

CodeIgniter

<http://tinyurl.com/khhfxoz>

CodeIgniter comes in three flavors: CodeIgniter 3 (current), CodeIgniter 4 (future) and CodeIgniter 2 (legacy)

CodeIgniter 3.x

CodeIgniter 3.1.4 is the current version of the framework.

There have been a number of refinements since version 2.x, notably with the database, session handling and encryption. Development of this version is ongoing.

[View CodeIgniter 3 on Github](#)[Download CodeIgniter 3](#)[Download System Message Translations](#)

CodeIgniter 2.x

CodeIgniter 2.2.6 is the legacy version of the framework.

The 2.x branch was originally released January 2011, the next major update (2.1.0) came in the fall of 2011, version (2.2.0) came out in July, 2014, and the current version (2.2.6) came out in October, 2015.

CodeIgniter 2 has reached its end-of-life for support and updates, as of October 31, 2015. No further updates are planned. We encourage you to upgrade to CodeIgniter 3.x!

[View CodeIgniter 2 on Github](#)[Download CodeIgniter 2](#)

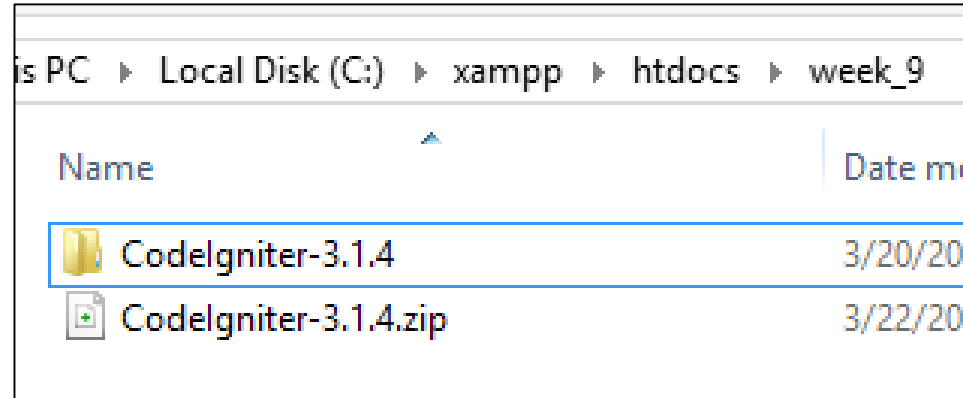
CodeIgniter 4.x

CodeIgniter 4 is the upcoming version of the framework.

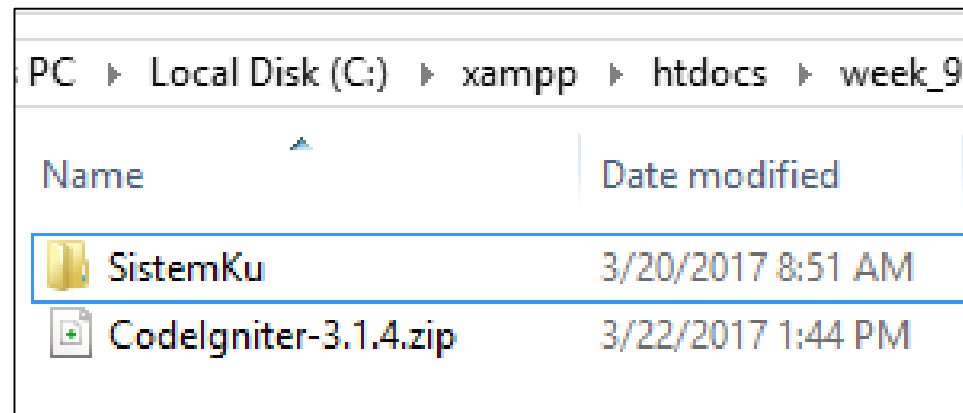
Development is underway, with Phase 1 completed in June 2016. It is **not** ready for production use - the current version is considered "pre-alpha1".

[View CodeIgniter 4 on Github](#)[Join the discussion on our forum!](#)

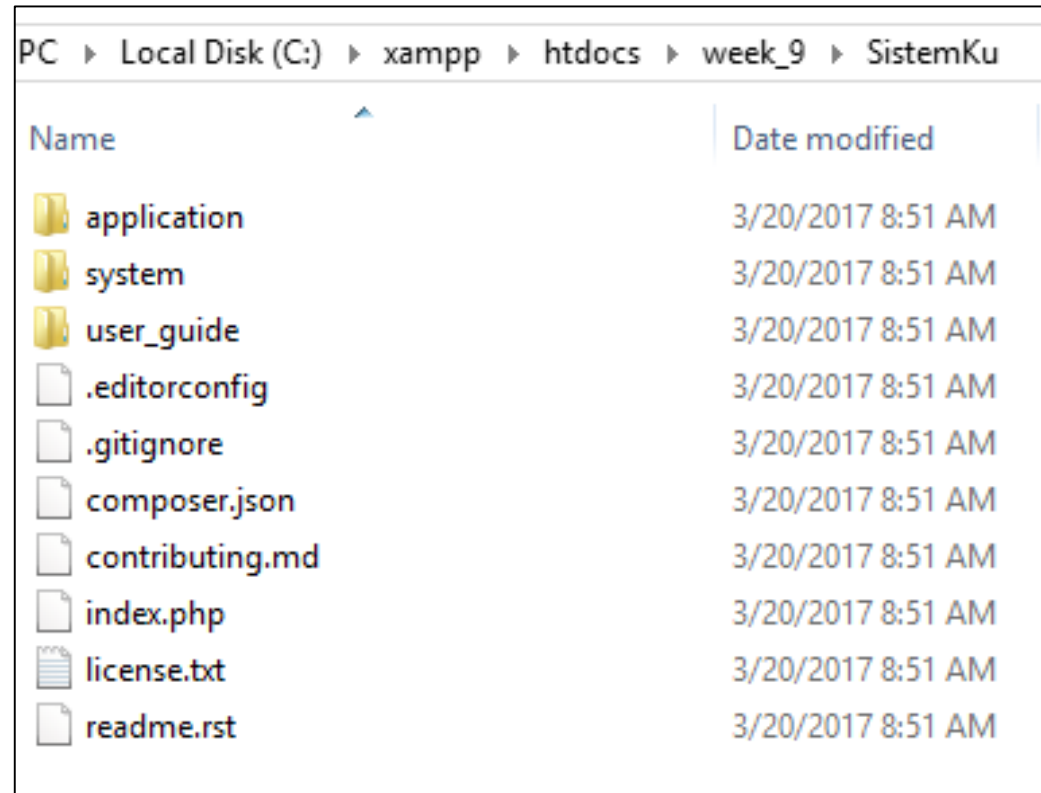
- Ekstrak file CodeIgniter.zip



- Rename Folder hasil ekstraksinya
 - Mis. SistemKu



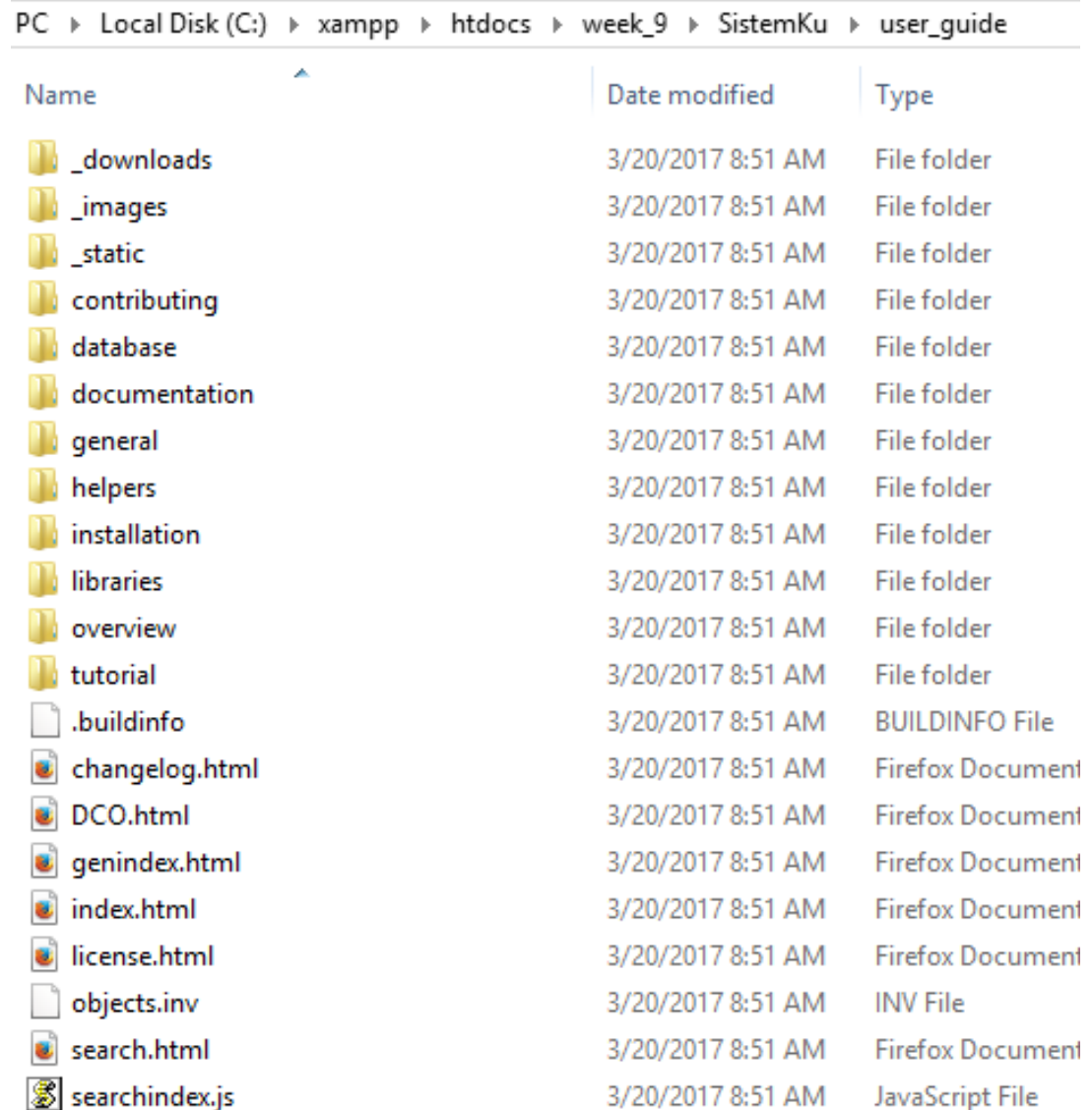
- Didalam folder hasil ekstraksi terdapat aplikasi CodeIgniter



The screenshot shows a Windows File Explorer window with the address bar displaying the path: PC > Local Disk (C:) > xampp > htdocs > week_9 > SistemKu. The main area shows a list of files and folders with columns for Name and Date modified. The files listed are:

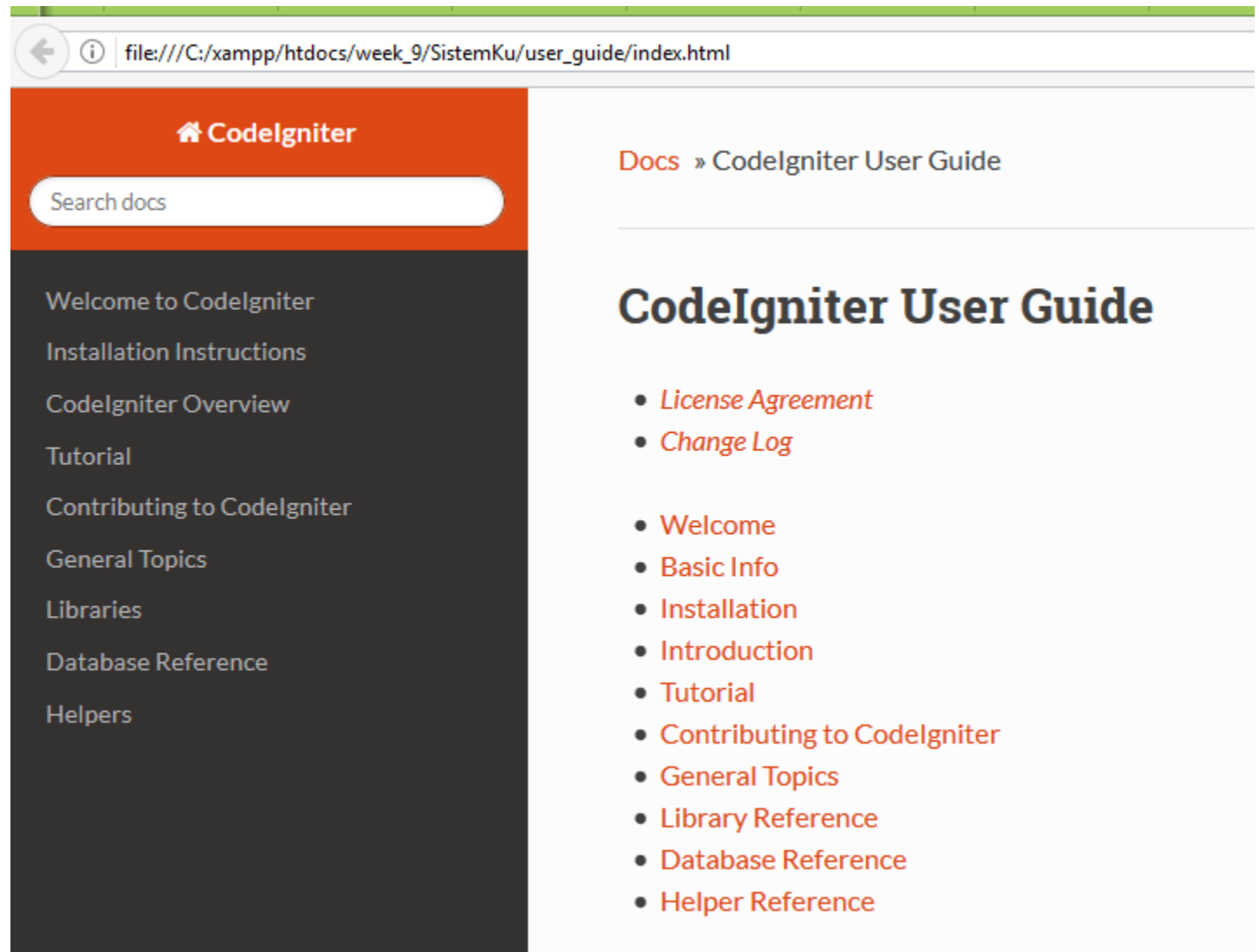
Name	Date modified
application	3/20/2017 8:51 AM
system	3/20/2017 8:51 AM
user_guide	3/20/2017 8:51 AM
.editorconfig	3/20/2017 8:51 AM
.gitignore	3/20/2017 8:51 AM
composer.json	3/20/2017 8:51 AM
contributing.md	3/20/2017 8:51 AM
index.php	3/20/2017 8:51 AM
license.txt	3/20/2017 8:51 AM
readme.rst	3/20/2017 8:51 AM

- Didalam folder user_guide sudah terdapat manual dan tutorial penggunaan CodeIgniter
- Dapat dibuka pada index.html
 - khusus file ini boleh double click saja,
 - tidak perlu lewat localhost



Name	Date modified	Type
_downloads	3/20/2017 8:51 AM	File folder
_images	3/20/2017 8:51 AM	File folder
_static	3/20/2017 8:51 AM	File folder
contributing	3/20/2017 8:51 AM	File folder
database	3/20/2017 8:51 AM	File folder
documentation	3/20/2017 8:51 AM	File folder
general	3/20/2017 8:51 AM	File folder
helpers	3/20/2017 8:51 AM	File folder
installation	3/20/2017 8:51 AM	File folder
libraries	3/20/2017 8:51 AM	File folder
overview	3/20/2017 8:51 AM	File folder
tutorial	3/20/2017 8:51 AM	File folder
.buildinfo	3/20/2017 8:51 AM	BUILDINFO File
changelog.html	3/20/2017 8:51 AM	Firefox Document
DCO.html	3/20/2017 8:51 AM	Firefox Document
genindex.html	3/20/2017 8:51 AM	Firefox Document
index.html	3/20/2017 8:51 AM	Firefox Document
license.html	3/20/2017 8:51 AM	Firefox Document
objects.inv	3/20/2017 8:51 AM	INV File
search.html	3/20/2017 8:51 AM	Firefox Document
searchindex.js	3/20/2017 8:51 AM	JavaScript File

- File index.html



The screenshot shows a web browser window displaying the CodeIgniter User Guide index page. The browser's address bar shows the file path: file:///C:/xampp/htdocs/week_9/SistemKu/user_guide/index.html. The page has a dark orange header with the CodeIgniter logo and a search bar labeled "Search docs". A dark grey sidebar on the left contains a list of navigation links: Welcome to CodeIgniter, Installation Instructions, CodeIgniter Overview, Tutorial, Contributing to CodeIgniter, General Topics, Libraries, Database Reference, and Helpers. The main content area on the right features the breadcrumb "Docs » CodeIgniter User Guide" and a large heading "CodeIgniter User Guide". Below the heading is a list of links, with the first two, "License Agreement" and "Change Log", in italics. The remaining links are: Welcome, Basic Info, Installation, Introduction, Tutorial, Contributing to CodeIgniter, General Topics, Library Reference, Database Reference, and Helper Reference.

file:///C:/xampp/htdocs/week_9/SistemKu/user_guide/index.html

CodeIgniter

Search docs

Welcome to CodeIgniter

Installation Instructions

CodeIgniter Overview

Tutorial

Contributing to CodeIgniter

General Topics

Libraries

Database Reference

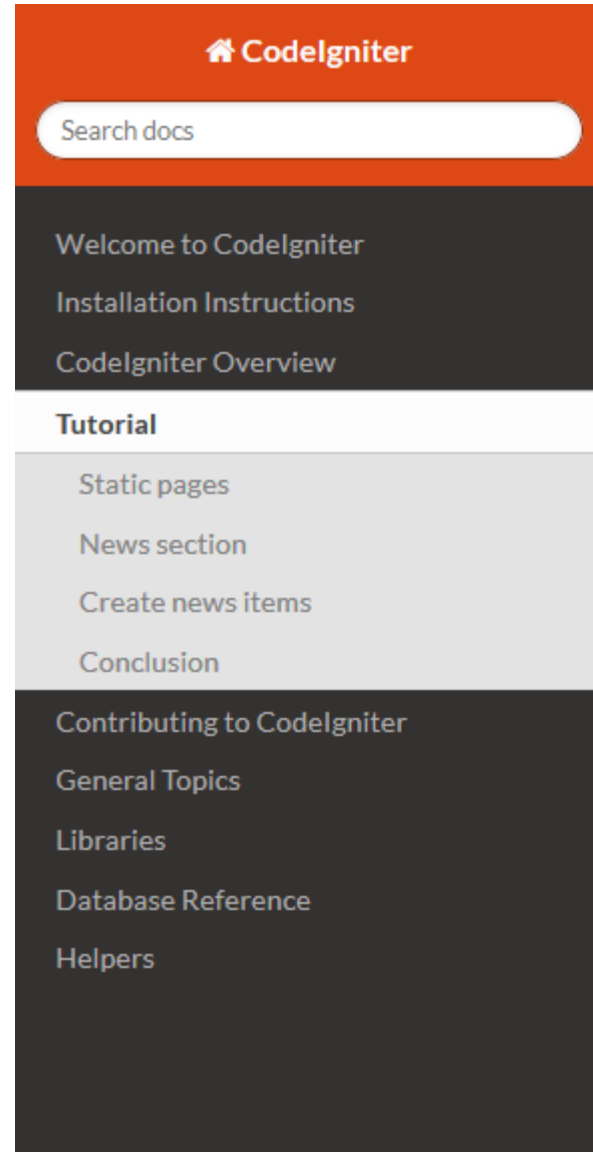
Helpers

Docs » CodeIgniter User Guide

CodeIgniter User Guide

- *License Agreement*
- *Change Log*
- Welcome
- Basic Info
- Installation
- Introduction
- Tutorial
- Contributing to CodeIgniter
- General Topics
- Library Reference
- Database Reference
- Helper Reference

- Juga terdapat tutorial penggunaan



The screenshot shows the CodeIgniter documentation website. At the top, there is an orange header with the CodeIgniter logo and a search bar labeled "Search docs". Below the header, there is a dark grey navigation menu with the following items: "Welcome to CodeIgniter", "Installation Instructions", "CodeIgniter Overview", "Tutorial", "Static pages", "News section", "Create news items", "Conclusion", "Contributing to CodeIgniter", "General Topics", "Libraries", "Database Reference", and "Helpers". The "Tutorial" item is highlighted in a lighter grey color.

Tutorial

This tutorial is intended to introduce you how a basic CodeIgniter application is co

In this tutorial, you will be creating a **bas** create a news section that reads news ite

This tutorial will primarily focus on:

- Model-View-Controller basics
- Routing basics
- Form validation
- Performing basic database queries us

The entire tutorial is split up over severa through the following pages:

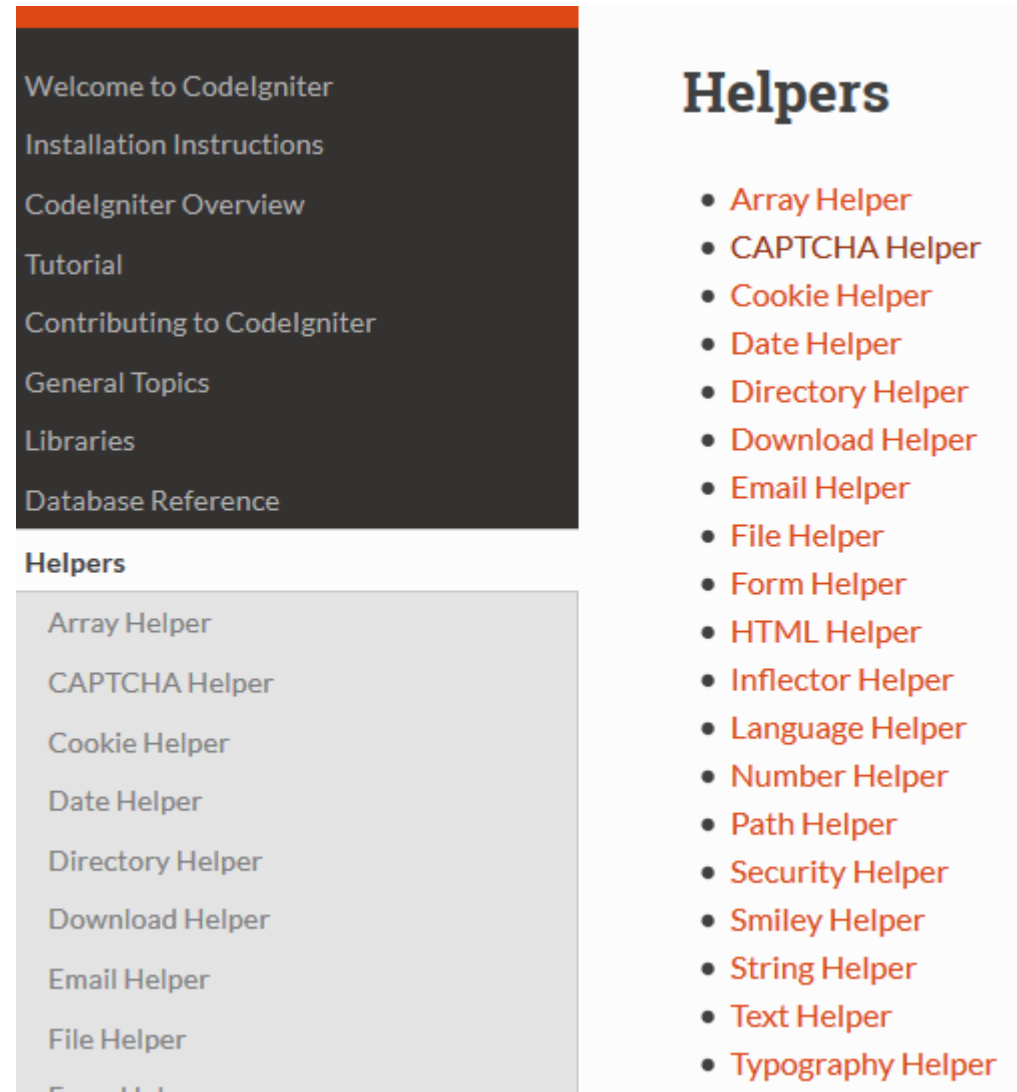
- Introduction, this page, which gives y
- *Static pages*, which will teach you the l
- *News section*, where you'll start using
- *Create news items*, which will introduc
- *Conclusion*, which will give you some p

Helper

- Yang mungkin akan sering dipelajari adalah macam-macam library yang telah dibuat
- Library inilah yang membuat pengembangan aplikasi jadi sangat cepat
 - Segala fungsi yang biasa dipakai di website sudah tersedia disini
 - Validasi form, enkripsi, paginasi, uploadfile, dll



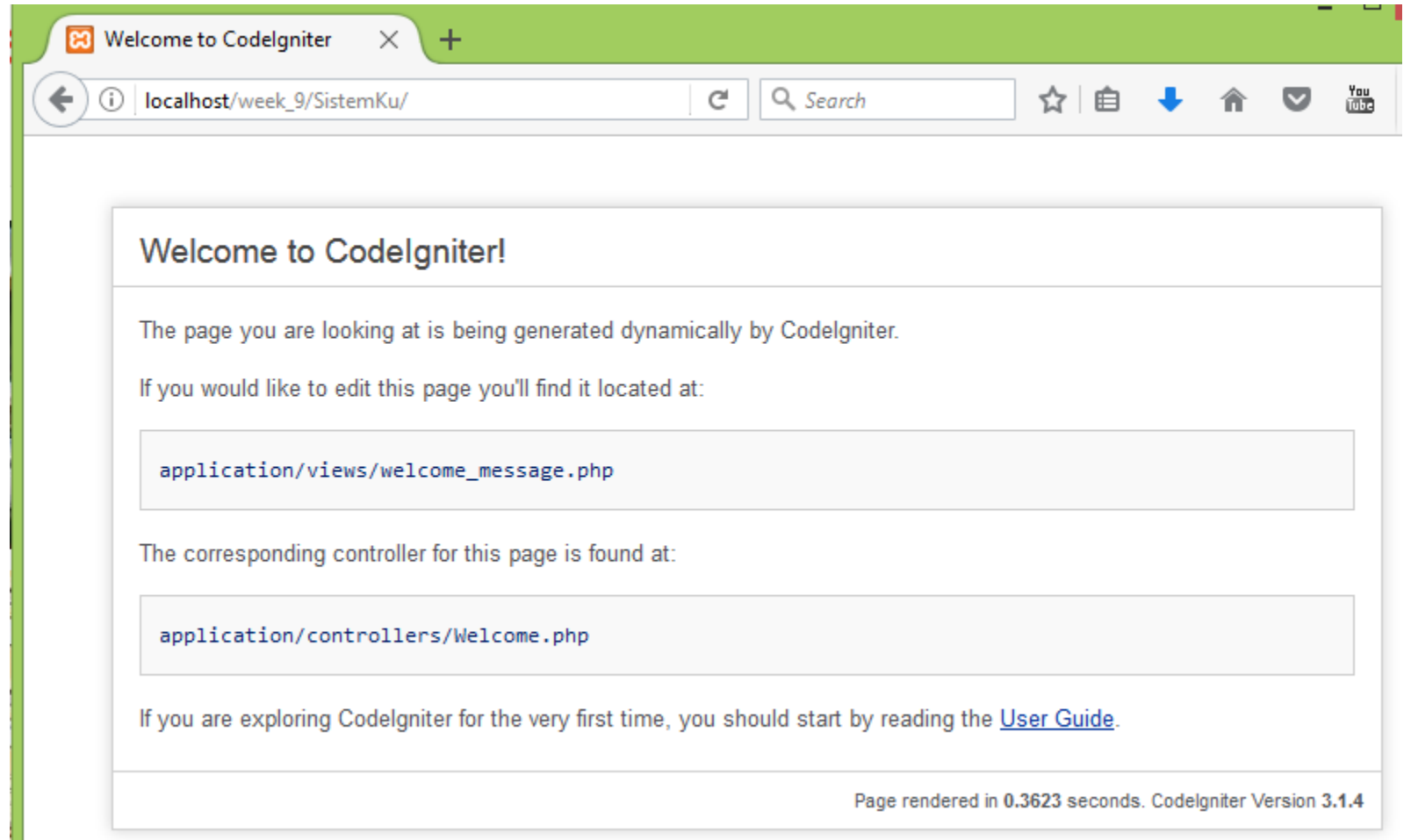
- CodeIgniter juga menyediakan fungsi-fungsi helper
 - Mis. Captcha, Membangkitkan form, format string, format URL, dll.
- Kalau library merupakan kelas, dengan method-method yang saling terhubung
 - Helper biasanya berupa fungsi-fungsi tidak saling terhubung



The image shows a screenshot of the CodeIgniter website. On the left, there is a dark navigation menu with the following items: Welcome to CodeIgniter, Installation Instructions, CodeIgniter Overview, Tutorial, Contributing to CodeIgniter, General Topics, Libraries, and Database Reference. Below this menu, the word "Helpers" is written in bold. To the right of the navigation menu, there is a light-colored box with the heading "Helpers" in bold. Below this heading, there is a list of helper classes, each preceded by a red bullet point: Array Helper, CAPTCHA Helper, Cookie Helper, Date Helper, Directory Helper, Download Helper, Email Helper, File Helper, Form Helper, HTML Helper, Inflector Helper, Language Helper, Number Helper, Path Helper, Security Helper, Smiley Helper, String Helper, Text Helper, and Typography Helper.



Jalankan Aplikasi

- Start XAMPP
 - Apache
 - Mysql
- Localhost/week_9/SistemKu






- Semua controller berada di folder application\controllers
- Agar mempermudah
 - Copas saja Controller “Welcome.php”
 - Rename menjadi “Kendali_utama.php”

Local Disk (C:) > xampp > htdocs > week_9 > SistemKu > application > controllers

Name	Date modified	Type	Size
 index.html	3/20/2017 8:51 AM	Firefox Document	
 Welcome.php	3/20/2017 8:51 AM	PHP File	

Local Disk (C:) > xampp > htdocs > week_9 > SistemKu > application > controllers

Name	Date modified	Type	Size
 index.html	3/20/2017 8:51 AM	Firefox Document	
 Kendali_utama.php	3/22/2017 7:51 PM	PHP File	
 Welcome.php	3/20/2017 8:51 AM	PHP File	

Penamaan Controller

File Naming

Class files must be named in a U
lowercase.

INCORRECT:

```
somelibrary.php  
somelibrary.php  
SOMELIBRARY.php  
Some_Library.php
```

```
Application_config.php  
Application_Config.php  
applicationConfig.php
```

CORRECT:

```
Somelibrary.php  
Some_library.php
```

```
applicationconfig.php  
application_config.php
```

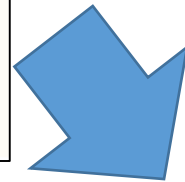
Furthermore, class file names sh
must be **Myclass.php**.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Welcome extends CI_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     *  - or -
     *      http://example.com/index.php/welcome/index
     *  - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see https://codeigniter.com/user_guide/general/urls.html
     */

    public function index()
    {
        $this->load->view('welcome_message');
    }
}
```



Edit Kendali_utama.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Kendali_utama extends CI_Controller {

}
```

- Membuat method

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Kendali_utama extends CI_Controller {

    public function contoh_method(){
        .....
        echo "Halo Dunia";
    }
}
```

Edit Kendali_utama.php

Penamaan Method

INCORRECT:

```
function fileproperties() // not de
function fileProperties() // not de
function getfileproperties() // Better
function getFileProperties() // uses C
function get_the_file_properties_from_the_file()
```

CORRECT:

```
function get_file_properties() // descriptive, u
```

URL

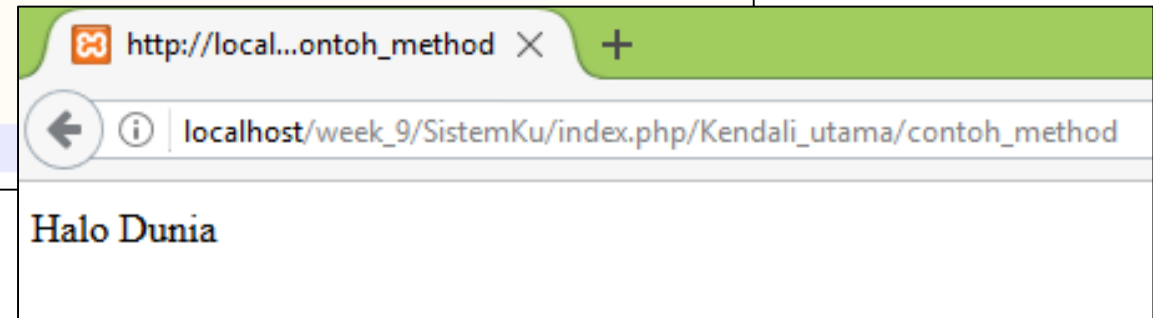
Format URL untuk aplikasi

<http://namadomain/index.php/namakontroller/namamethod/argumen>

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Kendali_utama extends CI_Controller {

    public function contoh_method() {
        echo "Halo Dunia";
    }
}
```



http://localhost/week_9/SistemKu/Kendali_utama/contoh_method

URL

Format URL untuk aplikasi

<http://namadomain/index.php/namakontroller/namamethod/argumen>

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Kendali_utama extends CI_Controller {

    public function contoh_method(){
        echo "Halo Dunia";
    }
}
```

Nama Kontroller

Nama Method

Argumen??

http://localhost/week_9/SistemKu/index.php/Kendali_utama/contoh_method

URL

- Method tentu bisa memiliki parameter, tambahkan parameter nama dan usia
- Jumlah parameter tidak terbatas, boleh ditambah, mis. nim, asal, dll

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Kendali_utama extends CI_Controller {

    public function contoh_method($nama, $usia){
        echo "Halo $nama <br/>";
        echo "$usia tahun <br/>";
    }
}
```

Edit Kendali_utama.php

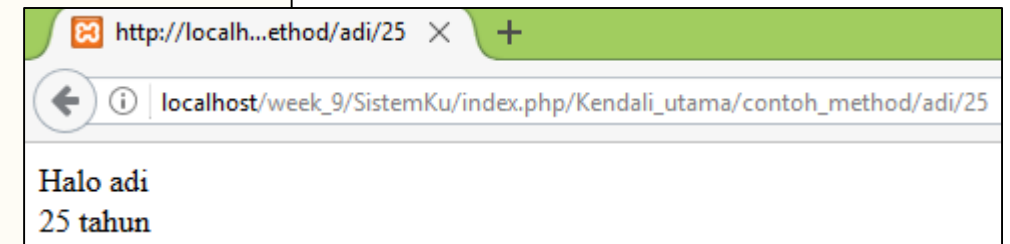
URL

Argumen

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Kendali_utama extends CI_Controller {

    public function contoh_method($nama, $usia){
        echo "Halo $nama <br/>";
        echo "$usia tahun <br/>";
    }
}
```



Nama Kontroller

Nama Method

Nilai argumen

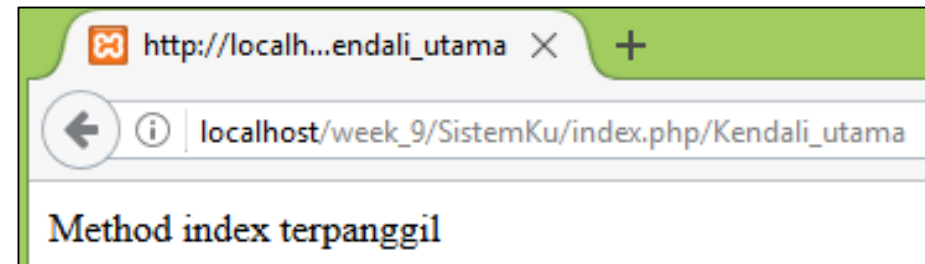
http://localhost/week_9/SistemKu/index.php/Kendali_utama/contoh_method/adi/25

Default Method

- Method **index()** adalah method default
- Kalau URL tidak menyebutkan nama method
 - Dianggap memanggil method index

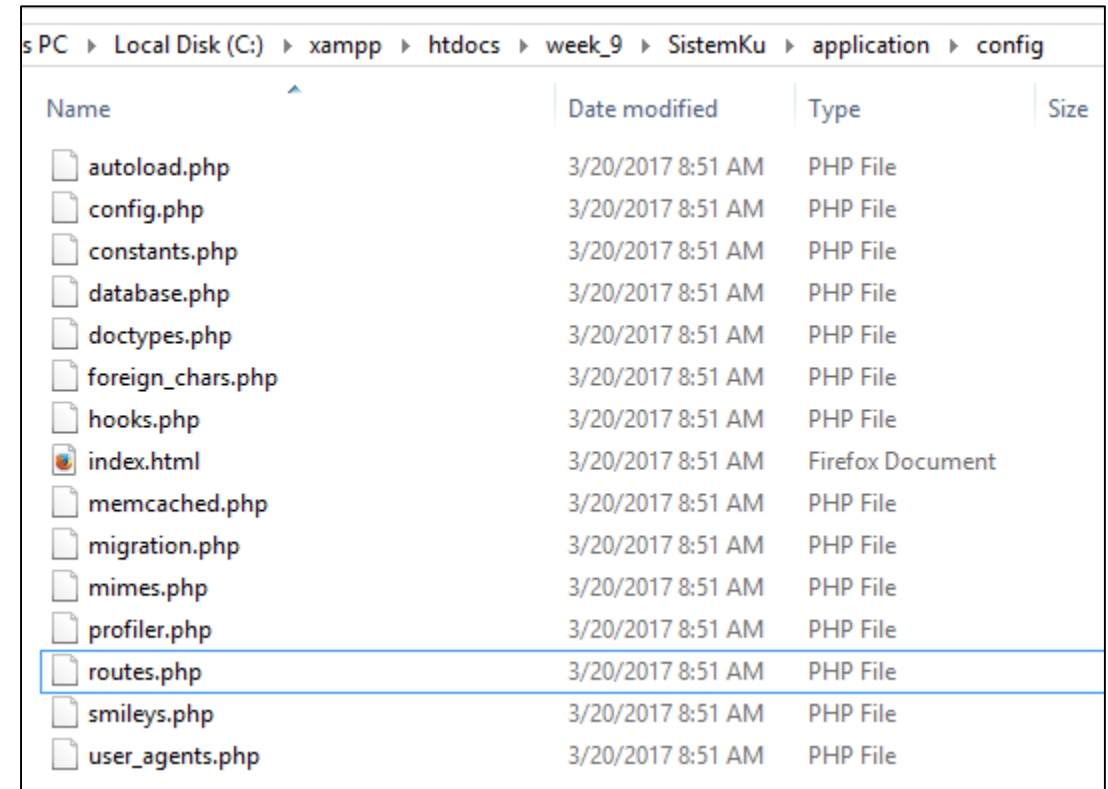
```
class Kendali_utama extends CI_Controller {  
  
    public function contoh_method($nama, $usia){  
        echo "Halo $nama <br/>";  
        echo "$usia tahun <br/>";  
    }  
  
    public function index(){  
        echo "Method index terpanggil";  
    }  
}
```

http://localhost/week_9/SistemKu/index.php/Kendali_utama



Default Controller

- Default Controller adalah controller yang otomatis dipanggil
 - apabila URL tidak menyebutkan nama controller
- Dapat diatur pada application/config/routes.php
- Edit bagian default_controller
 - Menjadi 'Kendali_utama'



Name	Date modified	Type	Size
autoload.php	3/20/2017 8:51 AM	PHP File	
config.php	3/20/2017 8:51 AM	PHP File	
constants.php	3/20/2017 8:51 AM	PHP File	
database.php	3/20/2017 8:51 AM	PHP File	
doctypes.php	3/20/2017 8:51 AM	PHP File	
foreign_chars.php	3/20/2017 8:51 AM	PHP File	
hooks.php	3/20/2017 8:51 AM	PHP File	
index.html	3/20/2017 8:51 AM	Firefox Document	
memcached.php	3/20/2017 8:51 AM	PHP File	
migration.php	3/20/2017 8:51 AM	PHP File	
mimes.php	3/20/2017 8:51 AM	PHP File	
profiler.php	3/20/2017 8:51 AM	PHP File	
routes.php	3/20/2017 8:51 AM	PHP File	
smileys.php	3/20/2017 8:51 AM	PHP File	
user_agents.php	3/20/2017 8:51 AM	PHP File	

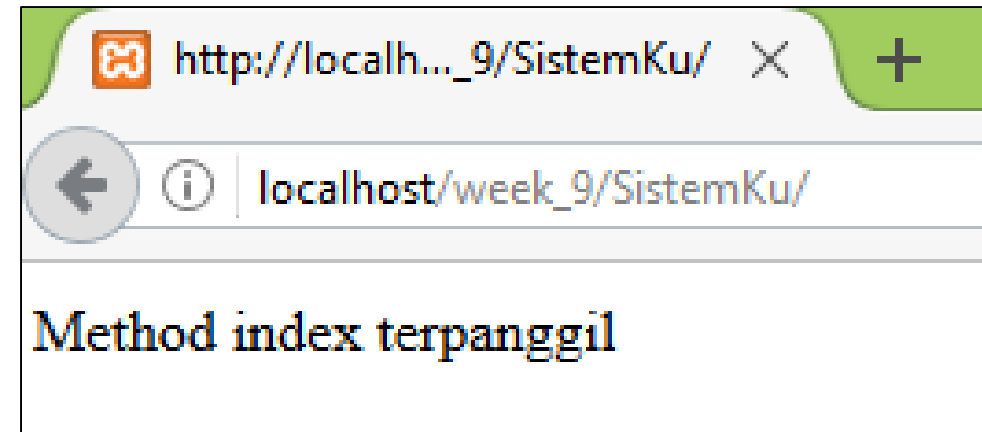
```
$route['default_controller'] = 'Kendali_utama';  
$route['404_override'] = '';  
$route['translate_uri_dashes'] = FALSE;
```

Default Controller

- Sekarang apabila URL tidak menyebutkan nama controller
 - Dianggap memanggil default controller
 - Kendali_utama
 - Dan memanggil default method
 - index()

http://localhost/week_9/SistemKu

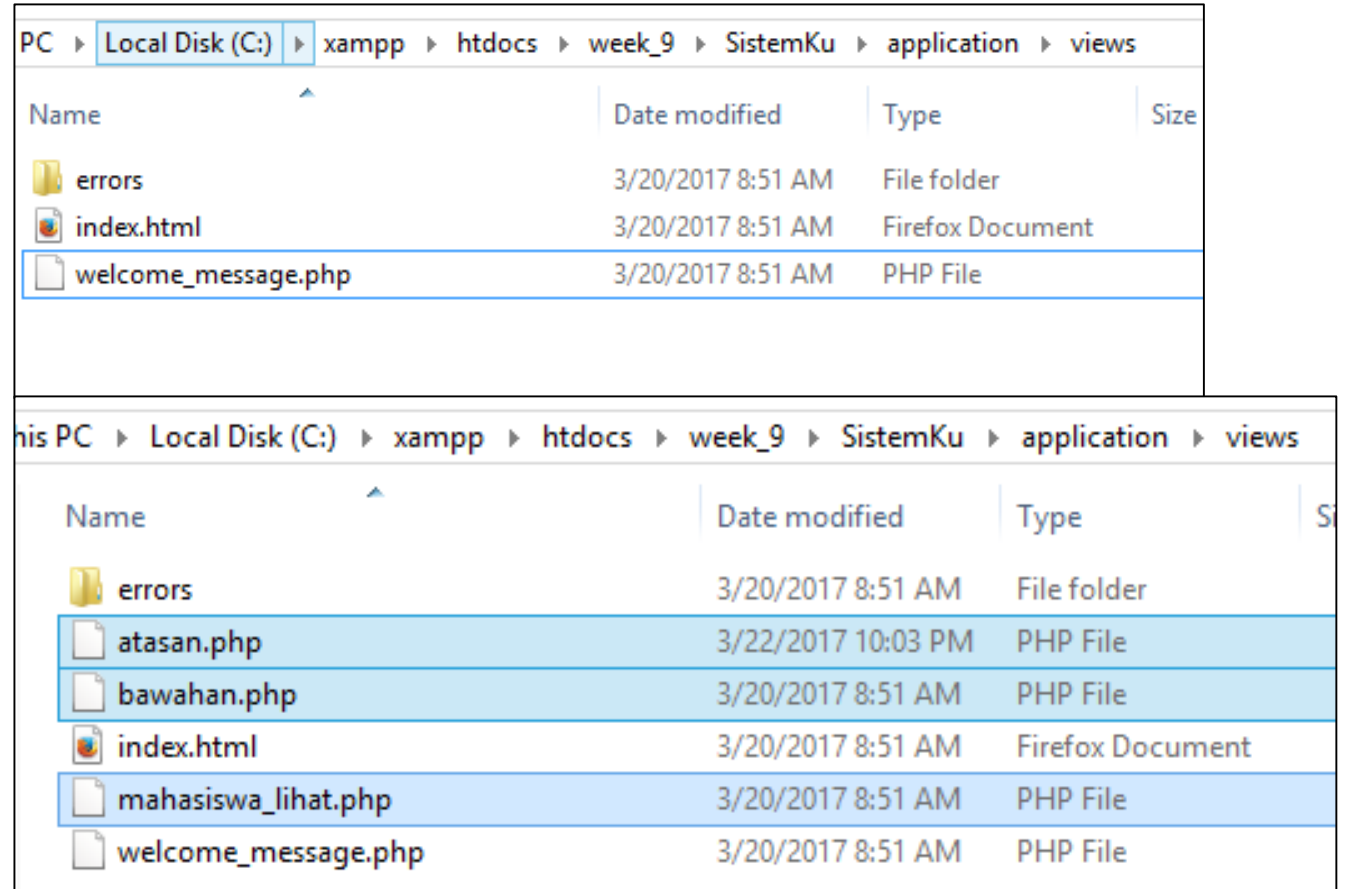
```
$route['default_controller'] = 'Kendali_utama';  
$route['404_override'] = '';  
$route['translate_uri_dashes'] = FALSE;
```



- Penggunaan echo pada controller sebenarnya melanggar konsep MVC
 - Controller seharusnya tidak menangani tampilan (View)
- echo terpaksa kita gunakan karena aplikasi belum mempunyai view

```
class Kendali_utama extends CI_Controller {  
  
    public function contoh_method($nama, $usia){  
        echo "Halo $nama <br/>";  
        echo "$usia tahun <br/>";  
    }  
    public function index(){  
        echo "Method index terpanggil";  
    }  
}
```

- Semua view terletak di application\views
- Copy paste contoh view “welcome_message.php”
 - Copas 3 kali
 - Rename menjadi
 - atasan.php
 - bawahan.php
 - mahasiswa_lihat.php



- Hapus semua bagian, kecuali beberapa baris diatas.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
?><!DOCTYPE html>
```

atasan.php

- Untuk atasan.php
 - **Jangan** hapus
<!DOCTYPE html>

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
?>
```

mahasiswa_lihat.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
?>
```

bawahan.php


```
<?php
defined('BASEPATH') OR exit('No
?><!DOCTYPE html>
<html>
<head></head>
<body>
    <h1>Sistem KU</h1>
```

Tambah pada atasan.php

```
<?php
defined('BASEPATH') OR exit(
?>
    <table border=1>
    <tr><th>Nama</th></tr>
    <tr><td>Adi</td></tr>
    </table>
```

Tambah pada mahasiswa_lihat.php

```
<?php
defined('BASEPATH') OR exit('No di
?><!DOCTYPE html>
    <p>Dibuat oleh Nama (Nim)</p>
    </body>
</html>
```

Tambah pada bawahan.php

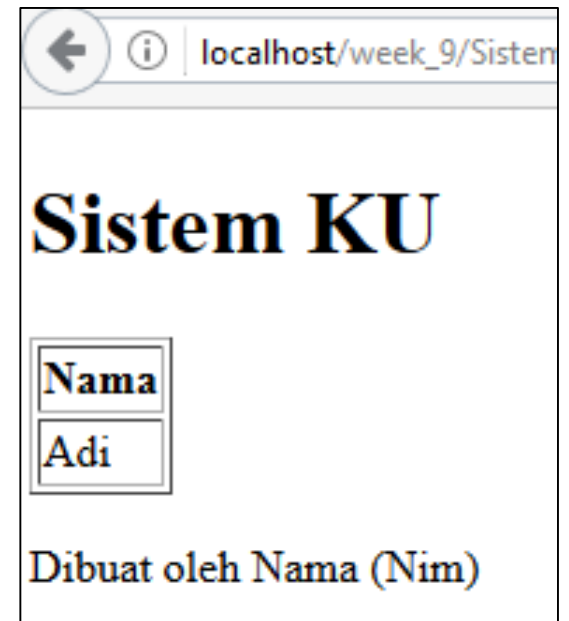
- Untuk memuat suatu view

```
$this->load->view("nama");
```

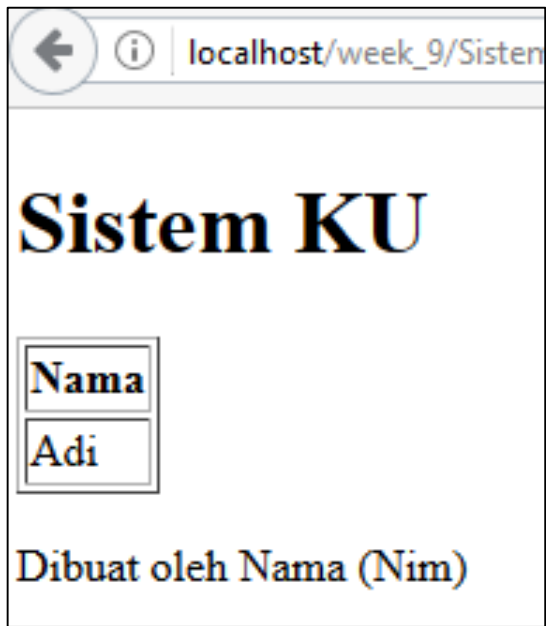
```
        echo "Method index terpanggil";  
    }  
    public function tampil_mahasiswa() {  
        $this->load->view("atasan");  
        $this->load->view("mahasiswa_lihat");  
        $this->load->view("bawahan");  
    }
```

Tambah pada Kendali_utama.php

http://localhost/week_9/SistemKu/index.php/Kendali_utama/tampil_mahasiswa



```
        echo "Method index terpanggil";
    }
    public function tampil_mahasiswa() {
        $this->load->view("atasan");
        $this->load->view("mahasiswa_lihat");
        $this->load->view("bawahan");
    }
}
```



atasan

mahasiswa_lihat

bawahan

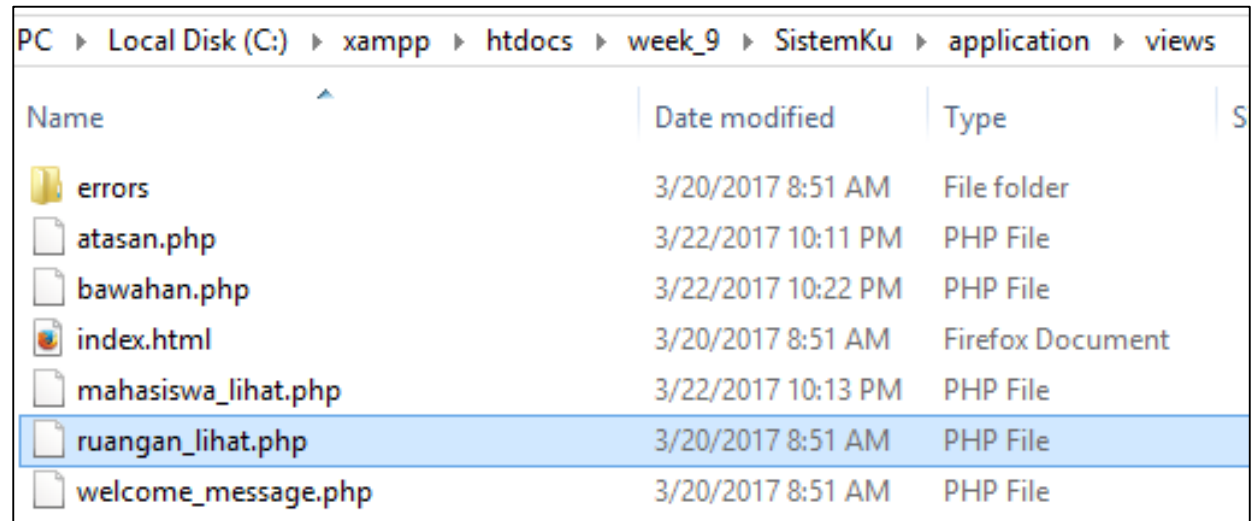
```
1 <!DOCTYPE html>
2 <html>
3 <head></head>
4 <body>
5     <h1>Sistem KU</h1>
6
7     <table border=1>
8     <tr><th>Nama</th></tr>
9     <tr><td>Adi</td></tr>
10    </table>
11    <p>Dibuat oleh Nama (Nim)</p>
12    </body>
13 </html>
14
```

View page source

Kenapa View Dibagi-bagi

- Kenapa harus dibagi-bagi (atasan, bawahan, mahasiswa_view) ?
- Karena biasanya ada bagian-bagian dari website yang tidak berubah
 - Mis. atasan, bawahan, menu, dll
 - Sehingga bagian-bagian tersebut ditulis sekali saja
 - (ingat DRY? Don't Repeat Yourself)
- Manfaat lainnya pada latihan selanjutnya..

- Copy file
“mahasiswa_lihat.php”
 - Rename menjadi
“ruangan_lihat.php”



Name	Date modified	Type	S
errors	3/20/2017 8:51 AM	File folder	
atasan.php	3/22/2017 10:11 PM	PHP File	
bawahan.php	3/22/2017 10:22 PM	PHP File	
index.html	3/20/2017 8:51 AM	Firefox Document	
mahasiswa_lihat.php	3/22/2017 10:13 PM	PHP File	
ruangan_lihat.php	3/20/2017 8:51 AM	PHP File	
welcome_message.php	3/20/2017 8:51 AM	PHP File	

Copas file mahasiswa_lihat.php ke ruangan_lihat.php

- Edit ruangan_lihat.php

```
<?php
defined('BASEPATH') OR ex
?>

<ol>
<li>Prodi</li>
<li>Kuliah</li>
</ol>
```

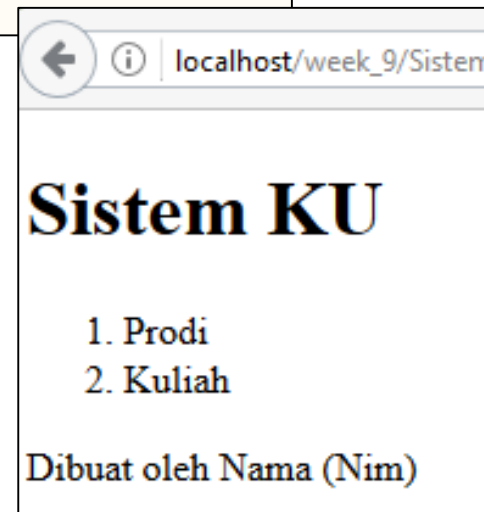
Edit ruangan_lihat.php

- Agar dapat diakses pada suatu URL sendiri
 - Edit Kendali_utama.php
 - Buat method baru
- Copas edit dari method tampil_mahasiswa()

```
public function tampil_mahasiswa() {  
    $this->load->view("atasan");  
    $this->load->view("mahasiswa_lihat");  
    $this->load->view("bawahan");  
}  
  
public function tampil_ruangan() {  
    $this->load->view("atasan");  
    $this->load->view("ruangan_lihat");  
    $this->load->view("bawahan");  
}
```

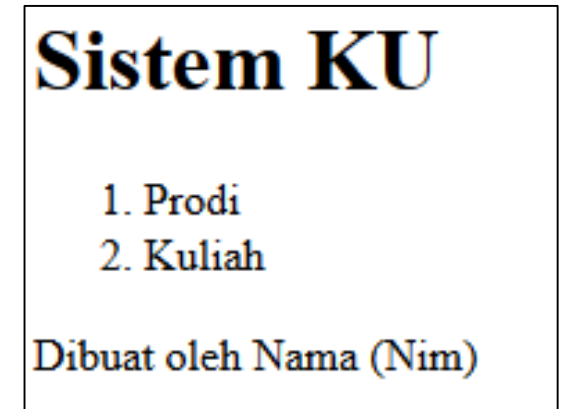
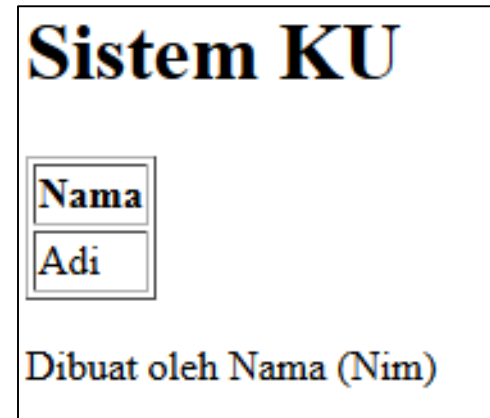
Tambah pada Kendali_utama.php
Copas edit method diatasnya saja!

http://localhost/week_9/SistemKu/index.php/Kendali_utama/tampil_mahasiswa



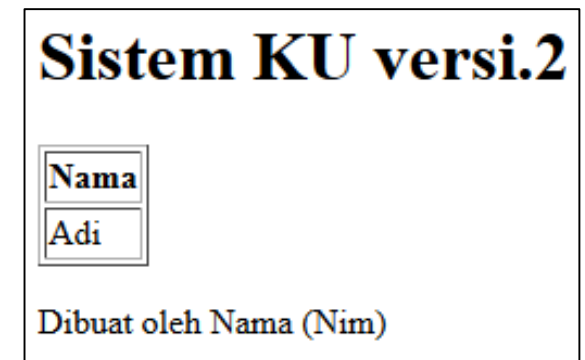
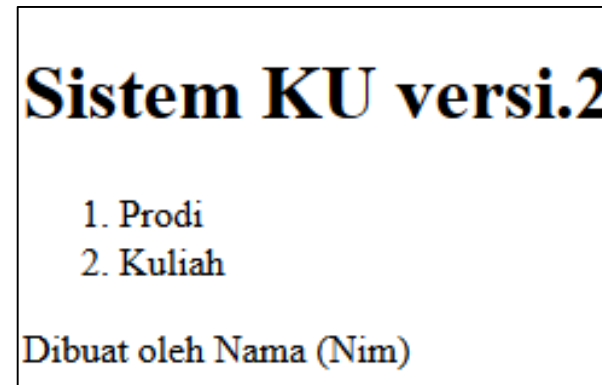
Template Website

- Atasan dan bawahan menjadi template website
- Perubahan pada bagian header cukup dilakukan pada atasan.php, footer juga pada bawahan.php



```
<?php
defined('BASEPATH') OR exit('No dir
?><!DOCTYPE html>
<html>
<head></head>
<body>
<h1>Sistem KU versi.2</h1>
```

Coba ubah atasan.php



- Hal ini terjadi karena pada controller view atasan dan bawahan tersebut dipanggil sebelum dan sesudah view lainnya.
- Terlihat perbedaan peran C dengan V
 - Controller: mengendalikan apa yang ditampilkan (atasan dulu, baru isi, terakhir bawahan)
 - View: mengatur bagaimana ditampilkan (table, list, dll)
- Bagaimana dengan M?
 - Pada MVC

```
<?php
defined('BASEPATH') OR exit('No direct script acc

class Kendali_utama extends CI_Controller {

    public function contoh_method($nama, $usia){
        echo "Halo $nama <br/>";
        echo "$usia tahun <br/>";
    }

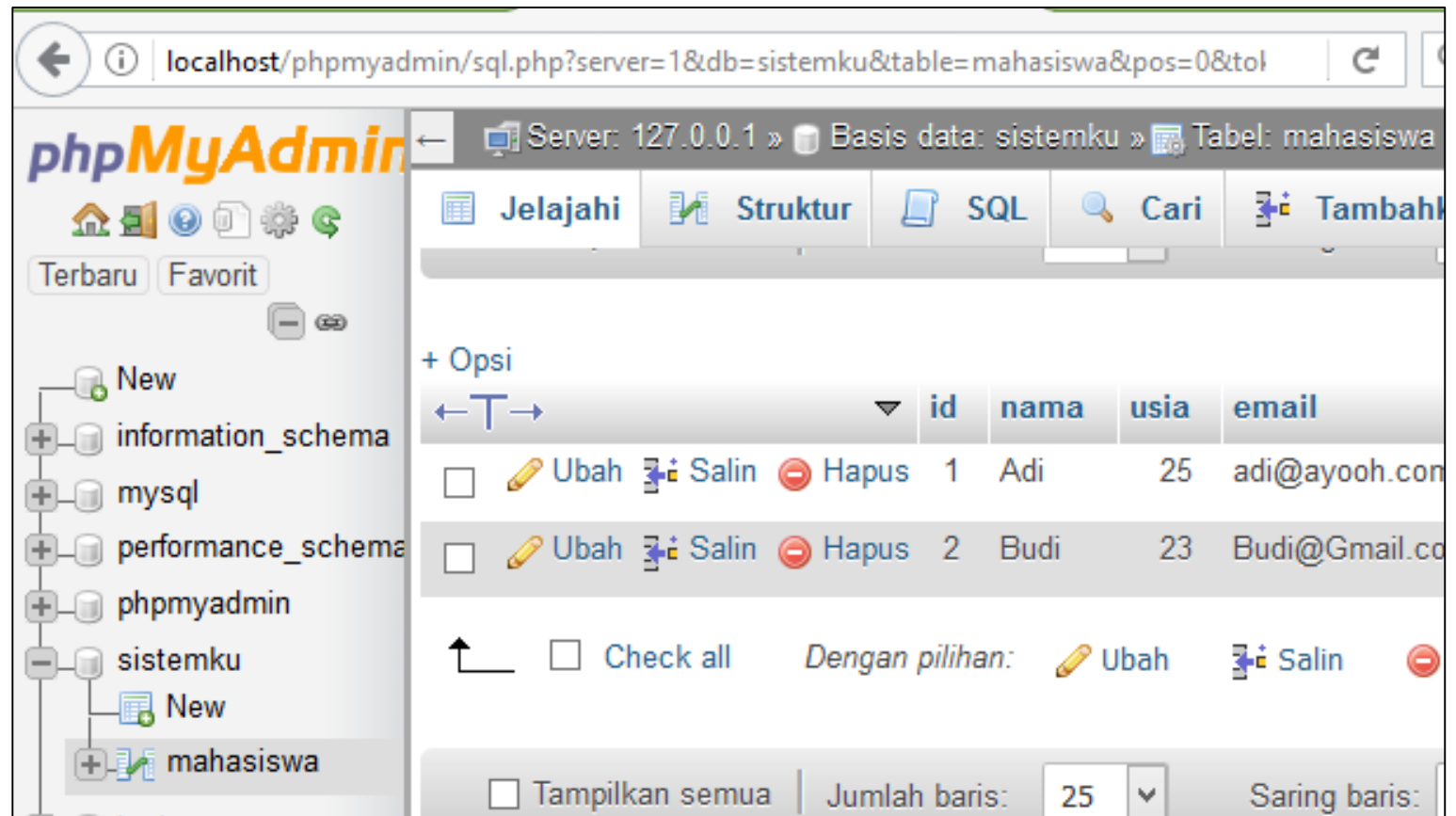
    public function index(){
        echo "Method index terpanggil";
    }

    public function tampil_mahasiswa(){
        $this->load->view("atasan");
        $this->load->view("mahasiswa_lihat");
        $this->load->view("bawahan");
    }

    public function tampil_ruangan(){
        $this->load->view("atasan");
        $this->load->view("ruangan_lihat");
        $this->load->view("bawahan");
    }
}
```


Persiapan

- Baca kembali
 - Slides minggu_6
 - Cara impor database
 - Impor file “**Basisdata Sistemku.sql**”
- Akan tercipta database **sistemku**
 - Tabel **mahasiswa**
 - Sudah terisi data



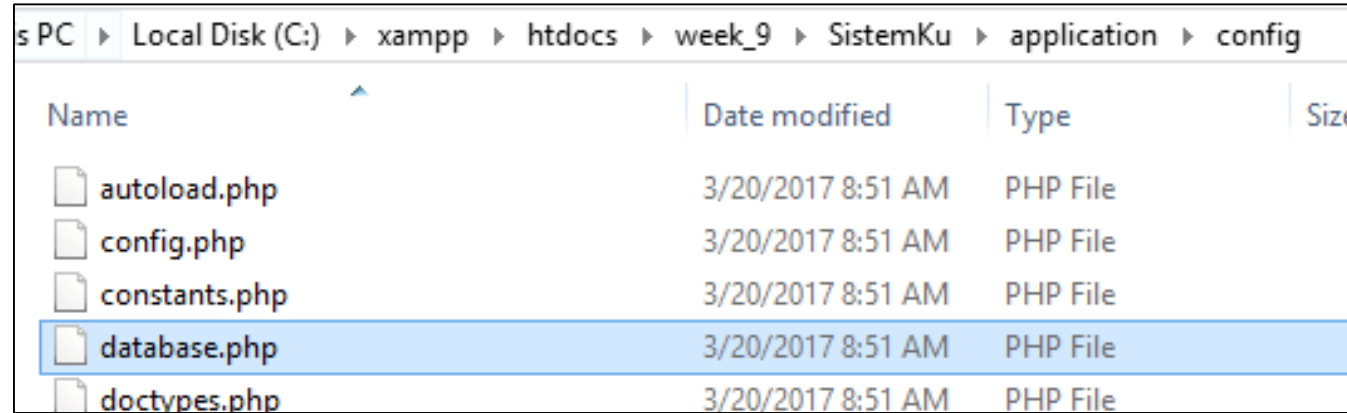
The screenshot shows the phpMyAdmin interface for a local server. The browser address bar indicates the URL: localhost/phpmyadmin/sql.php?server=1&db=sistemku&table=mahasiswa&pos=0&tol. The interface shows the 'sistemku' database selected, with the 'mahasiswa' table highlighted in the left sidebar. The main area displays the table structure and data. The table has four columns: id, nama, usia, and email. Two rows of data are visible:

	id	nama	usia	email
<input type="checkbox"/>	1	Adi	25	adi@ayooh.com
<input type="checkbox"/>	2	Budi	23	Budi@Gmail.co

Below the table, there are options to 'Check all' and 'Dengan pilihan: Ubah Salin Hapus'. At the bottom, there are options to 'Tampilkan semua', 'Jumlah baris: 25', and 'Saring baris:'.

Konfigurasi Database

- Konfigurasi database adalah pada file:
application/config/database.php



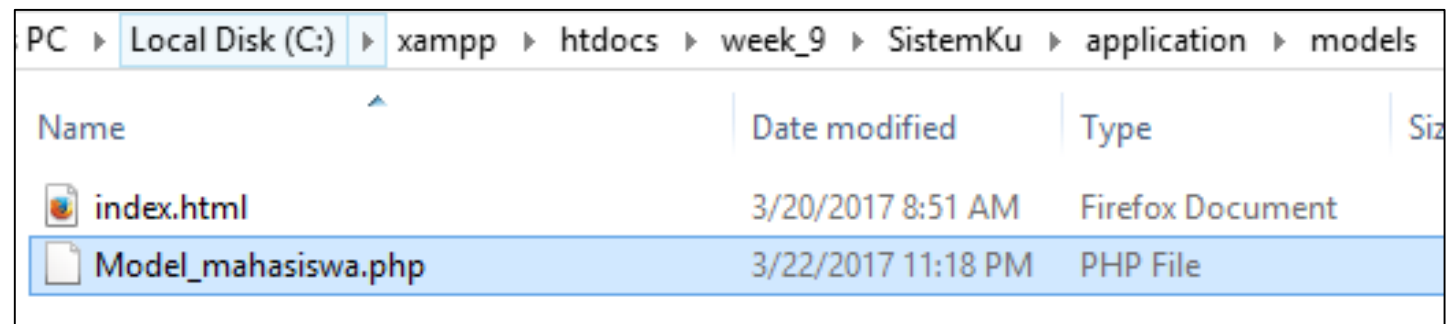
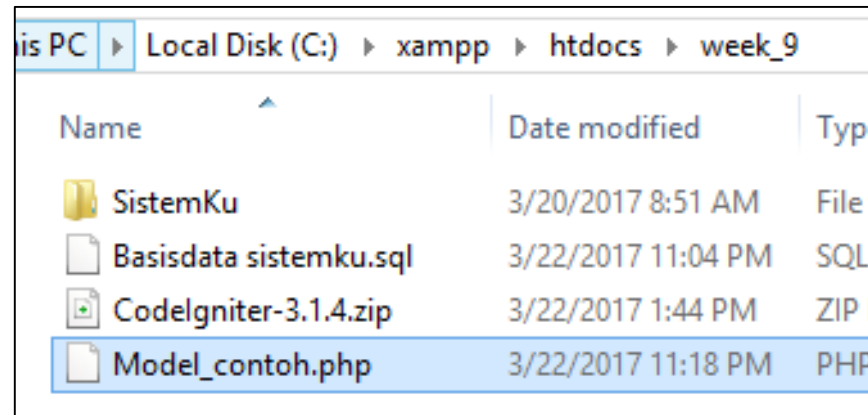
Name	Date modified	Type	Size
autoload.php	3/20/2017 8:51 AM	PHP File	
config.php	3/20/2017 8:51 AM	PHP File	
constants.php	3/20/2017 8:51 AM	PHP File	
database.php	3/20/2017 8:51 AM	PHP File	
doctypes.php	3/20/2017 8:51 AM	PHP File	

- Ubah bagian
 - hostname, username, password, database

```
$db['default'] = array(  
    'dsn' => '',  
    'hostname' => 'localhost',  
    'username' => 'root',  
    'password' => '',  
    'database' => 'sistemku',  
    'dbdriver' => 'mysqli',  
    'dbprefix' => '',  
    'pconnect' => FALSE,  
    'db_debug' => (ENVIRONMENT !  
    'cache on' => FALSE,
```

Membuat kelas Model

- Semua Model diletakkan di **application/models**
- Karena tidak ada contoh, sudah dibagikan file Model_contoh.php
 - Salin ke application/models
 - Rename menjadi Model_mahasiswa.php



Model

- Ubah nama Kelas menjadi Model_mahasiswa
- Suatu Model harus merupakan keturunan dari kelas CI_Model

```
<?php
class Model_mahasiswa extends CI_Model {

    public function __construct()
    {
        $this->load->database();
    }
}
```

Edit Model_mahasiswa.php

Memuat Library

- Untuk menggunakan database,
 - Harus memuat dahulu library database
`$this->load->database()`
- Agar otomatis memuat library database
 - perintah memuat library database, dapat diletakkan pada konstruktor Model

```
<?php
class Model_mahasiswa extends CI_Model {

    public function __construct()
    {
        $this->load->database();
    }
}
```

Edit Model_mahasiswa.php

Method pada Model

- Untuk melakukan eksekusi suatu query

`$this->db->query()`

Hasil query berupa array data, dapat diambil

menggunakan

`$hasil->result_array()`

```
<?php
class Model_mahasiswa extends CI_Model {

    public function __construct()
    {
        $this->load->database();
    }

    public function ambil_semua() {
        $hasil = $this->db->query("select * from mahasiswa");
        return $hasil->result_array();
    }
}
```

Tambah pada Model_mahasiswa.php

- Method ini melaksanakan suatu query, kemudian hasilnya dikembalikan pada pemanggil method
 - Biasanya pemanggil method adalah Controller

Memanggil Model dari Controller

- Untuk menggunakan suatu kelas Model,
 - Harus dimuat dulu

`$this->load->model("namanya");`

```
        echo "Method index terpanggil";  
    }  
    public function tampil_mahasiswa(){  
        $this->load->model("Model_mahasiswa");  
        $this->load->view("atasan");  
        $this->load->view("mahasiswa_lihat");  
        $this->load->view("bawahan");  
    }  
    public function tampil_ruangan(){
```

Tambah pada Kendali_utama.php

Memanggil method Model

Model_mahasiswa.php

- Setelah model dimuat
 - Maka method-methodnya sudah bisa dipanggil
- ```
$this->Nama_Model->nama_method($param,...)
```
- Karena method tersebut mengembalikan nilai, (return),
    - Maka nilainya ditangkap sementara ke suatu variabel

```
class Model_mahasiswa extends CI_Model {
 public function __construct()
 {
 $this->load->database();
 }
 public function ambil_semua(){
 $hasil = $this->db->query("se
 return $hasil->row_array();
 }
}
```

```
public function tampil_mahasiswa() {
 $this->load->model("Model_mahasiswa");
 $data['hasil'] = $this->Model_mahasiswa->ambil_semua();
 $this->load->view("atasan");
 $this->load->view("mahasiswa_lihat", $data);
 $this->load->view("bawahan");
}
```

Tambah pada Kendali\_utama.php



# Mengirim data ke View

- Setelah data tersimpan pada variable \$data
- Data tersebut dapat dikirim ke suatu view

```
$this->load->view("nama_view", variable);
```

- Karena data ingin ditampilkan pada view mahasiswa\_lihat, maka dapat dengan menambahkan parameter pada perintah yang sesuai

```
public function tampil_mahasiswa() {
 $this->load->model("Model_mahasiswa");
 $data['hasil'] = $this->load->Model_mahasiswa->ambil_semua();
 $this->load->view("atasan");
 $this->load->view("mahasiswa_lihat", $data);
 $this->load->view("bawahan");
}
```

Tambah pada Kendali\_utama.php

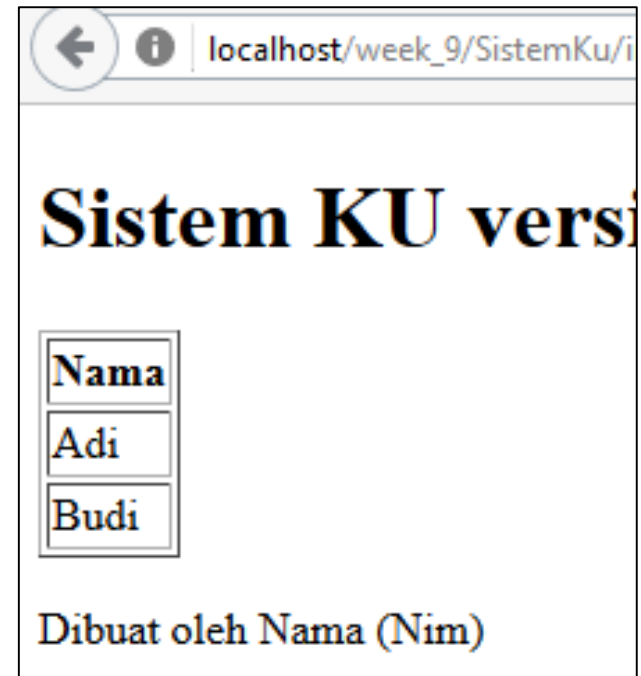
- Edit kembali mahasiswa\_view.php
- Komentari saja data dummynya.

```
<?php
defined('BASEPATH') OR exit('No direct scrip
?>
<!--
 <table border=1>
 <tr><th>Nama</th></tr>
 <tr><td>Adi</td></tr>
 </table>
-->
<?php
 echo "<table border=1>";
 echo "<tr><th>Nama</th>";
 foreach($hasil as $baris){
 echo "<tr>";
 echo "<td>".$baris["nama"]."</td>";
 echo "</tr>";
 }
 echo "</table>";
```

Tambah pada mahasiswa\_lihat.php

- Coba buka

[http://localhost/week\\_9/SistemKu/index.php/Kendali\\_utama/tampil\\_mahasiswa](http://localhost/week_9/SistemKu/index.php/Kendali_utama/tampil_mahasiswa)



# Variabel aneh

- Perhatikan mahasiswa\_lihat.php
- Terdapat variable aneh
  - Tiba-tiba ada variable \$hasil
  - Padahal tidak pernah dideklarasikan
    - Mis \$hasil = \$\_GET["a"];
    - atau \$hasil=\$\_SESSION["b"]
    - dll

```
<?php
defined('BASEPATH') OR exit('No direct scrip
?>
<!--
 <table border=1>
 <tr><th>Nama</th></tr>
 <tr><td>Adi</td></tr>
 </table>
-->
<?php
 echo "<table border=1>";
 echo "<tr><th>Nama</th>";
 foreach($hasil as $baris){
 echo "<tr>";
 echo "<td>".$baris["nama"]."</td>";
 echo "</tr>";
 }
 echo "</table>";
```

mahasiswa\_lihat.php

- Variabel tersebut adalah kiriman dari Controller
- Yang sebenarnya diambil dari Model\_Mahasiswa
- \$baris["nama"]
  - "nama" harus sesuai dengan nama kolom pada table
  - Lihat di phpmyadmin

```
public function tampil_mahasiswa() {
 $this->load->model("Model_mahasiswa");
 $data['hasil'] = $this->load->Model_mahasiswa->ambil_semua();
 $this->load->view("atasan");
 $this->load->view("mahasiswa_lihat", $data);
 $this->load->view("bawahan");
}
```

Harus sama

```
<?php
echo "<table border=1>";
echo "<tr><th>Nama</th>";
foreach($hasil as $baris) {
 echo "<tr>";
 echo "<td>".$baris["nama"]."</td>";
 echo "</tr>";
}
echo "</table>";
```

mahasiswa\_lihat.php

| Ops                      |                  | id | nama | usia | email  |
|--------------------------|------------------|----|------|------|--------|
| <input type="checkbox"/> | Ubah Salin Hapus | 1  | Adi  | 25   | adi@ay |
| <input type="checkbox"/> | Ubah Salin Hapus | 2  | Budi | 23   | Budi@G |

```
public function ambil_semua(){
 $hasil = $this->db->query("select * from mahasiswa");
 return $hasil->result_array();
}
```

```
public function tampil_mahasiswa(){
 $this->load->model("Model_mahasiswa");
 $data['hasil'] = $this->load->Model_mahasiswa->ambil_semua();
 $this->load->view("atasan");
 $this->load->view("mahasiswa_lihat", $data);
 $this->load->view("bawahan");
}
```

## • MVC

- Model hanya mengurus data
  - Tidak tahu datanya nanti digunakan untuk apa
  - Tidak tahu datanya ditampilkan seperti apa
- Controller hanya mengurus kendali
  - Tidak tahu data dari table mana
  - Tidak tahu data ditampilkan seperti apa
- View hanya mengurus tampilan
  - Tidak tahu datanya dari mana (database, session, cookies, hasil komputasi, dll)

```
<?php
echo "<table border=1>";
echo "<tr><th>Nama</th>";
foreach($hasil as $baris){
 echo "<tr>";
 echo "<td>".$baris["nama"]."</td>";
 echo "</tr>";
}
echo "</table>";
```