

LAPORAN TUGAS PEMROGRAMAN WEB LANJUT

Jobsheet 2

Read dan Input Database serta Form Validation



Disusun Oleh :

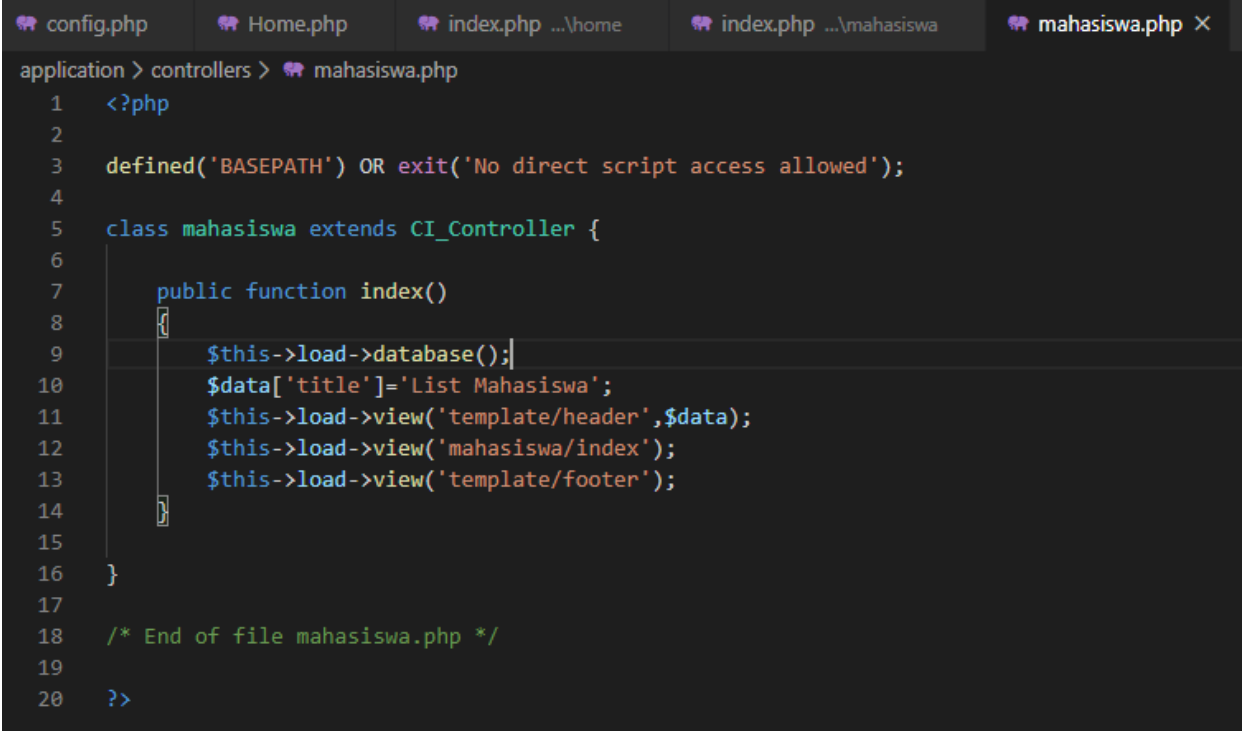
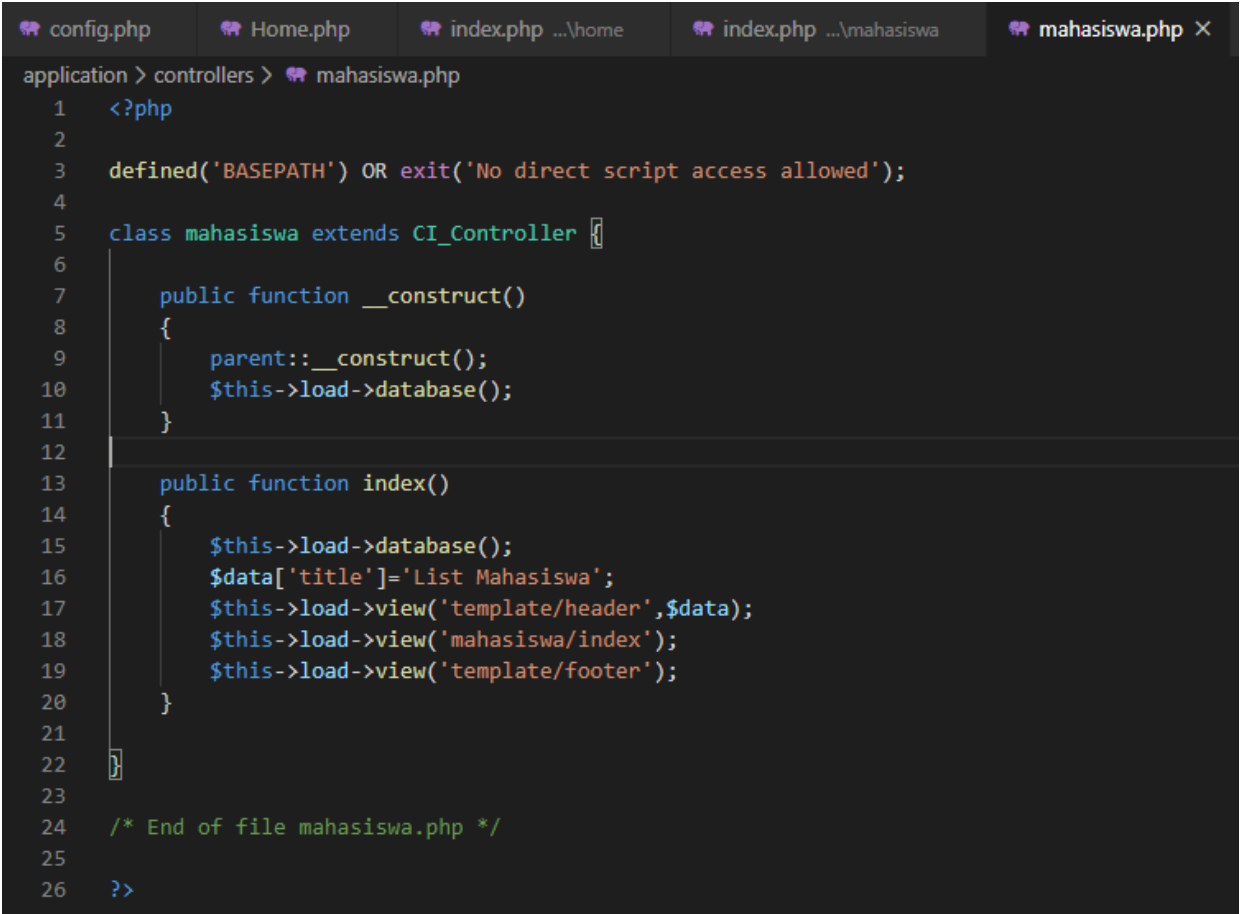
Nama : Hunayn Risatayn

NIM : 1841720148

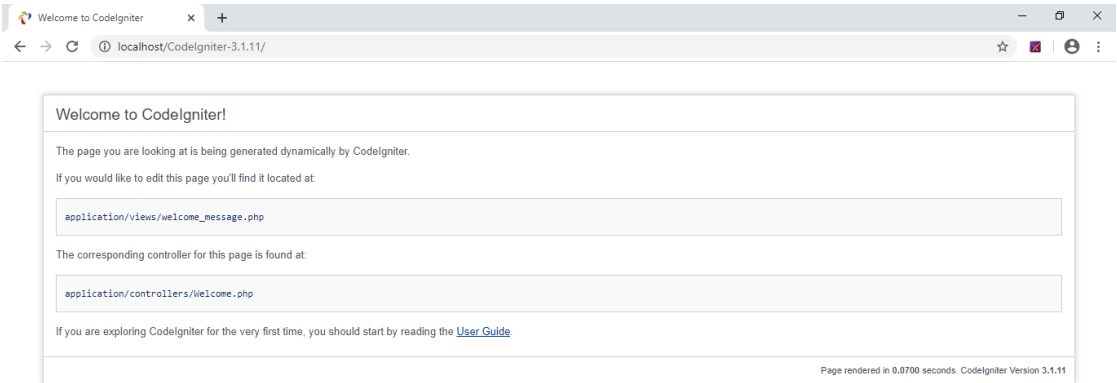
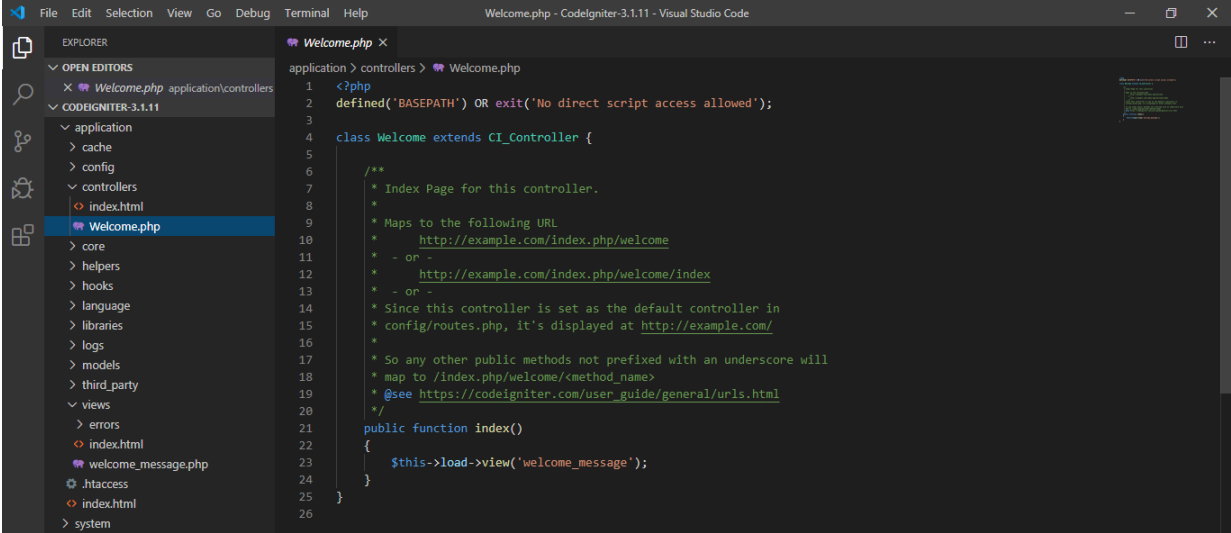
Kelas : TI-2B

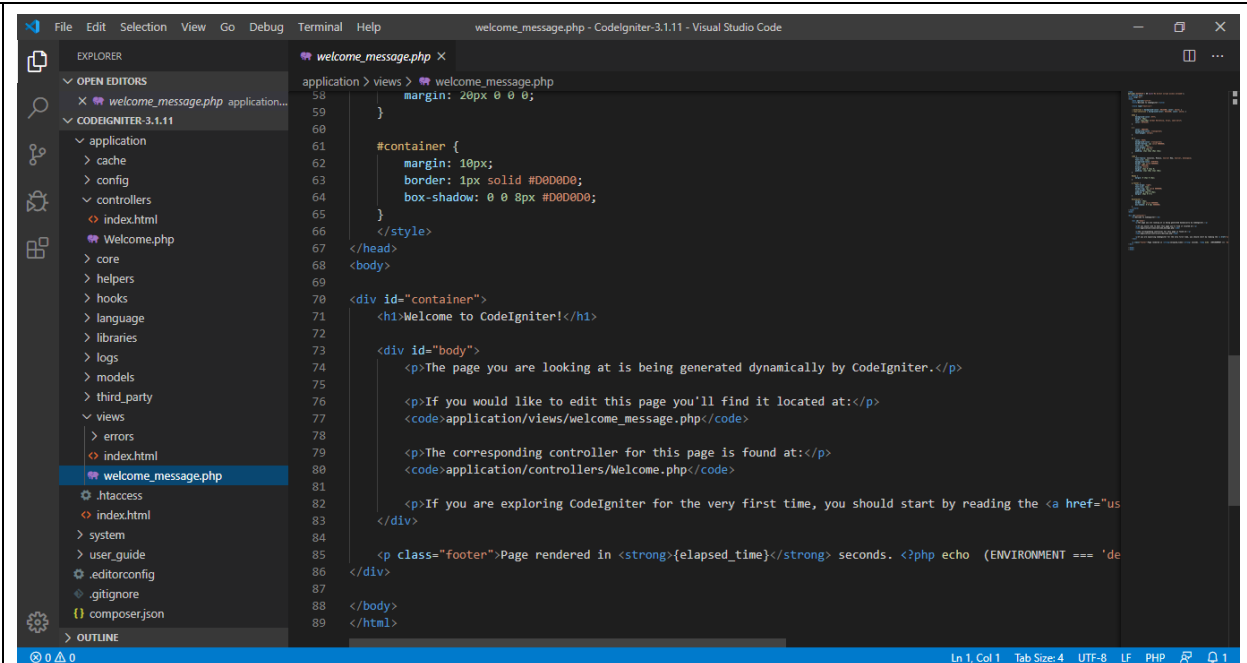
Program Studi D-IV Teknik Informatika
Jurusan Teknologi Informasi
Politeknik Negeri Malang

Praktikum – Bagian 1: Menampilkan data dari database menggunakan codeIgniter

Langkah	Keterangan
1	Buka kembali folder CodeIgniter yang telah dibuat pada jobsheet sebelumnya
2	<p>Buka controller mahasiswa dan tambahkan code berikut</p>  <pre> 1 <?php 2 3 defined('BASEPATH') OR exit('No direct script access allowed'); 4 5 class mahasiswa extends CI_Controller { 6 7 public function index() 8 { 9 \$this->load->database(); 10 \$data['title']='List Mahasiswa'; 11 \$this->load->view('template/header',\$data); 12 \$this->load->view('mahasiswa/index'); 13 \$this->load->view('template/footer'); 14 } 15 } 16 17 18 /* End of file mahasiswa.php */ 19 20 ?> </pre>
3	<p>Tambahkan code berikut agar database dapat digunakan banyak <i>function</i></p>  <pre> 1 <?php 2 3 defined('BASEPATH') OR exit('No direct script access allowed'); 4 5 class mahasiswa extends CI_Controller { 6 7 public function __construct() 8 { 9 parent::__construct(); 10 \$this->load->database(); 11 } 12 13 public function index() 14 { 15 \$this->load->database(); 16 \$data['title']='List Mahasiswa'; 17 \$this->load->view('template/header',\$data); 18 \$this->load->view('mahasiswa/index'); 19 \$this->load->view('template/footer'); 20 } 21 } 22 23 24 /* End of file mahasiswa.php */ 25 26 ?> </pre>

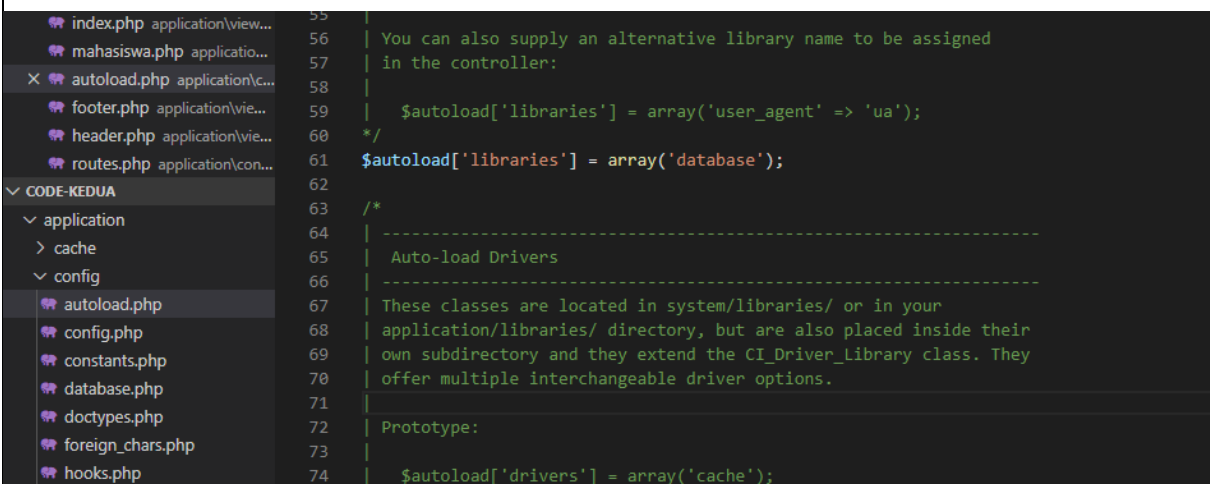
Praktikum – Bagian 1 : Menampilkan codeIgniter Pada Halaman Web

Langkah	Keterangan
1	<p>Setelah xampp aktif, ketik urlnya: sesuai dengan nama folder codeIgniter (ditambah localhost/ di depannya)</p> 
2	<p>Penjelasan dari tampilan default codeIgniter :</p> <ul style="list-style-type: none"> • Jika dilihat terdapat tulisan <code>application/controllers/welcome.php</code>, hal ini dapat dibuktikan dengan membuka folder <code>application/controller/welcome.php</code>  <p>Di dalam <code>Welcome.php</code> ini terdapat sebuah class dengan nama <code>Welcome</code> yang merupakan <i>extends</i> atau turunan dari <code>CI_Controller</code>. Di dalam Class ini terdapat sebuah <i>function</i> dengan nama <code>index</code> dan di dalam <i>function</i> <code>index</code> ini terdapat sebuah perintah untuk melakukan <i>load</i> pada halaman <i>view</i> dengan nama <code>welcome_message</code>. Dan untuk viewnya terdapat pada <code>application/view/welcome_message.php</code></p>



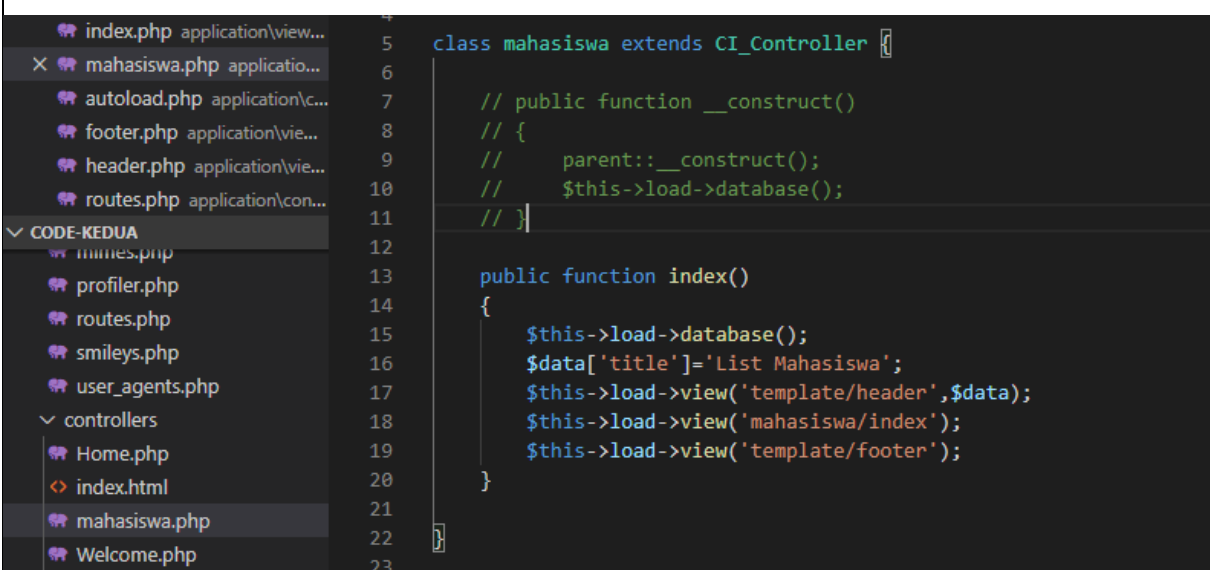
3

Selain 2 cara diatas kita juga dapat melakukan proses *load* database pada **application/config/autoload** dan tambahkan pada database pada **autoload['libraries']=array('database')** seperti pada gambar



4

Untuk kasus ini kita gunakan cara yang terakhir (no 4) karena kita asumsikan bahwa semua *controller* membutuhkan database. Maka kita *comment code* function construct yang ada di *controller* mahasiswa



5

Setelah itu kita akan mengatur konfigurasi database. Untuk mengatur konfigurasi database maka buka folder **application/config/database.php**



```
70 | The $query_builder variables lets you determine whether or not to load
71 | the query builder class.
72 */
73 $active_group = 'default';
74 $query_builder = TRUE;
75
76 $db['default'] = array(
77     'dsn' => '',
78     'hostname' => 'localhost',
79     'username' => 'root',
80     'password' => '',
81     'database' => 'kuliah_CI',
82     'dbdriver' => 'mysqli',
83     'dbprefix' => '',
84     'pconnect' => FALSE,
85     'db_debug' => (ENVIRONMENT !== 'production'),
86     'cache_on' => FALSE,
87     'cachedir' => '',
88     'char_set' => 'utf8',
89     'dbcollat' => 'utf8_general_ci',
90     'swap_pre' => '',
91     'encrypt' => FALSE,
92     'compress' => FALSE,
93     'stricton' => FALSE,
94     'failover' => array(),
95     'save_queries' => TRUE
96 );
97
```

6


Jika kita lihat konfigurasi database yang kita buat maka kita perlu sebuah database dengan nama **kuliah_CI**. Maka buat lah sebuah database dengan nama kuliah_CI dan dengan tabel mahasiswa seperti pada gambar dibawah ini



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	nama	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
3	nim	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
4	email	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
5	jurusan	varchar(50)	latin1_swedish_ci		No	None			Change Drop More

7

Tambahkan beberapa data ke dalam tabel mahasiswa seperti pada gambar (pada contoh terdapat 3 data, jumlah data lebih banyak tidak masalah)

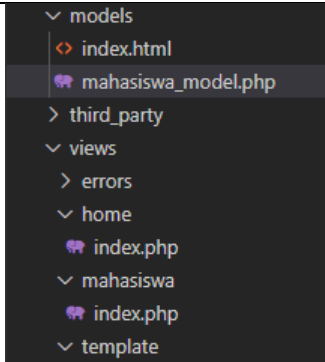


	id	nama	nim	email	jurusan
	1	dina	1002345100	dina@gmail.com	kimia
	2	dika	1002345101	dika@gmail.com	informatika
	3	soni	1002345102	soni@gmail.com	mesin

8

Setelah kita membuat sebuah *database*, maka langkah berikutnya adalah membuat sebuah model untuk melakukan proses *database*. Di dalam model ini kita akan membuat sebuah *function* untuk melakukan perintah ambil data dari *database*. Untuk lebih jelasnya ikuti langkah-langkah berikut:

a. Buatlah sebuah file model dengan nama **mahasiswa_model** pada folder **application/models**



b. Tambahkan *function* `getAllmahasiswa` pada `mahasiswa_model.php`

```
fig.php  Home.php  index.php ...\home  index.php ...\mahasiswa  mahasiswa.php

application > models > mahasiswa_model.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class mahasiswa_model extends CI_Model {
6
7      public function getAllmahasiswa()
8      {
9          $query=$this->db->get('mahasiswa');
10         return $query->result_array();
11     }
12
13 }
14
15 /* End of file mahasiswa_model.php */
16
17 ?>
```

9

Setelah kita mempunyai sebuah model dengan nama **mahasiswa_model** maka langkah berikutnya kita akan *load* model ini ke dalam **controller** mahasiswa.

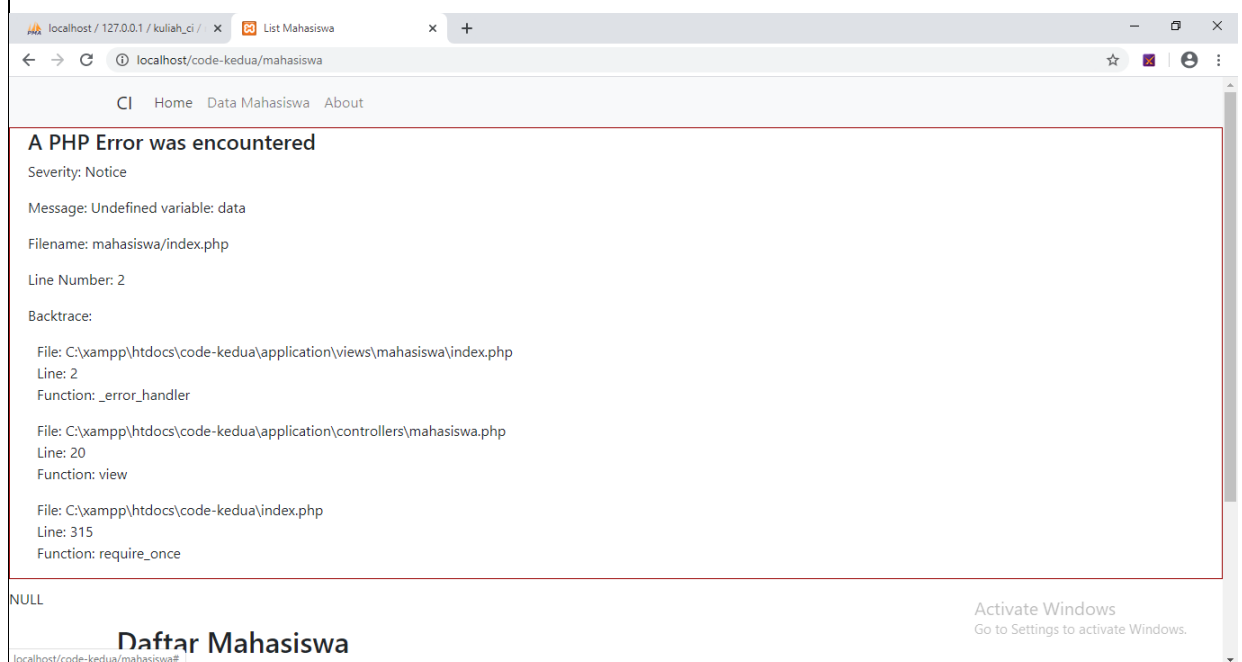
```
application > controllers > mahasiswa.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class mahasiswa extends CI_Controller {
6
7      // public function __construct()
8      // {
9          parent::__construct();
10         $this->load->database();
11     }
12
13     public function index()
14     {
15         $this->load->model('mahasiswa_model');
16         // $this->load->database();
17         $data['title']='List Mahasiswa';
18         $data['mahasiswa']=$this->mahasiswa_model->getAllmahasiswa();
19         $this->load->view('template/header',$data);
20         $this->load->view('mahasiswa/index',$data);
21         $this->load->view('template/footer');
22     }
23
24 }
25
26 /* End of file mahasiswa.php */
27
28 ?>
```

10

Jika kita lakukan `var_dump($mahasiswa)` pada `index.php` pada `views/mahasiswa/index`

```
application > views > mahasiswa > index.php
1  <?php
2  var_dump($mahasiswa);|
3  ?>
4  <div class="container">
5  <div class="row mt-3">
6      <div class="col-md-6">
7          <h2>Daftar Mahasiswa</h2>
8          <ul class="list-group">
9              <li class="list-group-item">Cras justo odio</li>
10             <li class="list-group-item">Dapibus ac facilisis in</li>
11             <li class="list-group-item">Morbi leo risus</li>
12             <li class="list-group-item">Porta ac consectetur ac</li>
13             <li class="list-group-item">Vestibulum at eros</li>
14         </ul>
15     </div>
16 </div>
17 </div>
```

Jika pada `mahasiswa/index` ke `var_dump $data` maka akan terjadi *error*



11

Jika berhasil maka data akan tampil seperti pada gambar berikut

localhost / 127.0.0.1 / kuliah_ci / x List Mahasiswa x +

localhost/code-kedua/mahasiswa

CI Home Data Mahasiswa About

```
array(3) { [0]=> array(5) { ["id"]=> string(1) "1" ["nama"]=> string(4) "dina" ["nim"]=> string(10) "1002345100" ["email"]=> string(14) "dina@gmail.com" ["jurusan"]=> string(5) "kimia" } [1]=> array(5) { ["id"]=> string(1) "2" ["nama"]=> string(4) "dika" ["nim"]=> string(10) "1002345101" ["email"]=> string(14) "dika@gmail.com" ["jurusan"]=> string(11) "informatika" } [2]=> array(5) { ["id"]=> string(1) "3" ["nama"]=> string(4) "soni" ["nim"]=> string(10) "1002345102" ["email"]=> string(14) "soni@gmail.com" ["jurusan"]=> string(5) "mesin" } }
```

Daftar Mahasiswa

Cras justo odio
Dapibus ac facilisis in
Morbi leo risus
Porta ac consectetur ac
Vestibulum at eros

Activate Windows
Go to Settings to activate Windows.

- 12 Agar semua isi dari tabel dapat tampil di halaman index mahasiswa maka kita harus lakukan proses **looping**. Gunakan code berikut pada **views/mahasiswa/index**

```
application > views > mahasiswa > index.php
1  <?php
2  var_dump($mahasiswa);
3  ?>
4  <div class="container">
5  <div class="row mt-3">
6  |   <div class="col-md-6">
7  |   |   <h2>Daftar Mahasiswa</h2>
8  |   |   <ul class="list-group">
9  |   |   |   <?php foreach($mahasiswa as $mhs):?>
10 |   |   |   |   <li class="list-group-item"><? = $mhs['nama'];?></li>
11 |   |   |   <?php endforeach; ?>
12 |   |   </ul>
13 |   </div>
14 </div>
15 </div>
```



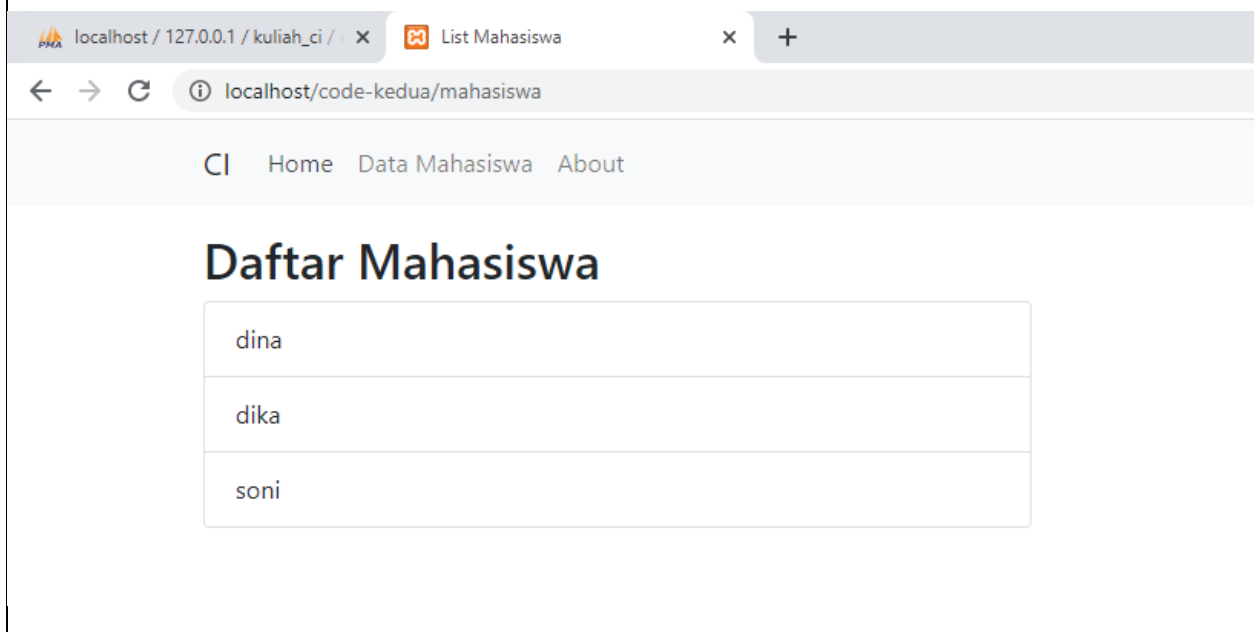
```

application > views > mahasiswa > index.php
1  <!-- <?php
2  var_dump($mahasiswa);
3  ?> -->
4  <div class="container">
5  <div class="row mt-3">
6      <div class="col-md-6">
7          <h2>Daftar Mahasiswa</h2>
8          <ul class="list-group">
9              <?php foreach($mahasiswa as $mhs):?>
10                 <li class="list-group-item"><?= $mhs['nama'];?></li>
11             <?php endforeach; ?>
12          </ul>
13      </div>
14  </div>
15  </div>

```

13

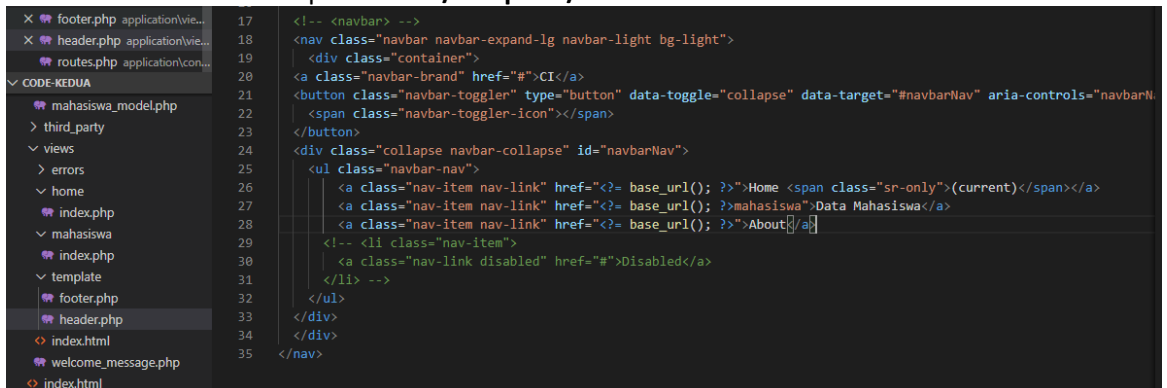
Jika berhasil maka tampilan akan seperti pada gambar dibawah ini



14

Jika kita klik Home, halaman Home belum dapat tampil. Maka dari itu kita akan set agar semua dapat saling terhubung. Untuk itu kita membutuhkan yang namanya **base_url**. untuk lebih jelasnya ikuti langkah-langkah berikut:

a. Tambahkan *code* berikut pada **views/template/header**



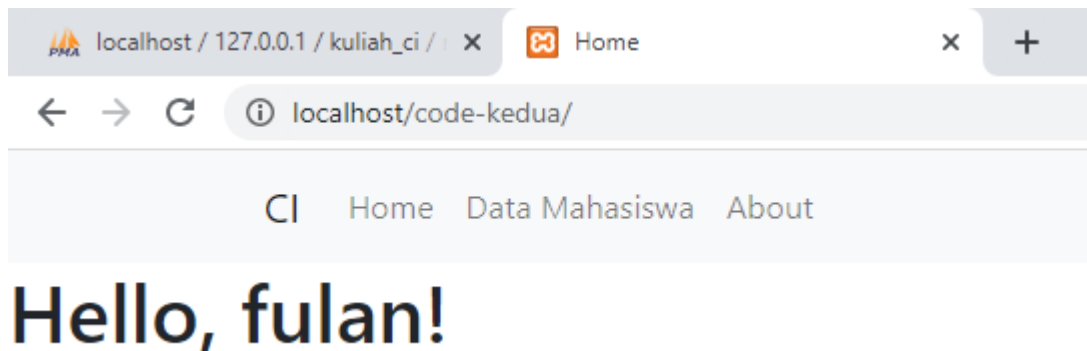
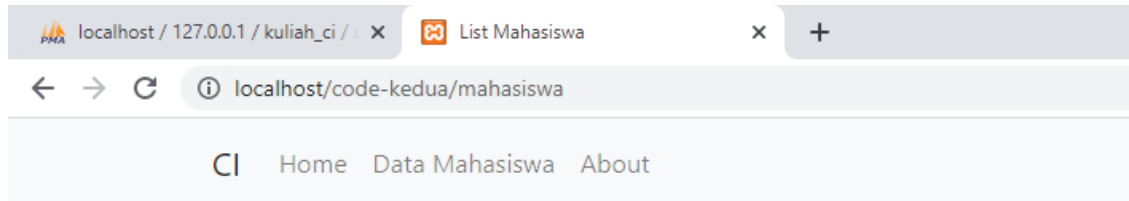
b. Agar **base_url** dikenali oleh sistem maka tambahkan konfigurasi pada **config/autoload.php** tambahkan url pada **\$autoload['helper']=array('url')**

```

89 |
90 | $autoload['helper'] = array('url', 'file');
91 | */
92 | $autoload['helper'] = array('url');
93 |
94 | /*
95 | -----
96 | Auto-load Config files
97 | -----
98 | Prototype:

```

- c. Jika berhasil maka semua halaman dapat ditampilkan pada saat *user* melakukan *event click*.



Praktikum – Bagian 2 : Membuat Form Input dan Menambahkan Form Validation

Langkah	Keterangan
1	Buatlah sebuah <i>button</i> tambah data pada <code>views/mahasiswa/index.php</code>

	<div data-bbox="229 94 1520 880" data-label="Code-Block"> <pre> Home.php index.php ...\home index.php ...\mahasiswa X mahasiswa.php database.php application > views > mahasiswa > index.php 1 <!-- <?php 2 var_dump(\$mahasiswa); 3 ?> --> 4 <div class="container"> 5 <div class="row mt-4"> 6 <div class="col-md-6"> 7 Tambah Data 8 </div> 9 </div> 10 <div class="row mt-3"> 11 <div class="col-md-6"> 12 <h2>Daftar Mahasiswa</h2> 13 <ul class="list-group"> 14 <?php foreach(\$mahasiswa as \$mhs):?> 15 <li class="list-group-item"><?= \$mhs['nama'];?> 16 <?php endforeach; ?> 17 18 </div> 19 </div> 20 </div> </pre> </div>
2	<p>Dikarenakan mahasiswa_model ini nantinya akan digunakan oleh semua <i>function</i> di <i>controller</i> mahasiswa, maka “<code>\$this->load->model('mahasiswa_model');</code>” yang awalnya hanya berada pada function index, kita ubah ke function construct agar semua <i>function</i> dapat menggunakan mahasiswa_model.</p> <div data-bbox="229 1041 1520 1973" data-label="Code-Block"> <pre> application > controllers > mahasiswa.php 1 <?php 2 3 defined('BASEPATH') OR exit('No direct script access allowed'); 4 5 class mahasiswa extends CI_Controller { 6 7 public function __construct() 8 { 9 parent::__construct(); 10 \$this->load->model('mahasiswa_model'); 11 } 12 13 public function index() 14 { 15 // \$this->load->model('mahasiswa_model'); 16 // \$this->load->database(); 17 \$data['title']='List Mahasiswa'; 18 \$data['mahasiswa']=\$this->mahasiswa_model->getAllmahasiswa(); 19 \$this->load->view('template/header',\$data); 20 \$this->load->view('mahasiswa/index',\$data); 21 \$this->load->view('template/footer'); 22 } 23 24 } 25 26 /* End of file mahasiswa.php */ 27 28 ?> </pre> </div>
3	<p>Setelah itu tambahkan function tambah pada controller mahasiswa dan tambahkan <i>code</i> berikut.</p>

```
4 controllers 23
  Home.php 24
  index.html 25
  mahasiswa.php 26
  Welcome.php 27
  core 28
  helpers 29
  hooks 30
  language 31
  libraries 32
  logs 33
  models 34

public function tambah() {
    $data['title'] = 'Form Menambahkan Data Mahasiswa';
    $this->load->view('template/header', $data);
    $this->load->view('mahasiswa/tambah', $data);
    $this->load->view('template/footer');
}

/* End of file mahasiswa.php */
?>
```

4

```
application > views > mahasiswa > tambah.php
1 <div class="container">
2     <div class="row mt-3">
3         <div class="col-md-6">
4             <div class="card">
5                 <div class="card-header">
6                     Form Tambah Data Mahasiswa
7                 </div>
8                 <div class="card-body">
9                     <form action="" method="post">
10                         <div class="form-group">
11                             <label for="nama">Nama</label>
12                             <input type="text" class="form-control" id="nama">
13                         </div>
14                         <div class="form-group">
15                             <label for="nim">Nim</label>
16                             <input type="number" class="form-control" id="nim">
17                         </div>
18                         <div class="form-group">
19                             <label for="email">Email</label>
20                             <input type="email" class="form-control" id="email">
21                         </div>
22                         <div class="form-group">
23                             <label for="jurusan">Jurusan</label>
24                             <select class="form-control" id="jurusan" name="jurusan">
25                                 <option selected>---</option>
26                                 <option value="1">Kimia</option>
27                                 <option value="2">Informatika</option>
28                                 <option value="3">Mesin</option>
29                             </select>
30                         </div>
31                         <button type="submit" name="submit" class="btn btn-primary float-right">Submit</button>
32                     </form>
```

```

application > views > mahasiswa > tambah.php
0      <div>
1          <div class="card-body">
2              <form action="" method="post">
3                  <div class="form-group">
4                      <label for="nama">Nama</label>
5                      <input type="text" class="form-control" id="nama">
6                  </div>
7                  <div class="form-group">
8                      <label for="nim">Nim</label>
9                      <input type="number" class="form-control" id="nim">
10                 </div>
11                 <div class="form-group">
12                     <label for="email">Email</label>
13                     <input type="email" class="form-control" id="email">
14                 </div>
15                 <div class="form-group">
16                     <label for="jurusan">Jurusan</label>
17                     <select class="form-control" id="jurusan" name="jurusan">
18                         <option selected>----</option>
19                         <option value="1">Kimia</option>
20                         <option value="2">Informatika</option>
21                         <option value="3">Mesin</option>
22                     </select>
23                 </div>
24                 <button type="submit" name="submit" class="btn btn-primary float-right">Submit</button>
25             </form>
26         </div>
27     </div>
28 </div>
29 </div>

```

Activate Windows
Go to Settings to activate

```

24     <select class="form-control" id="jurusan" name="jurusan">
25         <option selected>----</option>
26         <option value="kimia">Kimia</option>
27         <option value="informatika">Informatika</option>
28         <option value="mesin">Mesin</option>
29     </select>

```

5 Maka akan dihasilkan halaman seperti di bawah ini.

The screenshot shows a web browser window with the following details:

- Address Bar:** localhost / 127.0.0.1 / kuliah_ci / x Form Menambahkan Data Mahas: x +
- Navigation Bar:** CI Home Data Mahasiswa About
- Form Title:** Form Tambah Data Mahasiswa
- Form Fields:**
 - Nama:** Text input field.
 - Nim:** Text input field.
 - Email:** Text input field.
 - Jurusan:** Dropdown menu with '----' selected.
- Submit Button:** A blue button labeled 'Submit'.

6 Setelah berhasil membuat *form* tambah data, selanjutnya buka file **index** pada **views/mahasiswa/index.php** dan kita atur **href** nya

```

application > views > mahasiswa > index.php
1  <!-- <?php
2  var_dump($mahasiswa);
3  ?> -->
4  <div class="container">
5      <div class="row mt-4">
6          <div class="col-md-6">
7              <a href="<?= base_url();?>mahasiswa/tambah" class="btn btn-primary">Tambah Data</a>
8          </div>
9      </div>

```

7

Tambahkan validasi menggunakan `form_validation`. Contoh validasi *form* adalah seperti *field* nama harus diisi, nim harus angka dsb. Untuk lebih jelasnya ikuti langkah berikut ini :

a. Tambahkan `form_validation`

https://www.codeigniter.com/user_guide/libraries/form_validation.html?highlight=form#the-form (searching controller)

b. Buka *controller* mahasiswa dan tambahkan `form_validation`

```

application > controllers > mahasiswa.php
13  public function index()
14  {
15      // $this->load->model('mahasiswa_model');
16      // $this->load->database();
17      $data['title']='List Mahasiswa';
18      $data['mahasiswa']=$this->mahasiswa_model->getAllmahasiswa();
19      $this->load->view('template/header',$data);
20      $this->load->view('mahasiswa/index',$data);
21      $this->load->view('template/footer');
22  }
23
24  public function tambah() {
25      $data['title']='Form Menambahkan Data Mahasiswa';
26
27      if($this->form_validation->run()==FALSE) {
28          $this->load->view('template/header',$data);
29          $this->load->view('mahasiswa/tambah',$data);
30          $this->load->view('template/footer');
31      }
32      else {
33          echo "Data berhasil ditambah";
34      }
35  }
36
37  }
38
39  /* End of file mahasiswa.php */
40
41  ?>

```

c. Setelah kita tambahkan *form validation*, maka langkah berikutnya menambahkan *rule*-nya

d. https://www.codeigniter.com/user_guide/libraries/form_validation.html?highlight=form#the-form (searching pipe data)

e. Untuk melihat jenis-jenis *rule* bisa dilihat pada **rule reference**

f. `$this->form_validation-`

`>set_rules('fieldname','fieldlabel','trim|required|min_length[5]|max_length[12]');`

g. Jika kita lihat untuk menambahkan *rules* ada 3 parameter yaitu

- fieldname = nama dari htmlnya, semisal `<input type="text" class="form-control" id="nama" name="nama">` maka yang disebut dengan *fieldname* adalah **nama**
- fieldlabel = ini adalah alias digunakan untuk menampilkan kesalahannya (jika terjadi *error*)
- Rule* yang digunakan

h. Sebelum kita tambahkan *rule*-nya maka terlebih dahulu tambahkan **atribut name** pada **views/mahasiswa/tambah**

```

application > views > mahasiswa > tambah.php
8      <div class="card-body">
9          <form action="" method="post">
10             <div class="form-group">
11                 <label for="nama">Nama</label>
12                 <input type="text" class="form-control" id="nama" name="nama">
13             </div>
14             <div class="form-group">
15                 <label for="nim">Nim</label>
16                 <input type="number" class="form-control" id="nim" name="nim">
17             </div>
18             <div class="form-group">
19                 <label for="email">Email</label>
20                 <input type="email" class="form-control" id="email" name="email">
21             </div>
22             <div class="form-group">
23                 <label for="jurusan">Jurusan</label>
24                 <select class="form-control" id="jurusan" name="jurusan">
25                     <option selected>---</option>
26                     <option value="kimia">Kimia</option>
27                     <option value="informatika">Informatika</option>
28                     <option value="mesin">Mesin</option>
29                 </select>
30             </div>
31             <button type="submit" name="submit" class="btn btn-primary float-right">Submit</button>
32         </form>
33     </div>

```

- i. Setelah kita tambahkan **atribut name** maka tambahkan **rule** pada **controller/mahasiswa** seperti pada gambar dibawah ini

```

application > controllers > mahasiswa.php
24     public function tambah() {
25         $data['title']='Form Menambahkan Data Mahasiswa';
26
27         $this->form_validation->set_rules(['nama', 'Nama', 'required'])
28
29         if($this->form_validation->run()==FALSE) {
30             $this->load->view('template/header',$data);
31             $this->load->view('mahasiswa/tambah',$data);
32             $this->load->view('template/footer');
33         }
34         else {
35             echo "Data berhasil ditambah";
36         }
37     }
38
39 }
40
41 /* End of file mahasiswa.php */
42
43 ?>

```

- j. Tambahkan pesan **error** pada **views/mahasiswa/tambah**

```

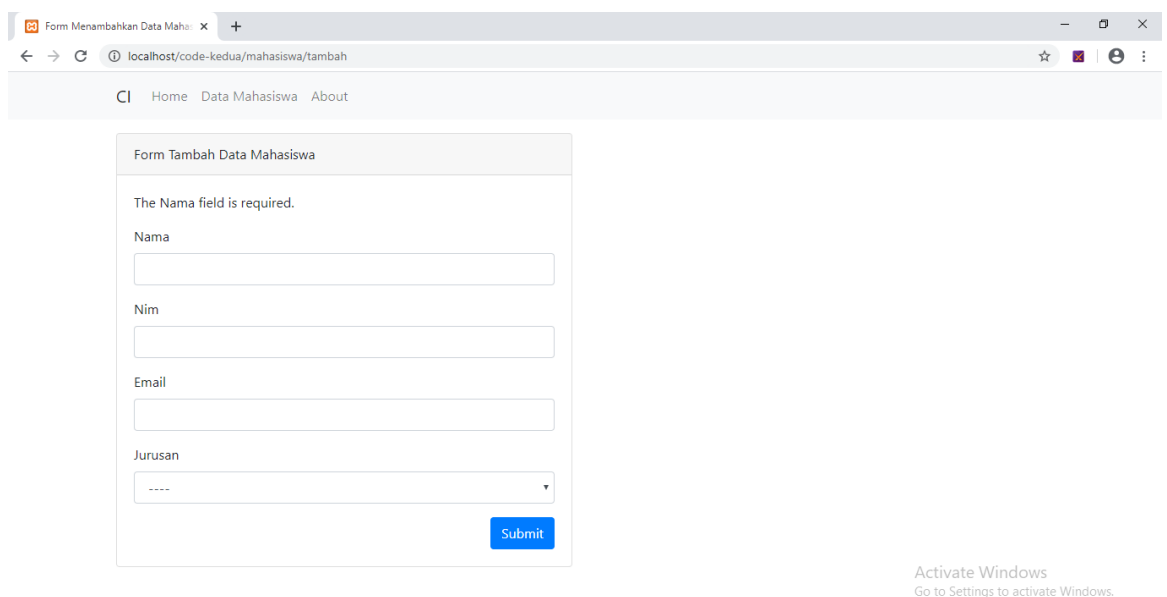
application > views > mahasiswa > tambah.php
1      <div class="container">
2          <div class="row mt-3">
3              <div class="col-md-6">
4                  <div class="card">
5                      <div class="card-header">
6                          Form Tambah Data Mahasiswa
7                      </div>
8                      <div class="card-body">
9                          <?= validation_errors() ?>
10                     <form action="" method="post">
11                         <div class="form-group">
12                             <label for="nama">Nama</label>
13                             <input type="text" class="form-control" id="nama" name="nama">
14                         </div>
15                         <div class="form-group">
16                             <label for="nim">Nim</label>
17                             <input type="number" class="form-control" id="nim" name="nim">
18                         </div>

```

- k. Tambahkan query untuk memuat library 'form_validation' pada function __construct()

```
application > controllers > mahasiswa.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class mahasiswa extends CI_Controller {
6
7      public function __construct()
8      {
9          parent::__construct();
10         $this->load->library('form_validation');
11         $this->load->model('mahasiswa_model');
12     }
```

- l. Hasilnya saat *field* nama tidak diisi maka akan muncul keterangan the Nama *Field is required*



Form Menambahkan Data Maha: x +

localhost/code-kedua/mahasiswa/tambah

CI Home Data Mahasiswa About

Form Tambah Data Mahasiswa

The Nama field is required.

Nama

Nim

Email

Jurusan

Submit

Activate Windows
Go to Settings to activate Windows.

- m. Tambahkan *alert* agar pesan *error* lebih menarik setelah *code validation error*

```
application > views > mahasiswa > tambah.php
1  <div class="container">
2      <div class="row mt-3">
3          <div class="col-md-6">
4              <div class="card">
5                  <div class="card-header">
6                      Form Tambah Data Mahasiswa
7                  </div>
8                  <div class="card-body">
9                      <?php if(validation_errors()): ?>
10                     <div class="alert alert-danger" role="alert">
11                         <?= validation_errors() ?>
12                     </div>
13                     <?php endif ?>
14                     <form action="" method="post">
15                         <div class="form-group">
16                             <label for="nama">Nama</label>
17                             <input type="text" class="form-control" id="nama" name="nama">
18                         </div>
```

- n. Akan dihasilkan seperti gambar di bawah ini

Form Menambahkan Data Maha: x +

localhost/code-kedua/mahasiswa/tambah

CI Home Data Mahasiswa About

Form Tambah Data Mahasiswa

The Nama field is required.

Nama

Nim

Email

Jurusan

Submit

Activate Windows
Go to Settings to activate Windows.

- o. Setelah kita berhasil memberikan **rules validation** pada **nama**, maka tambahkan juga untuk **nim** dan **email**.

```
application > controllers > mahasiswa.php
25 public function tambah() {
26     $data['title']='Form Menambahkan Data Mahasiswa';
27
28     // $this->form_validation->set_rules('fieldname','fieldlabel','trim|required|min_length[5]|max_length[12]');
29     $this->form_validation->set_rules('nama', 'Nama', 'required');
30     $this->form_validation->set_rules('nim', 'Nim', 'required');
31     $this->form_validation->set_rules('email', 'Email', 'required');
32 }
```

- p. Hasilnya seperti berikut ini.

Form Menambahkan Data Maha: x +

localhost/code-kedua/mahasiswa/tambah

CI Home Data Mahasiswa About

Form Tambah Data Mahasiswa

The Nama field is required.
The Nim field is required.
The Email field is required.

Nama

Nim

Email

Jurusan

Submit

Activate Windows
Go to Settings to activate Windows.

- q. Kita juga dapat mengecek ke **valid** an email dengan menambahkan rule pada email seperti pada gambar dibawah ini menggunakan **valid_email**

```

application > controllers > mahasiswa.php
25 public function tambah() {
26     $data['title']='Form Menambahkan Data Mahasiswa';
27
28     // $this->form_validation->set_rules('fieldname','fieldlabel','trim|required|min_length[5]|max_length[12]');
29     $this->form_validation->set_rules('nama', 'Nama', 'required');
30     $this->form_validation->set_rules('nim', 'Nim', 'required');
31     $this->form_validation->set_rules('email', 'Email', 'required|valid_email');
32

```

r. Hasilnya sebagai berikut.

Form Menambahkan Data Mahasiswa

The Nama field is required.
The Nim field is required.
The Email field is required.

Nama
dina

Nim
12344512323

Email
dina

Jurusan

Submit

Activate Windows
Go to Settings to activate Windows.

s. Selain itu kita juga dapat memberikan *rules* harus diisi dengan angka menggunakan *rules numeric*

```

application > controllers > mahasiswa.php
25 public function tambah() {
26     $data['title']='Form Menambahkan Data Mahasiswa';
27
28     // $this->form_validation->set_rules('fieldname','fieldlabel','trim|required|min_length[5]|max_length[12]');
29     $this->form_validation->set_rules('nama', 'Nama', 'required');
30     $this->form_validation->set_rules('nim', 'Nim', 'required|numeric');
31     $this->form_validation->set_rules('email', 'Email', 'required|valid_email');
32

```

t. Akan dihasilkan alert seperti berikut ini.

Form Menambahkan Data Mahasiswa

The Nama field is required.
The Nim field is required.
The Email field is required.

Nama
dina

Nim
123ee

Email
123ee

Jurusan

Submit

Activate Windows
Go to Settings to activate Windows.

Praktikum – Bagian 3 : Menambahkan Data ke Database / Insert Data

Langkah	Keterangan
1	<p>Buka controller/mahasiswa dan tambahkan <i>code</i> berikut pada function tambah agar pada saat form_validation bernilai true, maka akan ditambahkan data dengan nama modelnya -> model_mahasiswa dengan nama <i>function</i> tambahdatamhs() dan jika sudah ditambah ke database maka akan <i>redirect</i> ke <i>controller</i> mahasiswa</p> <pre> application > controllers > mahasiswa.php 25 public function tambah() { 26 \$data['title']='Form Menambahkan Data Mahasiswa'; 27 28 // \$this->form_validation->set_rules('fieldname','fieldlabel','trim required min_length[5] max_length[12]'); 29 \$this->form_validation->set_rules('nama', 'Nama', 'required'); 30 \$this->form_validation->set_rules('nim', 'Nim', 'required numeric'); 31 \$this->form_validation->set_rules('email', 'Email', 'required valid_email'); 32 33 if(\$this->form_validation->run()==FALSE) { 34 \$this->load->view('template/header',\$data); 35 \$this->load->view('mahasiswa/tambah',\$data); 36 \$this->load->view('template/footer'); 37 } 38 else { 39 \$this->mahasiswa_model->tambahdatamhs(); 40 redirect('mahasiswa','refresh'); 41 } 42 } 43 44 }</pre>
2	<p>Buatlah sebuah <i>function</i> tambahdatamhs() pada models/mahasiswa_model</p> <ul style="list-style-type: none"> Tambahkan <i>code</i> seperti pada gambar dibawah ini <pre> application > models > mahasiswa_model.php 13 public function tambahdatamhs() { 14 \$data=[15 "nama" => \$this->input->post('nama',true), 16 "nim" => \$this->input->post('nim',true), 17 "email" => \$this->input->post('email',true), 18 "jurusan" => \$this->input->post('jurusan',true) 19]; 20 \$this->db->insert('mahasiswa',\$data); 21 } 22 23 24 25 /* End of file mahasiswa_model.php */ 26 27 ?></pre>
3	<p>Langkah berikutnya kita coba untuk menambahkan data. Jika berhasil maka data akan tampil di halaman index.php</p> <ol style="list-style-type: none"> <code>redirect('mahasiswa','refresh');</code>(line 47 gambar <i>controller</i> mahasiswa) Hasil

Form Menambahkan Data Mahasiswa

CI Home Data Mahasiswa About

Form Tambah Data Mahasiswa

Nama
ani

Nim
12349888

Email
ani@gmail.com

Jurusan
Mesin

Submit

Activate Windows
Go to Settings to activate Windows.

List Mahasiswa

CI Home Data Mahasiswa About

Tambah Data

Daftar Mahasiswa

dina
dika
soni
ani

