownloading (100%)
- Installing symfony/event-dispatcher (v5.0.7): Downloading (100%)
- Installing psr/log (1.1.3): Downloading (100%)
- Installing symfony/error-handler (v5.0.7): Downloading (100%)
- Installing symfony/http-kernel (v5.0.7): Downloading (100%)
- Installing symfony/finder (v5.0.7): Downloading (100%)
- Installing psr/container (1.0.0): Loading from cache
- Installing symfony/service-contracts (v2.0.1): Loading from cache
- Installing symfony/console (v5.0.7): Loading from cache
- Installing symfony/polyfill-iconv (v1.15.0): Downloading (100%)
- Installing egulias/email-validator (2.1.17): Downloading (100%)
- Installing swiftmailer/swiftmailer (v6.2.3): Downloading (100%)
- Installing ramsey/collection (1.0.1): Downloading (100%)
- Installing brick/math (0.8.14): Downloading (100%)
- Installing ramsey/uuid (4.0.1): Downloading (100%)
- Installing psr/simple-cache (1.0.1): Downloading (100%)
- Installing opis/closure (3.5.1): Downloading (100%)
- Installing symfony/translation-contracts (v2.0.1): Downloading (100%)
- Installing symfony/translation (v5.0.7): Downloading (100%)
- Installing nesbot/carbon (2.32.2): Downloading (100%)
- Installing monolog/monolog (2.0.2): Downloading (100%)
- Installing league/flysystem (1.0.66): Downloading (100%)
- Installing league/commonmark (1.3.3): Downloading (100%)
- Installing laravel/framework (v7.4.0): Downloading (100%)
- Installing fideloper/proxy (4.3.0): Downloading (100%)
- Installing asm89/stack-cors (1.3.0): Downloading (100%)
- Installing fruitcake/laravel-cors (v1.0.5): Downloading (100%)
```

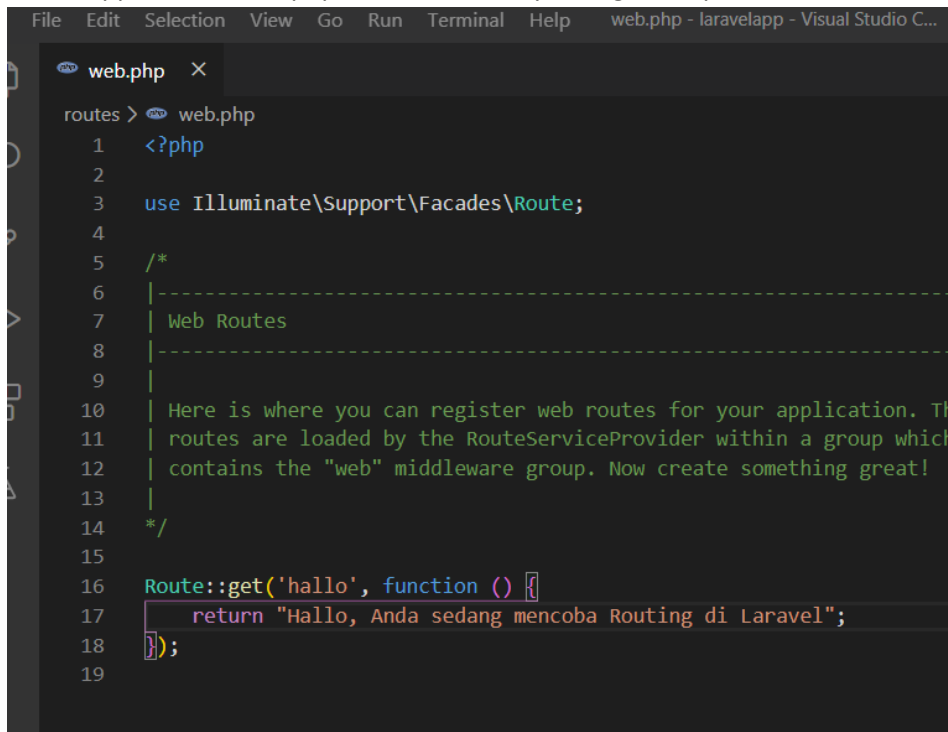
	<div data-bbox="280 195 1248 1318">  Command Prompt <pre> league/flysystem suggests installing league/flysystem-ziparchive (Allows you to use ZipArchive adapter) league/flysystem suggests installing spatie/flysystem-dropbox (Allows you to use Dropbox storage) league/flysystem suggests installing srmlive/flysystem-dropbox-v2 (Allows you to use Dropbox storage) laravel/framework suggests installing aws/aws-sdk-php (Required to use the SQS queue driver, DynamoDb) laravel/framework suggests installing doctrine/dbal (Required to rename columns and drop SQLite column) laravel/framework suggests installing ext-memcached (Required to use the memcache cache driver.) laravel/framework suggests installing ext-pcntl (Required to use all features of the queue worker.) laravel/framework suggests installing ext-posix (Required to use all features of the queue worker.) laravel/framework suggests installing ext-redis (Required to use the Redis cache and queue drivers (^4 laravel/framework suggests installing league/flysystem-aws-s3-v3 (Required to use the Flysystem S3 dri laravel/framework suggests installing league/flysystem-cached-adapter (Required to use the Flysystem c laravel/framework suggests installing league/flysystem-sftp (Required to use the Flysystem SFTP driver laravel/framework suggests installing moontoast/math (Required to use ordered UUIDs (^1.1).) laravel/framework suggests installing nyholm/psr7 (Required to use PSR-7 bridging features (^1.2).) laravel/framework suggests installing pda/pheanstalk (Required to use the beanstalk queue driver (^4.0 laravel/framework suggests installing pusher/pusher-php-server (Required to use the Pusher broadcast d laravel/framework suggests installing symfony/cache (Required to PSR-6 cache bridge (^5.0).) laravel/framework suggests installing symfony/psr-http-message-bridge (Required to use PSR-7 bridging laravel/framework suggests installing wilddbit/swiftmailer-postmark (Required to use Postmark mail driv guzzlehttp/guzzle suggests installing zendframework/zend-httpdierrunner (Emit PSR-7 responses) guzzlehttp/guzzle suggests installing ext-intl (Required for Internationalized Domain Name (IDN) suppo psy/psysh suggests installing ext-pcntl (Enabling the PCNTL extension makes PsySH a lot happier :)) psy/psysh suggests installing ext-pdo-sqlite (The doc command requires SQLite to work.) psy/psysh suggests installing ext-posix (If you have PCNTL, you'll want the POSIX extension as well.) psy/psysh suggests installing hoa/console (A pure PHP readline implementation. You'll want this if you filp/whoops suggests installing whoops/soap (Formats errors as SOAP responses) facade/ignition suggests installing laravel/telescope (^3.1) sebastian/global-state suggests installing ext-uopz (*) sebastian/environment suggests installing ext-posix (*) phpunit/php-code-coverage suggests installing ext-xdebug (^2.7.2) phpunit/phpunit suggests installing ext-soap (*) phpunit/phpunit suggests installing ext-xdebug (*) phpunit/phpunit suggests installing phpunit/php-invoker (^2.0.0) Package jakub-ondierka/php-console-color is abandoned, you should avoid using it. Use php-parallel-lint Package jakub-ondierka/php-console-highlighter is abandoned, you should avoid using it. Use php-parallel Generating optimized autoload files 29 packages you are using are looking for funding. Use the `composer fund` command to find out more! &gt; @php -r "file_exists('.env')    copy('.env.example', '.env');"; &gt; @php artisan key:generate --ansi Application key set successfully. &gt; illuminate\Foundation\ComposerScripts::postAutoloadDump &gt; @php artisan package:discover --ansi Discovered Package: facade/ignition Discovered Package: fideloper/proxy Discovered Package: fruitcake/laravel-cors Discovered Package: laravel/tinker Discovered Package: nesbot/carbon Discovered Package: nunomaduro/collision Package manifest generated successfully. Application ready! Build something amazing. </pre> </div>
3.	<p>Setelah proses instalasi Laravel selesai, Kita perlu menguji apakah hasil instalasi tersebut berjalan dengan baik atau tidak. Aktifkan Apache server lewat XAMPP Control Panel. Pada browser, ketik alamat <a href="http://localhost/laravelapp/public/">http://localhost/laravelapp/public/</a> Atau Kita dapat menggunakan perintah berikut di Command Prompt: <b>php artisan serve</b></p>



## Praktikum – Bagian 2: Mengenal dan membuat route

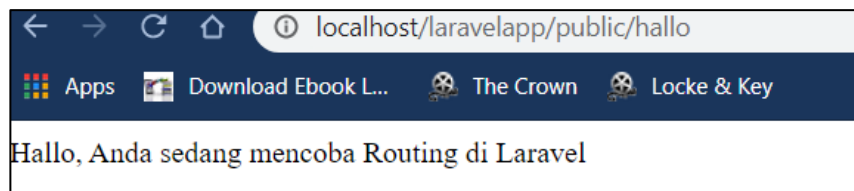
NO	STEP
1.	<p>Setiap request yang datang pada laravel akan diarahkan melalui sebuah route. Route ini yang akan menentukan respon apa yang akan dikerjakan untuk membalas request tersebut. Halaman homepage default Laravel seperti yang ditampilkan pada gambar di Bagian 1 adalah hasil dari script yang terletak pada file laravelapp\routes\web.php Berikut ini adalah perintah route untuk menampilkan halaman default Laravel tersebut.</p> <pre> web.php routes &gt; web.php 1  &lt;?php 2 3  use Illuminate\Support\Facades\Route; 4 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 16 Route::get('/', function () { 17     return view('welcome'); 18 }); </pre>

2. Kita dapat menuliskan perintah baru untuk membuat route, selain mengembalikan file view sebagai response, Kita dapat mengembalikan response berupa string atau teks biasa. Buka file `laravelapp\routes\web.php`, tambahkan isinya dengan script di bawah ini :



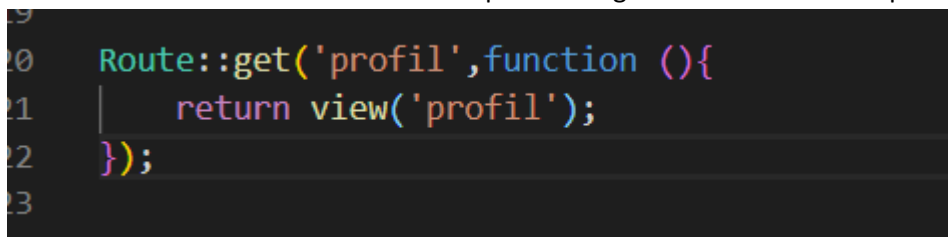
```
File Edit Selection View Go Run Terminal Help web.php - laravelapp - Visual Studio C...
web.php x
routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 /*
6 |-----
7 | Web Routes
8 |-----
9 |
10 | Here is where you can register web routes for your application. The
11 | routes are loaded by the RouteServiceProvider within a group which
12 | contains the "web" middleware group. Now create something great!
13 |
14 */
15
16 Route::get('hallo', function () {
17     return "Hallo, Anda sedang mencoba Routing di Laravel";
18 });
19
```

Hasil :




Saya masih menggunakan <http://localhost/laravelapp/public/hallo> dan belum bisa menggunakan <http://localhost:8000/hallo>

3. Selanjutnya, kita akan mencoba membuat route baru dengan menampilkan halaman view. Tambahkan route baru untuk halaman “profil” dengan menambahkan script berikut:



```
19
20 Route::get('profil',function (){
21     return view('profil');
22 });
23
```

Buat file baru bernama `profil.blade.php` di dalam folder `laravelapp\resources\views\`, ketikkan script berikut ini:

	<pre>resources &gt; views &gt; profil.blade.php 1  &lt;!DOCTYPE html&gt; 2  &lt;html&gt; 3  &lt;head&gt; 4         &lt;title&gt;Profil Saya&lt;/title&gt; 5  &lt;/head&gt; 6  &lt;body&gt; 7         &lt;h1&gt;Profil Saya&lt;/h1&gt; 8         &lt;p&gt;Perkenalkan Nama saya Martin Amanu Khusna, saya seorang mahasis 9  &lt;/body&gt; 10 &lt;/html&gt;</pre> <p>Hasil :</p> 
--	--

### **Praktikum – Bagian 3: Cara Menggunakan Controller pada Framework Laravel**

NO	STEP
1.	<p>Pada pembahasan sebelumnya, Kita sudah bisa menampilkan file view langsung dari Routenya. Tetapi sebuah aplikasi tidaklah sesimpel itu, akan ada logika dan data-data yang harus diolah terlebih dahulu. Caranya adalah dengan menggunakan Controller, seperti yang telah dipelajari ketika menggunakan CodeIgniter.</p> <p>Ada dua cara yang dapat digunakan untuk membuat Controller. Cara pertama, dengan membuat file controller secara manual dan tuliskan code untuk extends controller secara manual. Cara kedua, adalah dengan membuat file Controller lewat Artisan di Laravel.</p> <p>Kita gunakan cara yang kedua, pada latihan kali ini kita akan membuat Controller dengan nama CobaController dengan menuliskan di command prompt / terminal :</p> <pre>C:\xampp\htdocs\laravelapp&gt;php artisan make:controller CobaController Controller created successfully.</pre> <p>Maka akan terbentuk sebuah file dengan nama CobaController.php pada folder laravelapp\app\Http\Controllers\</p>

```

app > Http > Controllers > CobaController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class CobaController extends Controller
8  {
9      //
10 }
11

```

2. Ubah route halaman 'profil' pada praktikum sebelumnya menjadi seperti berikut :  
**Route::get('profil', 'CobaController@profil');**

```

9
10 Route::get('profil', 'CobaController@profil');
11

```

Langkah selanjutnya, tambahkan fungsi / method profil() pada class CobaController seperti berikut ini:

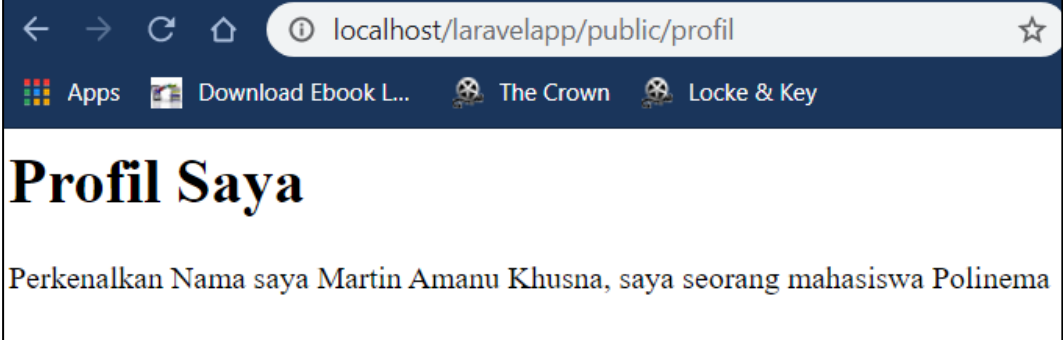
```

app > Http > Controllers > CobaController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class CobaController extends Controller
8  {
9      public function profil(){
10         return view('profil');
11     }
12 }

```

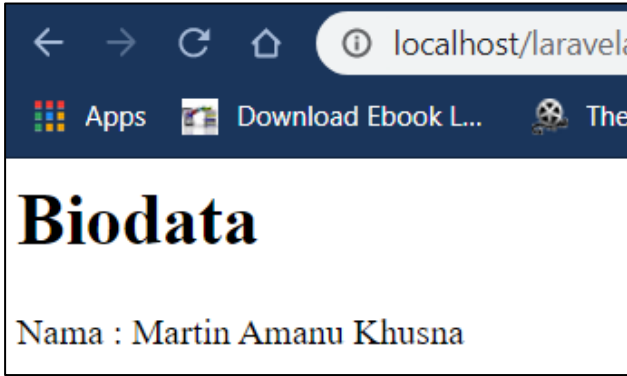
Hasil :



	
--	--

#### Praktikum – Bagian 4: Memberikan Data Controller kepada View

NO	STEP
1.	<p>Dalam sebuah aplikasi, sangat jarang Kita temukan isi dari file view yang bersifat statis. Pada umumnya, view dipakai untuk menampilkan data, contohnya adalah data yang berasal dari database. Namun, karena kita belum sampai pada pembahasan database, maka pada latihan kali ini kita akan mencoba menampilkan data dari variable</p> <p>Buat route baru untuk halaman biodata</p> <pre>Route::get('biodata','BiodataController@index');</pre> <p>Buat controller baru dengan nama BiodataController.php</p> <pre>C:\xampp\htdocs\laravelapp&gt;php artisan make:controller BiodataController Controller created successfully.</pre> <p>Ketikkan script berikut dengan memberikan method / fungsi index pada laravelapp\app\Http\Controllers\BiodataController.php</p> <pre>app &gt; Http &gt; Controllers &gt; BiodataController.php 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6 7  class BiodataController extends Controller 8  { 9      public function index(){ 10         \$nama = 'Martin Amanu Khusna'; 11         return view('biodata' , ['nama' =&gt; \$nama]); 12     } 13 } 14</pre>

2.	<p>Buat file view dengan nama biodata.blade.php pada laravelapp\resources\views\</p> <pre> resources &gt; views &gt; biodata.blade.php 1  &lt;!DOCTYPE html&gt; 2  &lt;html&gt; 3  &lt;head&gt; 4         &lt;title&gt;Biodata&lt;/title&gt; 5  &lt;/head&gt; 6  &lt;body&gt; 7         &lt;h1&gt;Biodata&lt;/h1&gt; 8         &lt;p&gt;Nama : {{ \$nama }}&lt;/p&gt; 9  &lt;/body&gt; 10 &lt;/html&gt; </pre> <p>Hasil :</p> 
----	--

### Praktikum – Bagian 5: Memberikan Data Array Kepada View

NO	STEP
1.	<p>Pada bagian ini, Kita coba untuk memberikan data array kepada view. Tambahkan variabel data array pada <b>BiodataController.php</b></p> <pre> app &gt; Http &gt; Controllers &gt; BiodataController.php 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6 7  class BiodataController extends Controller 8  { 9      public function index(){ 10         \$nama = 'Martin Amanu Khusna'; 11         \$materi = ["Web Design", "Web Programming", "Digital Marketing", "Graphic Design"]; 12         return view('biodata', ['nama' =&gt; \$nama, 'materi' =&gt; \$materi]); 13     } 14 } </pre>

2. Tambahkan isi file view biodata.blade.php menjadi seperti berikut:

```
resources > views > biodata.blade.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Biodata</title>
5  </head>
6  <body>
7      <h1>Biodata</h1>
8      <p>Nama : {{ $nama }}</p>
9      <p>Materi Mengajar</p>
10     <ul>
11         @foreach($materi as $datamateri)
12             <li>{{ $datamateri }}</li>
13         @endforeach
14     </ul>
15 </body>
16 </html>
```

Hasil :

