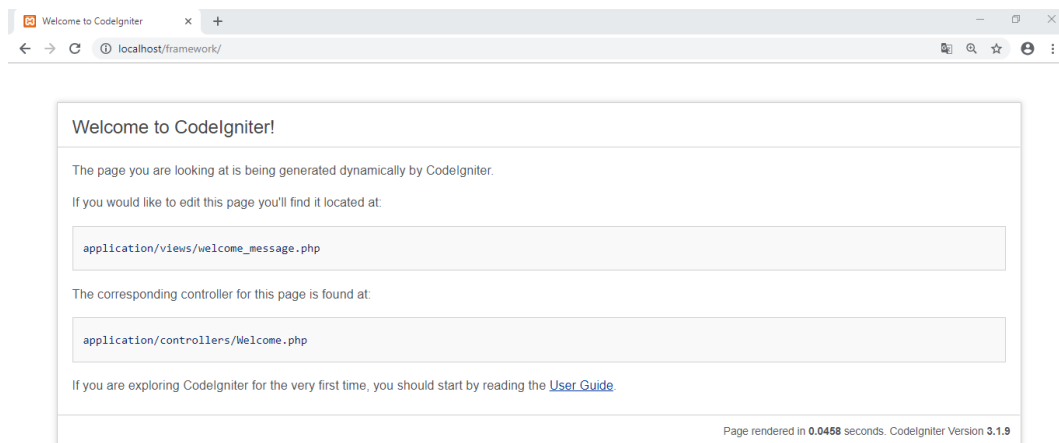


INSTALASI

1. Download codeigniter di www.codeigniter.com
2. Ekstrak hasil download ke localhost (C:/XAMPP/htdocs)
3. Ubah nama folder nya dari Codeigniter 3.1.9 (tergantung versi yg kamu download) menjadi nama project yang mau di buat (framework).

Disarankan membuat nama dalam huruf kecil semua

4. Jalankan apache web server pada XAMPP.
5. Jalankan pada browser “http://localhost/nama_folder_project”
(<http://localhost/framework>)



STRUKTUR DIREKTORI


Terdapat dua direktori penting di dalam CI: 📁 application dan 📁 system.

Selain itu terdapat juga direktori 📁 user_guide dan beberapa file. Berikut ini penjelasannya:
















1. **application** berisi semua kode aplikasi. Di dalam direktori inilah kita akan menulis semua kode aplikasi kita.

2. **system** berisi kode-kode inti dari Codeiniter. Jangan mengubah apapun di dalam direktori ini. Jika kita ingin upgrade versi, kita cukup me-replace direktori ini dengan yang baru.
3. **tests** berisi kode untuk melakukan unit testing.
4. **user_guide** berisi dokumentasi codeigniter. Kita bisa menghapus direktori ini saat web sudah jadi.
5. **.editor_config** berisi konfigurasi untuk teks editor.
6. **.gitignore** berisi daftar file dan folder yang akan diabaikan oleh Git.
7. **comspoer.json** adalah file yang berisi keterangan project dan keterangan library yang digunakan. File ini dibutuhkan oleh composer.
8. **contributing.md** adalah file yang berisi penjelasan cara berkontribusi di proyek CI. Kita bisa menghapus file ini, apabila web sudah jadi.
9. **license.txt** adalah file yang berisi keterangan lisensi dari CI.
10. **readme.rst** sama seperti file contributing.md file ini berisi penjelasan dan informasi tentang project CI. Kita juga bisa menghapus file ini saat web sudah selesai.
11. **index.php** adalah file utama dari CI. File yang akan dibuka pertamakali saat kita mengakses web.

Selanjutnya silahkan buka direktori  application dan perhatikan direktori yang ada di sana.


 cache berisi cache dari aplikasi.


 config berisi konfigurasi aplikasi.


-  autoload.php tempat kita mendefinisikan autoload;
-  config.php konfigurasi aplikasi;
-  constants.php berisi konstanta;
-  database.php konfigurasi database aplikasi;
-  doctypes.php berisi definisi untuk doctype HTML;
-  foreign_chars.php berisi karakter dan simbol;
-  hooks.php berisi konfigurasi hooks;
-  index.html untuk mencegah direct access;
-  memcached.php untuk berisi konfigurasi untuk memcached;
-  migration.php konfigurasi untuk migrasi;
-  mimes.php berisi definisi tipe file;
-  profiler.php konfigurasi untuk profiler;
-  routers.php tempat kita menulis route aplikasi;
-  smileys.php berisi kode untuk emoji;
-  user_agents.php berisi definisi untuk user agents.

 controller berisi kode untuk controller.

 core berisi kode untuk custom core.

 helpers berisi fungsi-fungsi helper.

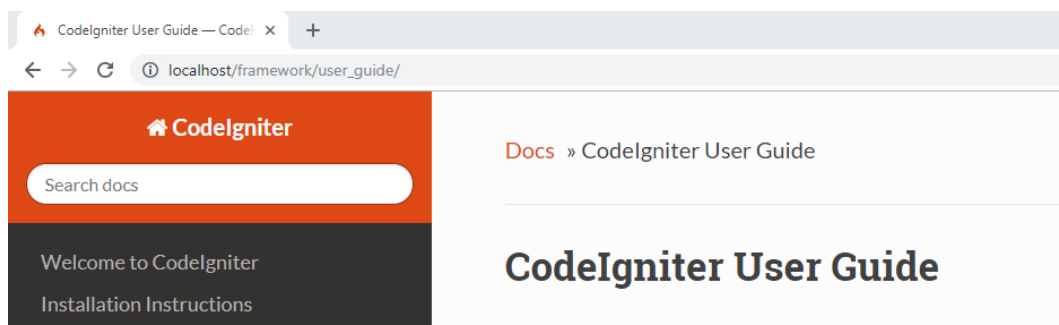
 hooks berisi kode untuk script hook.

 language berisi string untuk bahasa, apabila web mendukung multibahasa.

- 📁 libraries berisi library.
- 📁 logs berisi logs dari aplikasi.
- 📁 models berisi kode untuk model.
- 📁 thrid_party berisi library dari pihak ketiga.
- 📁 views berisi kode untuk view.
- 📄 index.html file html untuk mencegah direct access.

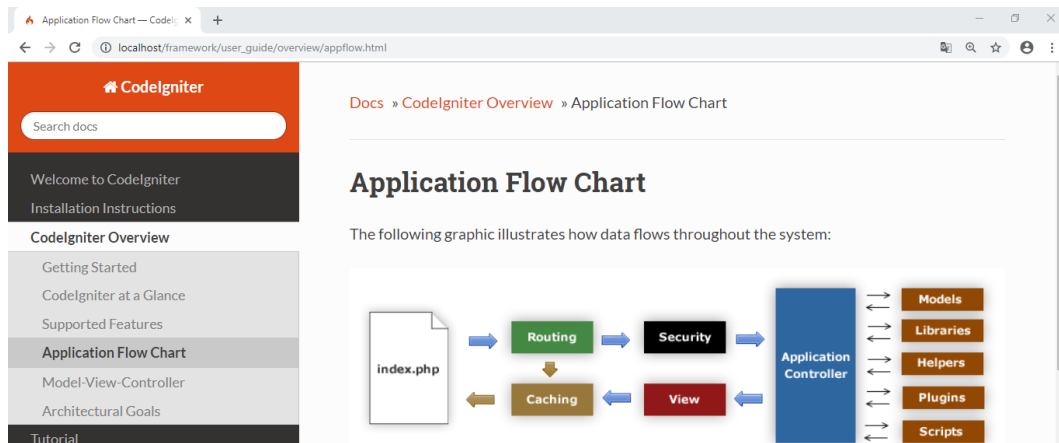
USER GUIDE

Secara default Codeigniter menyediakan panduan cara penggunaan codeigniter pada folder codeigniter yang anda download. Yaitu terdapat pada folder user_guide dan untuk mempelajari panduan dari penggunaan codeigniter anda dapat langsung menjalankannya pada browser dengan alamat http://localhost/framework/user_guide/.



APP FLOW CI

Untuk melihat alur kerja codeigniter kita dapat mengaksesnya pada url berikut: http://localhost/framework/user_guide/overview/appflow.html

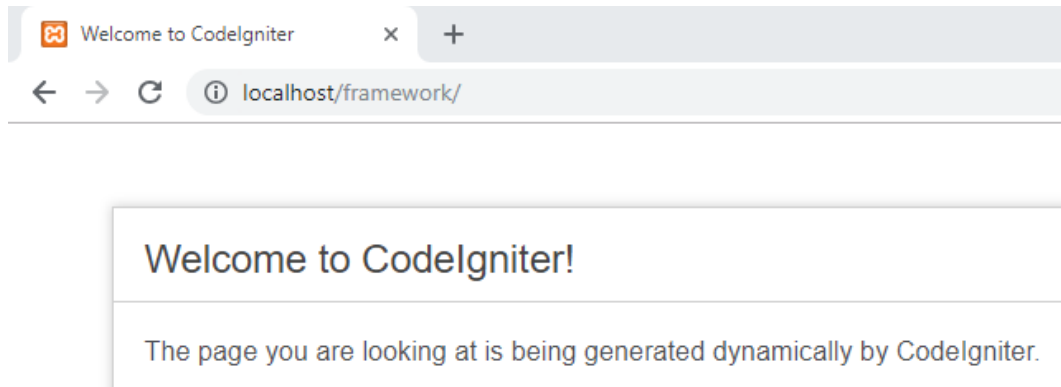


Alur kerjanya seperti ini:

1. User mengirim request ke web;
2. File yang pertama kali dieksekusi adalah index.php;
3. Lalu dari index.php, request akan diteruskan oleh routers.php;
4. routers.php akan mencari cache di server, apabila terdapat cache maka cache itu yang akan dikirim sebagai balasan (response). Apabila tidak ada cache barulah request diteruskan ke Controller;
5. Controller akan bertanggung jawab untuk mengambil data dari Model dan me-rendernya ke dalam View dengan menggunakan library, plugin, dan helper yang ada.
6. Hasil render (view) dikirim ke pengguna dan disimpan dalam cache, apabila fitur cache aktif;
7. Selesai.

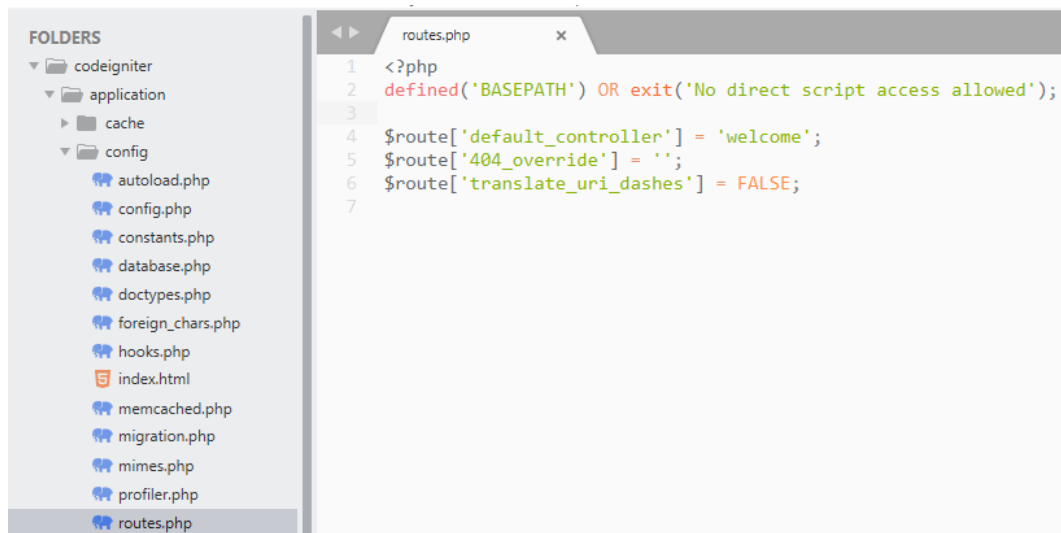
HALAMAN DEFAULT

Halaman default yang tampil ketika kita telah selesai menginstal CI yaitu sebagai berikut:



Alur kerjanya adalah sebagai berikut:


1. Diawali dari file pengaturan routes.php yang terletak di application/config/routes.php



- a. `defined('BASEPATH') OR exit('No direct script access allowed');`, syntax di atas berfungsi untuk mencegah akses langsung pada file controller dan harus diakses melalui www.namaweb.com/controller.

- b. Pengaturan routes codeigniter di atas, pengaturan default_controller di setting controller “welcome “,
- c. Pengaturan untuk menangani halaman 404 atau halaman yang di tampilkan jika tidak di temukannya data ada url. anda dapat mengatur halaman 404 anda dengan cara memasukkan controller yang ingin anda jadikan untuk menetapkan halaman 404 pada aplikasi anda.
- d. Pengaturan `$route['translate_uri_dashes']=false` adalah pengaturan untuk menetapkan nilai true atau false untuk izin penggunaan tanda “-” (dash) pada controller di url pada saat di jalankan.

2. File routes.php memanggil controller default yaitu welcome yang terletak di application/controllers/welcome.php



The screenshot shows a code editor with two tabs: 'routes.php' and 'Welcome.php'. The 'EXPLORER' sidebar on the left shows the project structure: FRAMEWORK, application, cache, config, controllers, index.html, Welcome.php, core, and helpers. The 'Welcome.php' file is selected and its content is displayed in the editor. The code is as follows:

```

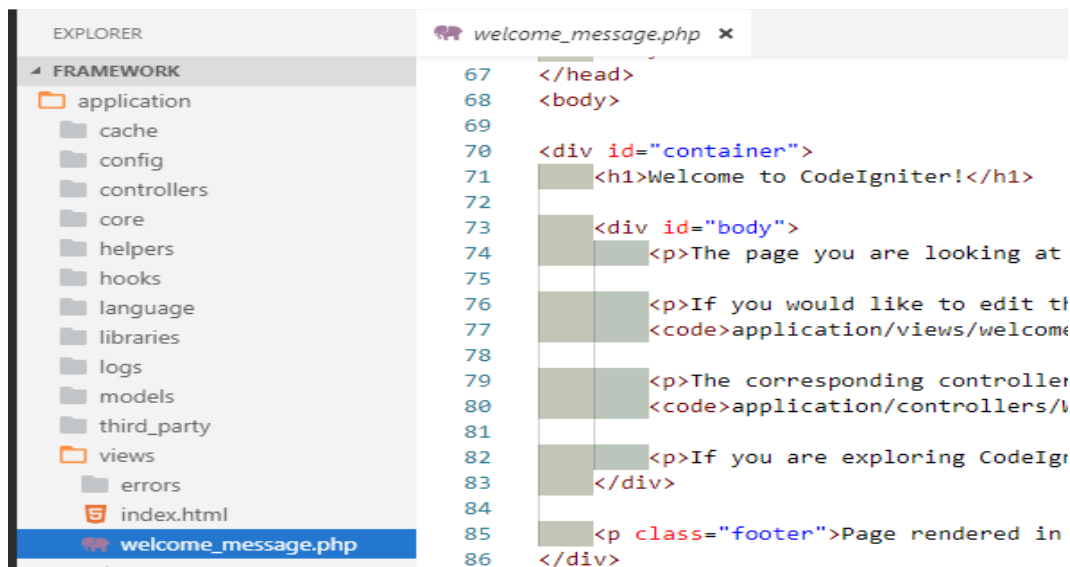
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Welcome extends CI_Controller {
5
6      public function index()
7      {
8          $this->load->view('welcome_message');
9      }
10 }
```

- a. `class Welcome extends CI_Controller`, merupakan pendeklarasian class yang bernama *Welcome* yang meng-ekstend class *Inti Codeigniter*. Peraturan dalam pembuatan class/controller dalam Codeigniter yaitu, nama class harus sama dengan nama file controller dan berawalan dengan huruf besar. (ex: **welcome.php** maka class nya adalah *Class Welcome extends CI_Controller*)

- b. `public function index()`, merupakan pendeklarasian function dalam *class*.

Sama seperti pembuatan php pada umumnya jika kita membuat `index.php` maka apabila kita membuka parent folder, akan langsung terhubung ke halaman `index.php`, begitu pula dengan `public function index`, jika kita mengakses suatu Controller yang memiliki function `index`. maka Controller yang dipilih akan langsung memproses function `index`

3. Kemudian akan memanggil view welcome message yang terletak di `application/views/welcome_message.php`




The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows the following structure:

- FRAMEWORK
 - application
 - cache
 - config
 - controllers
 - core
 - helpers
 - hooks
 - language
 - libraries
 - logs
 - models
 - third_party
 - views
 - errors
 - index.html

The code editor shows the content of `welcome_message.php`:

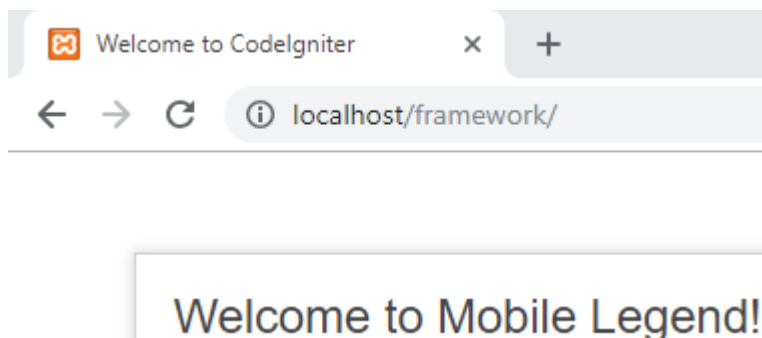
```
67 </head>
68 <body>
69
70 <div id="container">
71 <h1>Welcome to CodeIgniter!</h1>
72
73 <div id="body">
74 <p>The page you are looking at
75
76 <p>If you would like to edit th
77 <code>application/views/welcome
78
79 <p>The corresponding controller
80 <code>application/controllers/v
81
82 <p>If you are exploring CodeIgn
83 </div>
84
85 <p class="footer">Page rendered in
86 </div>
```


4. Untuk membuktikannya kita dapat mengedit file tersebut. Contoh:



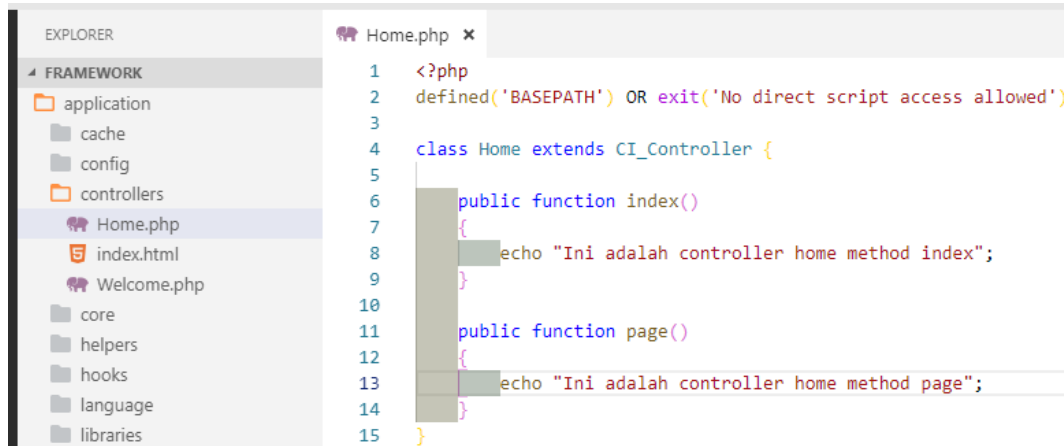
```
67 </head>
68 <body>
69
70 <div id="container">
71 <h1>Welcome to Mobile Legend!</h1>
72
73 <div id="body">
74 <p>The page you are looking at is:
75
76 <p>If you would like to edit this file, go to the file:
77 <code>application/views/welcome_message.php</code>
78
79 <p>The corresponding controller file is:
80 <code>application/controllers/welcome_message.php</code>
81
82 <p>If you are exploring CodeIgniter, you can find the file:
83 <code>application/controllers/welcome_message.php</code>
84
85 <p class="footer">Page rendered in <code>welcome_message.php</code>
86 </div>
```

5. Tampilan pada browser juga akan berubah



MEMBUAT CONTROLLER

1. Buat file baru pada application/controllers dengan nama Home.php

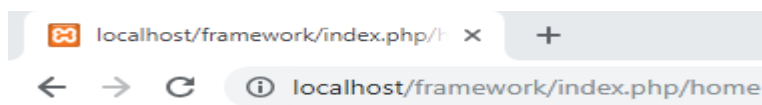


2. Perhatikan pada controller yang kita buat di atas. Pertama kali yang harus di lakukan adalah meng-extends controller baru ini dengan CI_Controller.

class Home extends CI_Controller

3. Nama class harus diawali dengan huruf besar
4. Nama class harus sesuai dengan nama file controller
5. Untuk menjalankan method index anda bisa mengaksesnya dengan alamat

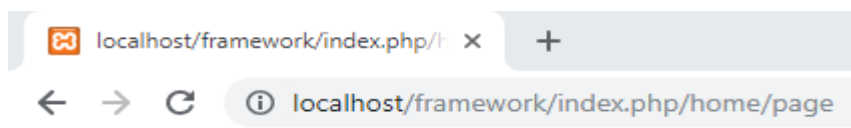
<http://localhost/framework/index.php/home/index>



Ini adalah controller home method index

6. Untuk menjalankan method index anda bisa mengaksesnya dengan alamat

<http://localhost/framework/index.php/home/page>



Ini adalah controller home method page

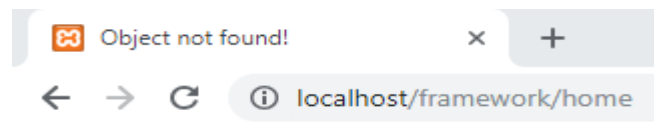
MENGHILANGKAN INDEX.PHP

1. Untuk mengakses sebuah controller kita harus menambahkan index.php

contoh: <http://localhost/framework/index.php/home>

2. Jika kita tidak menggunakan index.php maka hasilnya akan eror contoh:

<http://localhost/framework/home>

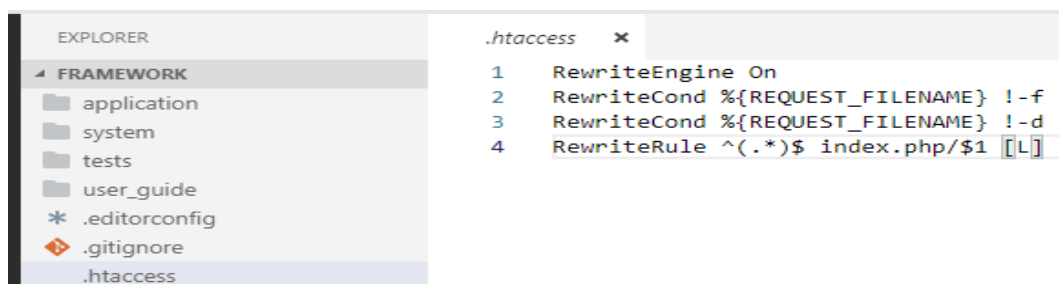


Object not found!

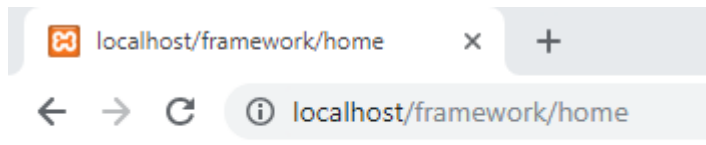
3. Untuk menghilangkan index.php agar alamat url kita terlihat menarik kita perlu membuat sebuah file .htaccess
4. Untuk menghilangkan index.php pada url codeigniter yang pertama kali kita lakukan adalah mengubah pengaturan config “index page” codeigniter yang terletak pada file config.php di [application/config/config.php](#).



5. buat htaccess pada directory root codeigniter atau folder paling luar, dapat dilihat pada http://localhost/framework/user_guide/general/urls.html?highlight=htaccess



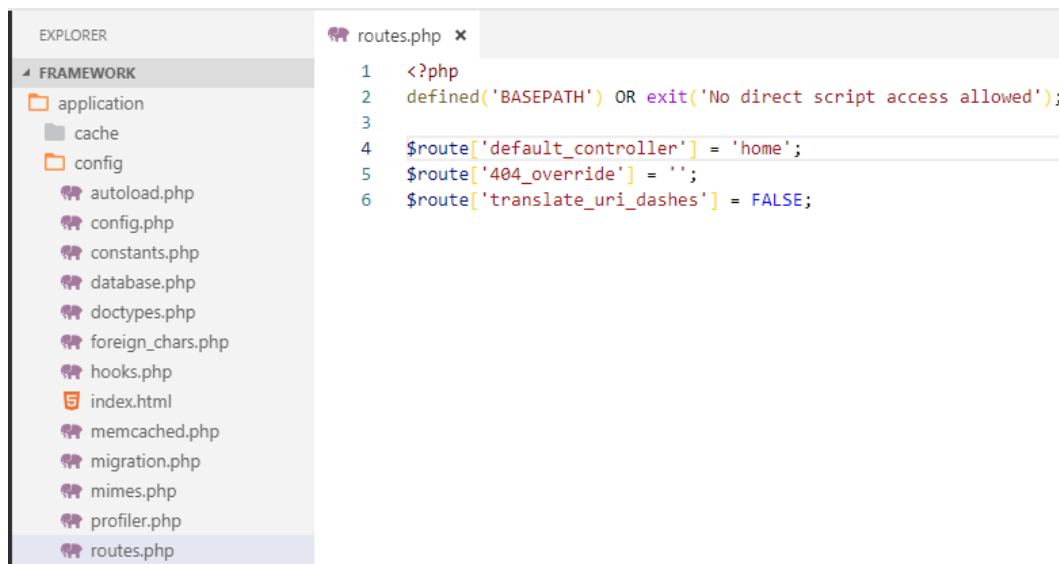
6. coba jalankan di url: <http://localhost/framework/home>



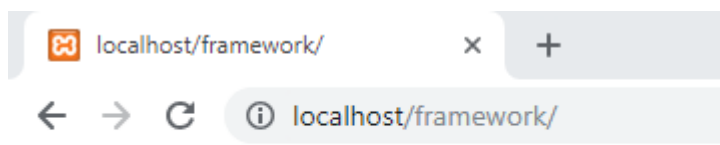
Ini adalah controller home method index

MENGGANTI CONTROLLER DEFAULT

1. Buka dan edit file application/config/routes.php
2. Ganti welcome menjadi controller yang anda inginkan



3. Lakukan pengujian di browser, url: <http://localhost/informatika>



Ini adalah controller home method index

MEMBUAT VIEW

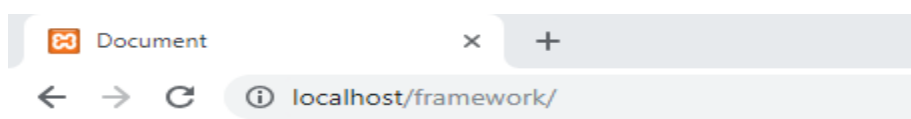
1. Buat folder baru pada application/views dengan nama home
2. Pada controller home di application/controllers/Home.php lakukan pemanggilan view yang akan dibuat

```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Home extends CI_Controller {
5
6     public function index()
7     {
8         $this->load->view('home/index');
9     }
10 }
```

3. Pada folder application/views buat folder baru dengan nama home dan buat file baru dengan nama index.php

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <meta http-equiv="X-UA-Compatible" content="IE=edge">
7     <title>Document</title>
8 </head>
9 <body>
10     <h1>Ini adalah halaman home</h1>
11 </body>
12 </html>
```

4. Jalankan di browser



Ini adalah halaman home