



Nama : Mirza Zarqani Rayhan
 Kelas : TI-2A
 NIM : 1841720205

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 <i>controller</i> mahasiswa dan tambahkan <i>code</i> berikut (line 9)</p> <pre> 14 public function index() 15 { 16 //modul untuk ngeload database 17 \$this->load->database(); 18 \$data['title']='List Mahasiswa'; 19 \$this->load->view('template/header',\$data); 20 \$this->load->view('mahasiswa/index'); 21 \$this->load->view('template/footer'); 22 } 23 } 24 } 25 } 26 </pre> <p>Keterangan :</p> <ul style="list-style-type: none"> - Pada line 9 menjelaskan bawa <i>function</i> index ini akan melakukan <i>load</i> database. <i>Load</i> database ini hanya dilakukan di <i>function</i> index saja jika ada <i>function</i> lain maka kita harus melakukan <i>load</i> database lagi. Sehingga hal ini dirasa tidak praktis maka untuk itu kita harus tambahkan <i>code</i> agar <i>load</i> database ini dapat digunakan banyak <i>function</i>.
3	Tambahkan <i>code</i> berikut agar database dapat digunakan banyak <i>function</i> (line 7-12)

```

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              //digunakan untuk dijalankan saat di instansiasi
10             parent::__construct();
11             $this->load->database();
12
13         }
14         public function index()
15         {
16             //modul untuk ngeload database
17             $this->load->database();
18             $data['title']='List Mahasiswa';
19             $this->load->view('template/header',$data);
20             $this->load->view('mahasiswa/index');
21             $this->load->view('template/footer');
22
23         }
24     }
25
26     /* End of file Mahasiswa.php */
27
28
29

```

Keterangan:

- Line 7 – 12 adalah sebuah *function* construct (**PERHATIKAN ITU MENGGUNAKAN DOUBLE UNDERSCORE __construct (tanpa spasi)**)
- *Function* construct dijalankan pada saat classnya diinstansiasi
- Line 10 digunakan untuk menjalankan fungsi construct pada class parent dimana class parent ini adalah CI_Controller
- Line 11 digunakan agar untuk mengload database (Semua fungsi di class mahasiswa dapat menggunakannya)
- Jika kita sudah melakukan load database pada saat *function* construct maka pada *function* index kita comment agar tidak bekerja dua kali

- 4 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

```

51
52 | Prototype:
53 |
54 | $autoload['libraries'] = array('database', 'email', 'session');
55 |
56 | You can also supply an alternative library name to be assigned
57 | in the controller:
58 |
59 | $autoload['libraries'] = array('user_agent' => 'ua');
60 |
61 | $autoload['libraries'] = array('database');
62

```

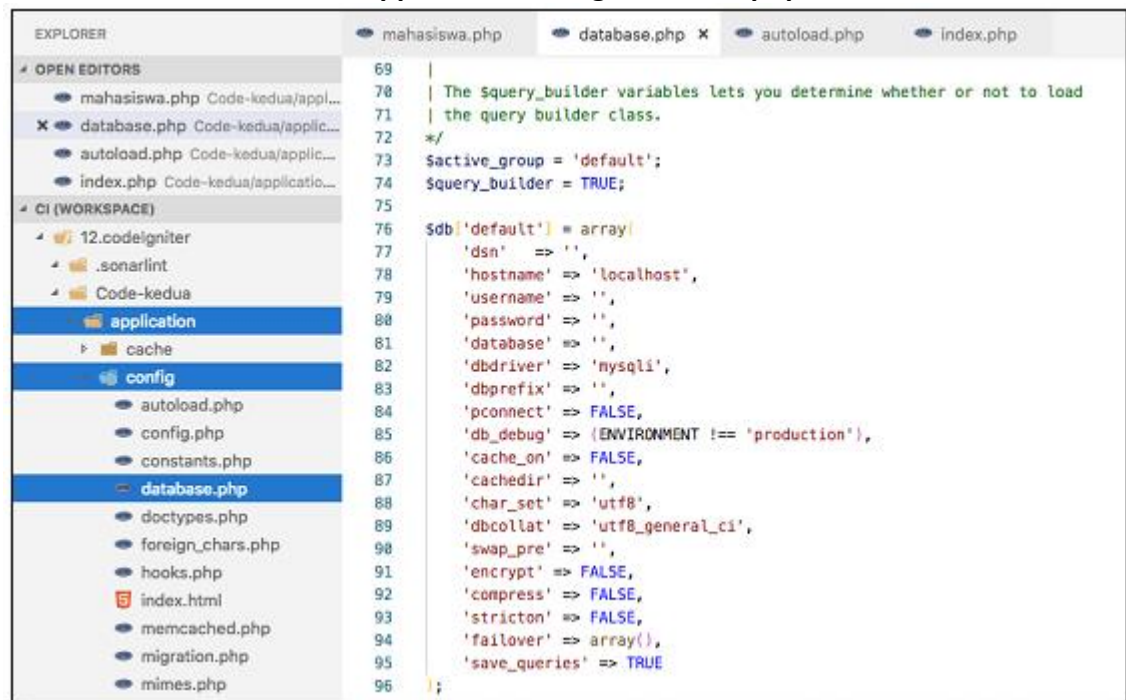
- 5 Untuk kasus ini kita gunakan cara yang terakhir (no 4) karena kita asumsikan bahwa semua *controller* membutuhkan database. Maka kita *comment code* function construct yang ada di *controller* mahasiswa

```

5      class Mahasiswa extends CI_Controller {
6
7          // public function __construct()
8          //{
9              //digunakan untuk dijalankan saat di instansiasi
10             //parent::__construct();
11             //$this->load->database();
12
13         //}

```

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



- 7 Kita set konfigurasinya seperti pada gambar dibawah ini

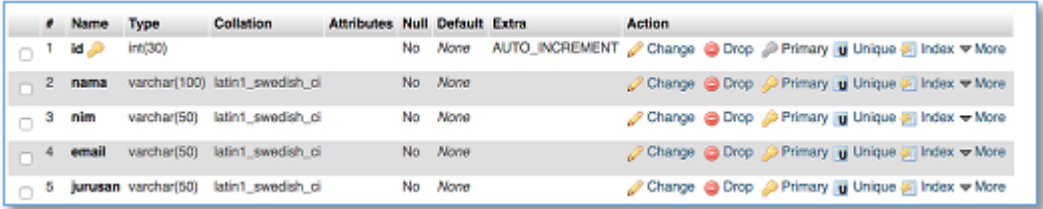
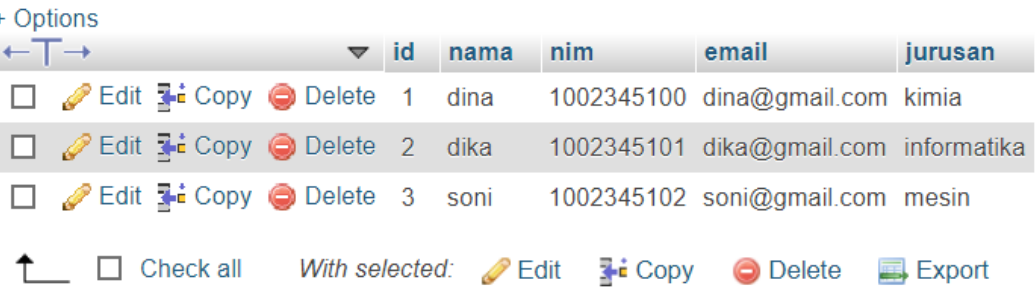
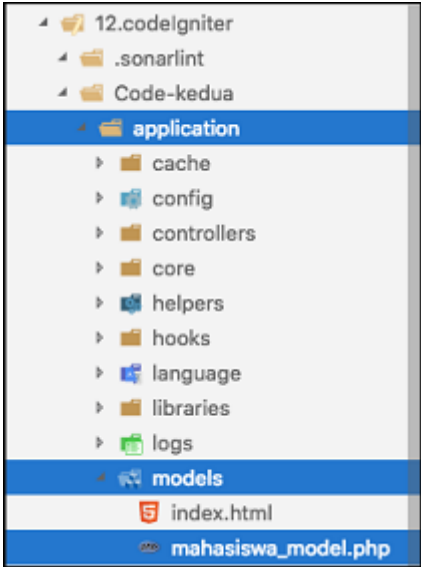
```

73 $active_group = 'default';
74 $query_builder = TRUE;
75
76 $db['default'] = array(
77     'dsn' => '',
78     'hostname' => 'localhost',
79     'username' => 'root',
80     'password' => 'mysql',
81     'database' => 'kuliah_CI',
82     'dbdriver' => 'mysql',
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 );

```

Keterangan :

- Line 78 artinya *hostname* yang digunakan adalah localhost

	<ul style="list-style-type: none"> - <i>Line 79</i> artinya <i>username</i> (phpmyadmin) yang digunakan apa - <i>Line 80</i> artinya <i>password</i> (phpmyadmin) yang digunakan apa - <i>Line 81</i> artinya nama database yang nanti digunakan
8	<p>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</p> 
9	<p>Tambahkan beberapa data ke dalam tabel mahasiswa seperti pada gambar (pada contoh terdapat 3 data, jumlah data lebih banyak tidak masalah)</p> 
10	<p>Setelah kita membuat sebuah <i>database</i>, maka langkah berikutnya adalah membuat sebuah model untuk melakukan proses <i>database</i>. Di dalam model ini kita akan membuat sebuah <i>function</i> untuk melakukan perintah ambil data dari <i>database</i>. Untuk lebih jelasnya ikuti langkah-langkah berikut:</p> <p>a. Buatlah sebuah file model dengan nama mahasiswa_model pada folder application/models</p>  <p>b. Tambahkan <i>function</i> getAllmahasiswa pada mahasiswa_model.php</p>

```

mahasiswa.php  mahasiswa_model.php  index.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          // https://www.codeigniter.com/user_guide/database/query_builder.html?highlight=query%20builder
10         // $query digunakan untuk menampung isi dari tabel mahasiswa
11         $query=$this->db->get('mahasiswa');
12
13         // https://www.codeigniter.com/user_guide/database/results.html
14         // untuk menampilkan isi dari query
15         return $query->result_array();
16     }
17
18
19 }
20
21 /* End of file mahasiswa_model.php */
22
23 ?>

```

Keterangan :

- Pada *line 11* membuat sebuah *query builder* yang digunakan untuk mengambil semua data
- https://www.codeigniter.com/user_guide/database/query_builder.html#selecting-data
- Pada *line 15* digunakan untuk mengenerate hasil query dalam bentuk array
- https://www.codeigniter.com/user_guide/database/results.html
- https://www.codeigniter.com/user_guide/database/results.html#result-arrays

c. Selain menggunakan query builder kita juga bisa menggunakan cara lain seperti gambar dibawah ini

```

1  <?php
2
3      defined('BASEPATH') OR exit('No direct script access allowed');
4
5      class mahasiswa_model extends CI_Controller {
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     ?>

```

11

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

```

13 //}
14 public function index()
15 {
16     $this->load->model('mahasiswa_model');
17
18     //modul untuk ngeload database
19     //$this->load->database();
20     $data['title']='List Mahasiswa';
21     $data['mahasiswa']=$this->mahasiswa_model->getAllMahasiswa();
22     $this->load->view('template/header',$data);
23     $this->load->view('mahasiswa/index');
24     $this->load->view('template/footer');
25 }
26 }
27

```

Keterangan :

- *Line 17* digunakan untuk melakukan *load* model dengan nama modelnya adalah **mahasiswa_model**
- *Line 22* artinya ada data yang berupa array dengan isi arraynya adalah mahasiswa dimana terdapat isi berupa function **getAllmahasiswa()** yang ada pada model dengan nama **mahasiswa_model**
- *Line 24* tambahkan **\$data**

```
mahasiswa.php  mahasiswa_model.php  index.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
10
11         // atau bisa juga menggunakan code berikut
12         return $this->db->get('mahasiswa')->result_array();
13     }
14
15 }
16
17 /* End of file mahasiswa_model.php */
18
19 ?>
```

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

12 Jika kita lakukan var_dump(\$mahasiswa) pada **index.php** pada **views/mahasiswa/index**

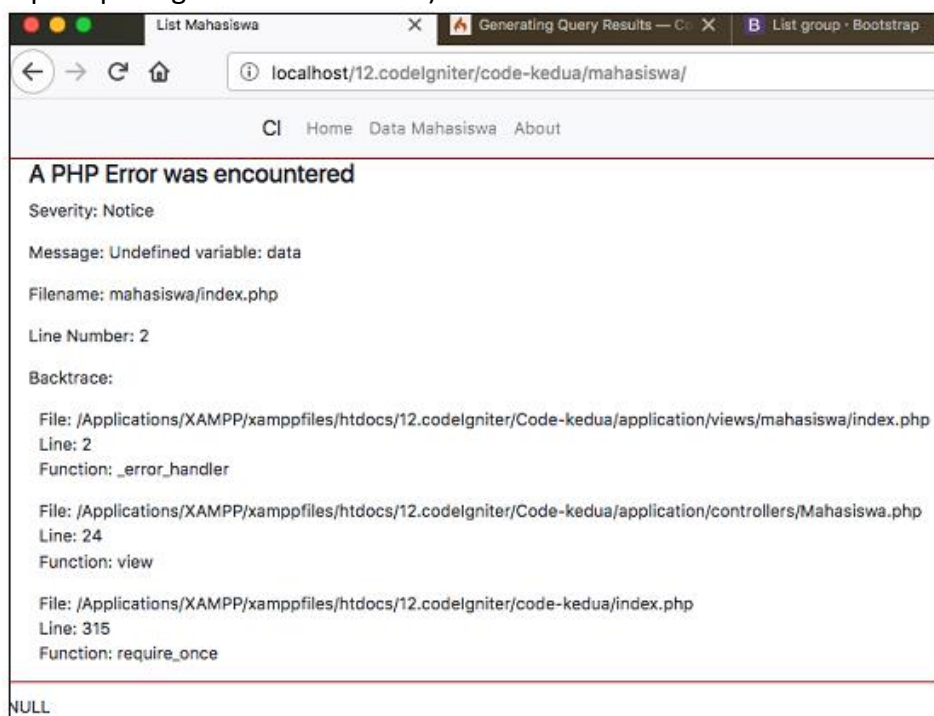
```

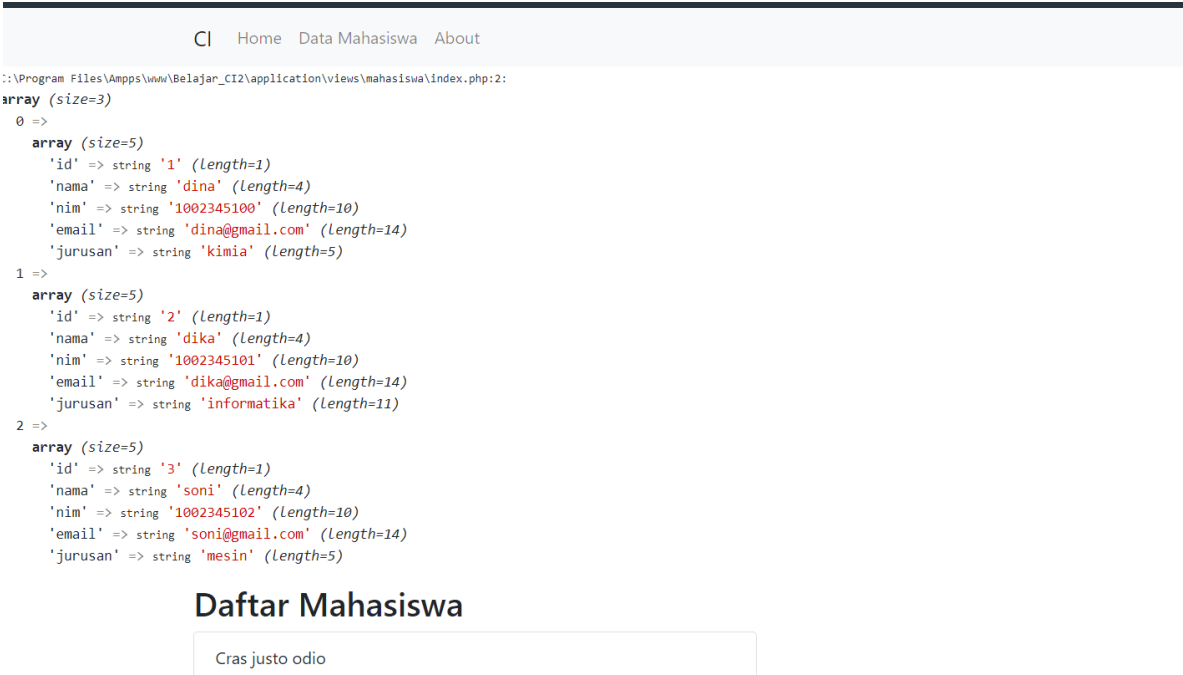
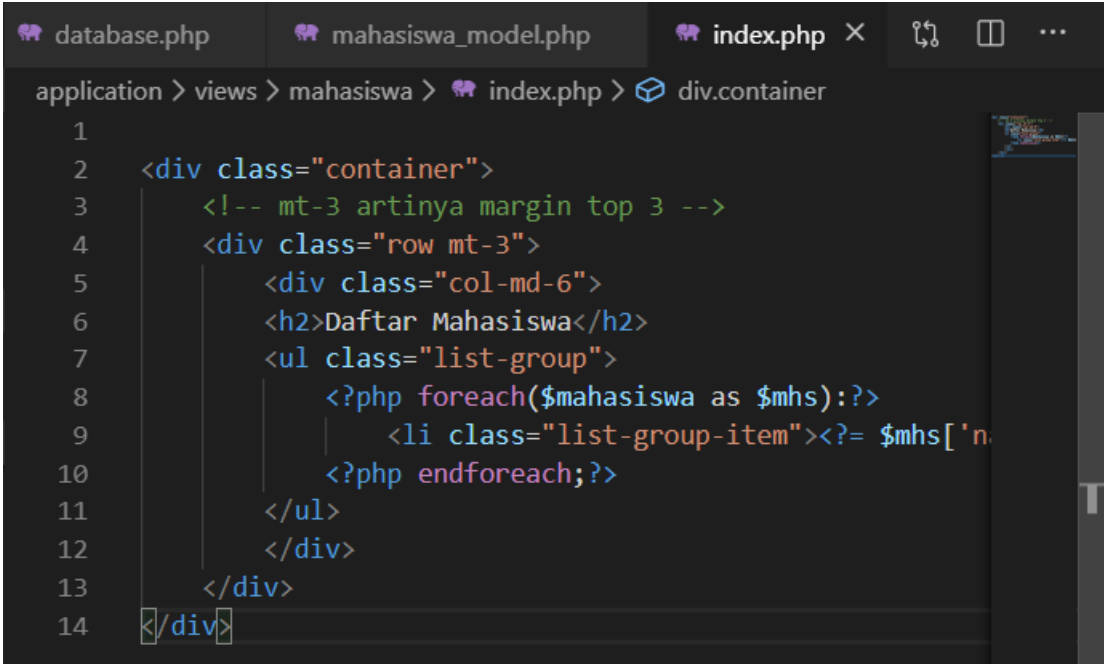
1 <?php
2     var_dump($mahasiswa);
3 ?>
4 <div class="container">
5     <!-- mt-3 artinya margin top 3 -->
6     <div class="row mt-3">
7         <div class="col-md-6">
8             <h2>Daftar Mahasiswa</h2>
9             <ul class="list-group">
10                <li class="list-group-item">Cras justo odio
11                <li class="list-group-item">Dapibus ac faci
12                <li class="list-group-item">Morbi leo risus
13                <li class="list-group-item">Porta ac consec
14                <li class="list-group-item">Vestibulum at e
15            </ul>
16        </div>
17    </div>
18 </div>

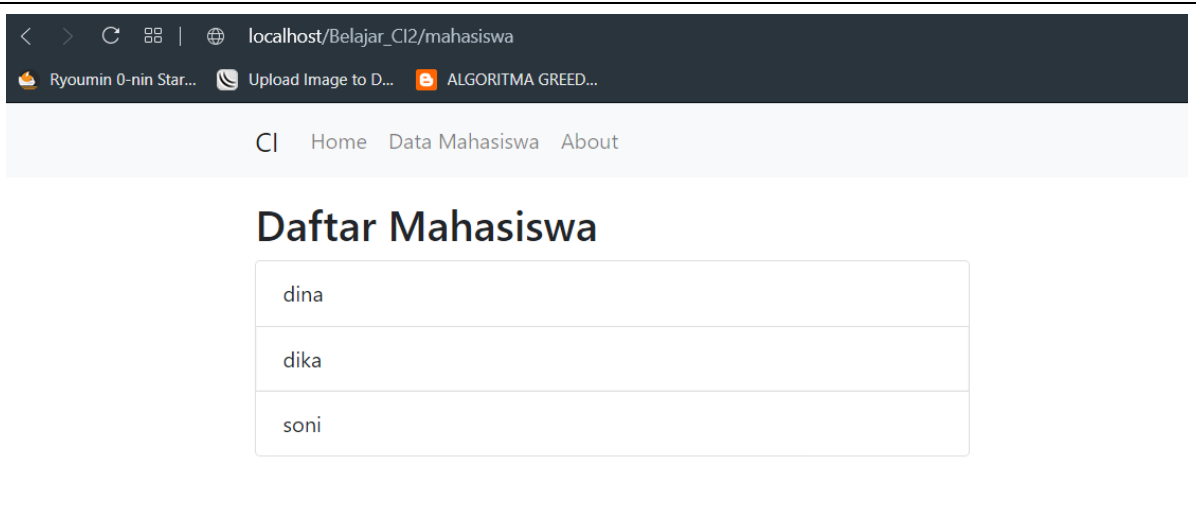
```

Keterangan

- \$mahasiswa didapat dari mana ??? -> jika kita lihat *line 22* pada *controller* mahasiswa terdapat [mahasiswa]
- Jika kita lihat pada *controller* mahasiswa terdapat perintah/code seperti berikut:
 - `$data['mahasiswa']=$this->mahasiswa_model->getAllMahasiswa();` (Code pertama)
 - `$this->load->view('mahasiswa/index',$data);` (Code kedua)
- Jika kita lihat *code* kedua, ada \$data yang isi arraynya adalah mahasiswa dan isi mahasiswa tersebut adalah **function** `getAllMahasiswa` pada model dengan nama mahasiswa_model. Dimana nanti isi dari `$data['mahasiswa']` ini dikirimkan ke halaman **view** dengan **2 paramater** yaitu `mahasiswa/index` dan `$data` (yang isinya mahasiswa, sehingga jika mau `var_dump` di halaman `mahasiswa/index`, kita menggunakan `$mahasiswa`. Jika pada `mahasiswa/index` ke `var_dump $data` maka akan terjadi *error* seperti pada gambar dibawah ini)



	<p>Hal ini disebabkan karena tidak ketahu data pada mahasiswa/index.php pada <i>line</i> 2 berupa <code>var_dump(\$mahasiswa)</code>, maka pastikan terlebih dahulu pada <i>controller</i> mahasiswa.php sudah terdapat <code>\$data</code> di <i>line</i> 13 dan 15.</p>
13	<p>Jika berhasil maka data akan tampil seperti pada gambar berikut</p>  <pre> ::\Program Files\Amps\www\Belajar_CI2\application\views\mahasiswa\index.php:2: array (size=3) 0 => array (size=5) 'id' => string '1' (length=1) 'nama' => string 'dina' (length=4) 'nim' => string '1002345100' (length=10) 'email' => string 'dina@gmail.com' (length=14) 'jurusan' => string 'kimia' (length=5) 1 => array (size=5) 'id' => string '2' (length=1) 'nama' => string 'dika' (length=4) 'nim' => string '1002345101' (length=10) 'email' => string 'dika@gmail.com' (length=14) 'jurusan' => string 'informatika' (length=11) 2 => array (size=5) 'id' => string '3' (length=1) 'nama' => string 'soni' (length=4) 'nim' => string '1002345102' (length=10) 'email' => string 'soni@gmail.com' (length=14) 'jurusan' => string 'mesin' (length=5) </pre> <p>Daftar Mahasiswa</p> <p>Cras justo odio</p>
14	<p>Agar semua isi dari tabel dapat tampil di halaman index mahasiswa maka kita harus lakukan proses looping. Gunakan <i>code</i> berikut pada views/mahasiswa/index</p>  <pre> application > views > mahasiswa > index.php > div.container 1 2 <div class="container"> 3 <!-- mt-3 artinya margin top 3 --> 4 <div class="row mt-3"> 5 <div class="col-md-6"> 6 <h2>Daftar Mahasiswa</h2> 7 <ul class="list-group"> 8 <?php foreach(\$mahasiswa as \$mhs):?> 9 <li class="list-group-item"><?= \$mhs['n 10 <?php endforeach;?> 11 12 </div> 13 </div> 14 </div> </pre>
15	<p>Jika berhasil maka tampilan akan seperti pada gambar dibawah ini</p>



16

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** (line 26,27,28)

```

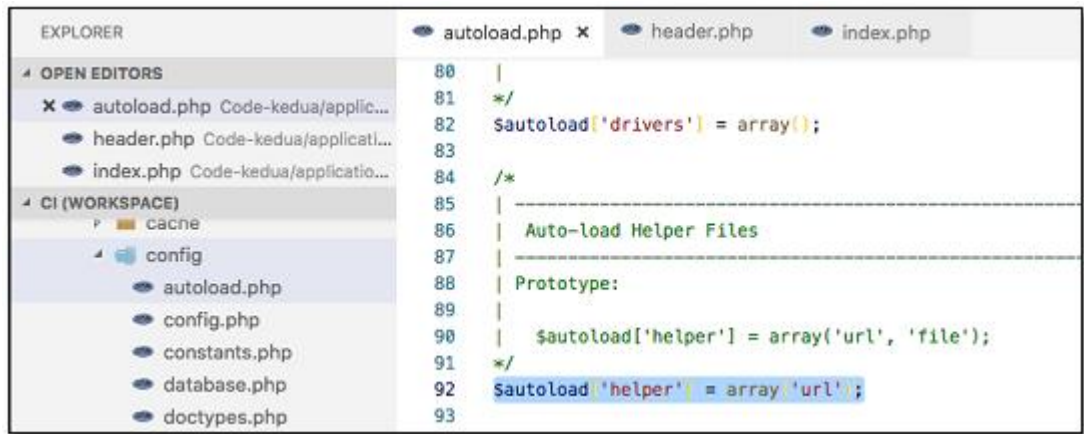
1 <!doctype html>
2 <html lang="en">
3 <head>
4 <!-- Required meta tags -->
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8 <!-- Bootstrap CSS -->
9 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8X4CGs03+H4v8T/QSP
10
11 <title><?=$title?></title>
12 </head>
13 <body>
14 <!--Navbar-->
15 <nav class="navbar navbar-expand-lg navbar-light bg-light">
16 <div class="container">
17 <a class="navbar-brand" href="#">CI</a>
18 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-
19 <span class="navbar-toggler-icon"></span>
20 </button>
21 <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
22 <ul class="navbar-nav">
23 <a class="nav-item nav-link" href="<?=$base_url(); ?>">Home <span class="sr-only">(current)</span></a>
24 <a class="nav-item nav-link" href="<?=$base_url(); ?>mahasiswa">Data Mahasiswa</a>
25 <a class="nav-item nav-link" href="<?=$base_url(); ?>">About</a>
26 <!--<a class="nav-link disabled" href="#" tabindex="-1" aria-disabled="true">Disabled</a-->
27 </ul>
28 </div>
29 </div>
30 </nav>

```

Keterangan :

- Line 26, 27, 28 tambahkan **base_url** (pastikan **config/config.php** sudah diset)
`$config['base_url'] = 'http://localhost/12.codeigniter/code-kedua/';` (code_kedua adalah nama project CI jadi sesuaikan dengan nama project CI kalian)
- Agar halaman web mahasiswa dapat ditampilkan saat *user* mengklik mahasiswa, maka setelah **base_url** tambahkan **controller mahasiswa**

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



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	<p>Buatlah sebuah <i>button</i> tambah data pada views/mahasiswa/index.php (line 3-7). Tambahkan <i>code</i> pada line 3-7 untuk membuat sebuah <i>button</i></p> <pre> 1 <div class="container"> 2 <div class="row mt-4"> 3 <div class="col-md-6"> 4 Tambah Data 5 </div> 6 </div> 7 <!-- mt-3 artinya margin top 3 --> 8 <div class="row mt-3"> 9 <div class="col-md-6"> 10 <h2>Daftar Mahasiswa</h2> 11 <ul class="list-group"> 12 <?php foreach(\$mahasiswa as \$mhs):?> 13 <li class="list-group-item"><?php echo \$mhs['nama'];?> 14 <?php endforeach;?> 15 16 </div> 17 </div> 18 </div> </pre>
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 <i>function index</i>, kita ubah ke function construct agar semua <i>function</i> dapat menggunakan mahasiswa_model.</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 //digunakan untuk dijalankan saat di instansiasi 10 parent::__construct(); 11 \$this->load->model('mahasiswa_model'); 12 13 } 14 public function index() 15 { 16 17 //modul untuk ngeload database 18 //\$this->load->database(); 19 \$data['title']='List Mahasiswa'; 20 \$data['mahasiswa']=\$this->mahasiswa_model->getAllMahasiswa(); 21 \$this->load->view('template/header',\$data); 22 \$this->load->view('mahasiswa/index',\$data); 23 \$this->load->view('template/footer'); 24 25 } 26 27 } 28 29 /* End of file Mahasiswa.php */ 30 31 ?> </pre>
3	<p>Setelah itu tambahkan function tambah pada controller mahasiswa dan tambahkan code berikut.</p> <p>(Tips : jika menggunakan vsCode untuk menduplikat 1 baris dapat menggunakan alt+shift+panah_bawah)</p> <pre> 26 public function tambah() 27 { 28 \$data['title']='Form Menambahkan Data Mahasiswa'; 29 \$this->load->view('template/header',\$data); 30 \$this->load->view('mahasiswa/tambah',\$data); 31 \$this->load->view('template/footer'); 32 } 33 34 } 35 </pre> <p>Keterangan :</p> <ul style="list-style-type: none"> - Pada line 32, \$this->load->view('mahasiswa/tambah') artinya bahwa function tambah akan melakukan <i>load view</i> pada folder mahasiswa, dengan nama filenya adalah tambah
4	<p>Setelah kita membuat function tambah pada controller mahasiswa, selanjutnya buatlah form tambah pada views/mahasiswa/tambah.php</p>

```

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

```

Keterangan :

- Pada *line 33*, terdapat atribut class dengan *value* btn btn-primary float-right yang artinya kita membuat sebuah *button* dan posisi buttonnya ada di kanan
- Ganti ***value*** dari *line 28-31* menjadi seperti di bawah ini (**Agar di tabel tidak tersimpan *value* berupa angka**)

```

25 <select class="form-control" name="jurusan" id="jurusan">
26   <option selected>----</option>
27   <option value="1">Kimia</option>
28   <option value="2">Informatika</option>
29   <option value="3">Mesin</option>
30 </select>

```

5

Maka akan dihasilkan halaman seperti di bawah ini.

6

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

```

1 <div class="container">
2     <div class="row mt-4">
3         <div class="col-md-6">
4             <a href="<?= base_url(); ?>mahasiswa/tambah">Tambah Data</a>
5         </div>
6     </div>
7     <!-- mt-3 artinya margin top 3 -->
8     <div class="row mt-3">
9         <div class="col-md-6">
10            <h2>Daftar Mahasiswa</h2>
11            <ul class="list-group">
12                <?php foreach($mahasiswa as $mhs):?>
13                    <li class="list-group-item"><?= $mhs['nama'];?></li>
14                <?php endforeach;?>
15            </ul>
16        </div>
17    </div>
18 </div>

```

Keterangan :

- Tambahkan *code* pada line 5 agar pada saat *user* melakukan *event click* maka *user* akan ditampilkan halaman tambah.

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*

```

27
28
29
30 public function tambah() {
31
32     $data['title'] = 'Form Menambahkan Data Mahasiswa';
33
34     if ($this->form_validation->run() == FALSE) {
35         # code...
36         $this->load->view('template/header', $data);
37         $this->load->view('mahasiswa/tambah', $data);
38         $this->load->view('template/footer');
39     } else {
40         # code...
41         echo "data berhasil ditambah";
42     }
43 }
44
45 }
46
47 /* End of file Controllername.php */
48 ?>

```

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

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

- Untuk melihat jenis-jenis *rule* bisa dilihat pada **rule reference**
- `$this->form_validation->set_rules('fieldname','fieldlabel','trim|required|min_length[5]|max_length[12]');`
- Jika kita lihat untuk menambahkan *rules* ada 3 parameter yaitu
 1. `fieldname` = nama dari htmlnya, semisal `<input type="text" class="form-control" id="nama" name="nama">` maka yang disebut dengan `fieldname` adalah `nama`
 2. `fieldlabel` = ini adalah alias digunakan untuk menampilkan kesalahannya (jika terjadi *error*)
 3. *Rule* yang digunakan

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

```
<div class="form-group">
    <label for="nama">Nama</label>
    <input type="text" class="form-control" id="nama" name="nama">
</div>
<div class="form-group">
    <label for="nim">Nim</label>
    <input type="number" class="form-control" id="nim" name="nim">
</div>
<div class="form-group">
    <label for="email">email</label>
    <input type="email" class="form-control" id="email" name="email">
</div>
<div class="form-group">
    <label for="jurusan">Jurusan</label>
    <select class="form-control" id="jurusan" name="jurusan">
        <option selected>----</option>
        <option value="1">Kimia</option>
        <option value="2">Informatika</option>
        <option value="3">Mesin</option>
    </select>
</div>
```

e. Setelah kita tambahkan **atribut name** maka tambahkan **rule** pada **controller/mahasiswa** seperti pada gambar dibawah ini (*line 35-46*)

```
30 public function tambah() {
31
32     $data['title'] = 'Form Menambahkan Data Mahasiswa';
33
34     // $this->form_validation->set_rules('fieldname', 'fieldlabel', 'trim|required|min_length[5]|max_length[12]');
35     $this->form_validation->set_rules('nama', 'Nama', 'required');
36
37     if ($this->form_validation->run() == FALSE) {
38         # code...
39         $this->load->view('template/header', $data);
40         $this->load->view('mahasiswa/tambah', $data);
41         $this->load->view('template/footer');
42     } else {
43         # code...
44         echo "data berhasil ditambah";
45     }
46 }
```

f. Tambahkan pesan *error* pada **views/mahasiswa/tambah** (*line 11*)

```

10 <div class="card-body">
11 <?= validation_errors() ?>
12 <form action="" method="post">
13 <!-- https://getbootstrap.com/docs/4.1/components/forms/ -->
14 <div class="form-group">
15 <label for="nama">Nama</label>
16 <input type="text" class="form-control" id="nama" name="nama">
17 </div>

```

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

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

- <https://getbootstrap.com/docs/4.1/components/alerts/>

```

<?php if(validation_errors()):?>
    <div class="alert alert-danger" role="alert">
        <?= validation_errors();?>
    </div>
<?php endif ?>

```

- Tambahkan tanda ; pada akhir **validation_errors()** sehingga menjadi seperti berikut
<?= validation_errors(); ?>

- i. Akan dihasilkan seperti gambar di bawah ini

Form Tambah Data Mahasiswa

The Nama field is required.

Nama

Nim

email

Jurusan

Submit

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

```

$data['title']='Form Menambahkan Data Mahasiswa';
$this->form_validation->set_rules('nama', 'Nama', 'required');
$this->form_validation->set_rules('nim', 'Nim', 'required');
$this->form_validation->set_rules('email', 'email', 'required');

```

- k. Hasilnya seperti berikut ini.

Form Tambah Data Mahasiswa

The Nama field is required.

The Nim field is required.

The Email field is required.

Nama

Nim

email

Jurusan

Submit

- l. Kita juga dapat mengecek ke *valid* an email dengan menambahkan rule pada email seperti pada gambar dibawah ini menggunakan **valid_email (line 37)**

```
34 // $this->form_validation->set_rules('fieldname', 'fieldlabel', 'trim|required|min_length[5]|max_length[12]');  
35 $this->form_validation->set_rules('nama', 'Nama', 'required');  
36 $this->form_validation->set_rules('nim', 'Nim', 'required');  
37 $this->form_validation->set_rules('email', 'Email', 'required|valid_email');
```

- m. Hasilnya sebagai berikut.

Form Tambah Data Mahasiswa

The Email field is required.

Nama
violet

Nim
1841720205

email
violet

Please include an '@' in the email address. 'violet' is missing an '@'.

Submit

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

```

34 // $this->form_validation->set_rules('fieldname', 'fieldlabel', 'trim|required|min_length[5]|max_length[12]');
35 $this->form_validation->set_rules('nama', 'Nama', 'required');
36 $this->form_validation->set_rules('nim', 'Nim', 'required|numeric');
37 $this->form_validation->set_rules('email', 'Email', 'required|valid_email');

```

- o. Akan dihasilkan alert seperti berikut ini.

	<div> <div>Form Tambah Data Mahasiswa</div> <div> <div>Nama</div> <input type="text" value="dina"/> </div> <div> <div>Nim</div> <input type="text" value="122a"/> </div> <div> <div>Please enter a number.</div> <input type="text"/> </div> <div> <div>Jurusan</div> <input type="text" value="****"/> </div> <div>Submit</div> </div>
--	---

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

Setelah kita berhasil membuat **form_validation** langkah berikutnya adalah memasukkan data ke database (**insert database**)

Langkah	Keterangan
1	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 (<i>line</i> 46 - 47)

	<pre> 37 public function tambah(){ 38 \$data['title']='Form Menambahkan Data Mahasiswa'; 39 40 41 42 //\$this->\$this->form_validation->set_rules('fieldname', 'fieldlabel', 'trim required min_length[5] max_length[12]'); 43 44 \$this->form_validation->set_rules('nama','Nama','required'); 45 \$this->form_validation->set_rules('nim','Nim','required numeric'); 46 \$this->form_validation->set_rules('email','Email','required valid_email'); 47 48 if(\$this->form_validation->run() == FALSE){ 49 \$this->load->view('template/header',\$data); 50 \$this->load->view('mahasiswa/tambah',\$data); 51 \$this->load->view('template/footer'); 52 } else { 53 \$this->load->model('mahasiswa_model'); 54 \$this->mahasiswa_model->tambahdatamhs(); 55 redirect(['mahasiswa'],'refresh'); 56 } 57 } 58 59 60 } </pre>
2	<p>Jika kita lihat <i>code</i> pada controller mahasiswa(line 46) terdapat model dengan nama mahasiswa_model dan <i>function</i>-nya tambahdatamhs(). Untuk itu kita harus membuat sebuah <i>function</i> dengan nama tambahdatamhs()</p>
3	<p>Buatlah sebuah <i>function</i> tambahdatamhs() pada models/mahasiswa_model</p> <ul style="list-style-type: none"> • https://www.codeigniter.com/user_guide/database/query_builder.html?highlight=insert#inserting-data • Tambahkan <i>code</i> seperti pada gambar dibawah ini <pre> 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 10 return \$this->db->get('mahasiswa')->result_array(); 11 } 12 function tambahdatamhs(){ 13 \$data=[14 "nama"=> \$this->input->post('nama',true), 15 "nim"=> \$this->input->post('nim',true), 16 "email"=> \$this->input->post('email',true), 17 "jurusan"=> \$this->input->post('jurusan',true) 18]; 19 \$this->db->insert('mahasiswa', \$data); 20 } 21 } 22 } </pre> <p>Keterangan:</p> <ul style="list-style-type: none"> - Line 22-27 adalah sebuah array dimana nantinya isi array adalah data-data yang akan dikirim (POST) ke database - Line 23 "nama" => \$this->input->post('nama',true) <ul style="list-style-type: none"> • nama->alias • Parameter nama pada post adalah name pada html seperti pada gambar dibawah ini • Parameter <i>true</i> digunakan untuk mengaktifkan htmlspecialchars

```
$data=[
    "nama" => $this->input->post('nama'),
    "nim" => $this->input->post('nim',true),
    "email" => $this->input->post('email'),
    "jurusan" => $this->input->post('jurusan');
];
```

<div class="form-group">

<label for="nama">Nama</label>

<input type="text" class="form-control" id="nama" name="nama">

</div>

- Line 29 adalah proses *insert* data ke database dengan **2 parameter** yaitu **nama** **tablenya** dan **objectnya** (dalam hal ini data yang akan dimasukkan ke database)

4

Langkah berikutnya kita coba untuk tambahkan data.

Jika berhasil maka data akan tampil di halaman **index.php**

a. `redirect('mahasiswa','refresh');`(line 47 gambar *controller* mahasiswa)

b. Hasil

Tambah Data

Daftar Mahasiswa

violet

dina

dika

soni

-- Selamat Mengerjakan --

22