

LAPORAN PEMROGRAMAN WEB FRAMEWORK

“Model dan Parsing Data”



DISUSUN OLEH :

Ratih Cahyani Apriliya E31191660

GOLONGAN C

MANAJEMEN INFORMATIKA

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI JEMBER

2021

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam dunia industri yang maju sangat pesat ini, tidak menutup kemungkinan dari hasil karya sekumpulan manusia yang mempunyai suatu tujuan. Di masa yang sudah canggih ini kita sering menggunakan Web (website atau situs) dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink). Bersifat statis apabila isi informasi website tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik website. Bersifat dinamis apabila isi informasi website selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna website. Contoh website statis adalah berisi profil perusahaan, sedangkan website dinamis adalah seperti Friendster, Multiply, dll. Dalam sisi pengembangannya, website statis hanya bisa diupdate oleh pemiliknya saja, sedangkan website dinamis bisa diupdate oleh pengguna maupun pemilik.

Website ini digunakan untuk mencari tahu informasi yang kita butuhkan, bahkan kita bisa berkomunikasi dengan orang lain pun lewat jejaring sosial. Di dalam kita berselancar di dunia maya kita memerlukan browser untuk mencari informasi apa yang kita inginkan di dunia maya. Lewat browser terdapat script – script yang di mengerti oleh computer. Dan penulis membahasnya dalam laporan praktikum ini yaitu mengenai HTML (Hyper Text Markup Language), CSS (Cascading Style Sheets), dan PHP (HyperText Preprocessor)

1.2 Rumusan Masalah

- 1.2.1 Bagaimana konsep Model pada CI?
- 1.2.2 Bagaimana menerapkan parsing data pada CI?

1.3 Tujuan

- 1.3.1 Mahasiswa dapat memahami konsep Model pada CI.
- 1.3.2 Mahasiswa dapat menerapkan parsing data pada CI.

BAB II

DASAR TEORI

2.1 Model

Model mewakili struktur data dari website yang bisa berupa basis data maupun data lain, misalnya dalam bentuk file teks atau file xml. Biasanya didalam model akan berisi class dan fungsi untuk mengambil, melakukan update dan menghapus data website. Karena sebuah website biasanya menggunakan basis data dalam menyimpan data maka bagian Model biasanya akan berhubungan dengan perintah-perintah query SQL. Model bisa dibilang khusus digunakan untuk melakukan koneksi ke basis data oleh karena itu logika-logika pemrograman yang berada didalam model juga harus yang berhubungan dengan basis data. Misalnya saja pemilihan kondisi tetapi untuk memilih melakukan query yang mana. Bentuk umum Model adalah sebagai berikut: class Persegipanjang_model extends CI_model{

```
//badan class
// Konstruktor kelas model Function
construct(); parent::_construct();

}
```

Note : nama model harus sama dengan nama class- nya Contoh

:

```
1 <?php
2 class temp_model extends Model {
3     function temp_model()
4     {
5         parent::Model();
6     }
7
8     function getProducts()
9     {
10         return $this->db->query("SELECT * FROM AA_PRODUCTS");
11     }
12 }
13 ?>
```

2.2 Parsing Data

Proses parsing data terjadi pada fungsi `parsingData()`. Sebelum proses parsing dimulai, data yang diterima yang telah ditampung sebelumnya ditampilkan/dikirimkan kembali melalui serial port. Parsing data dilakukan dengan cara mengecek tiap karakter yang ada pada variabel `dataIn`. Tiap karakter tersebut akan dibandingkan dengan *tail* (karakter terakhir paket data) yang berupa karakter pagar (#) dan karakter pemisah antar data yang berupa karakter koma(.). Pengecekan dimulai dari index 1 (karakter kedua) variabel `dataIn`, hal ini dikarenakan index 0 (karakter pertama) merupakan *header* (karakter pertama) dalam paket data.

Setiap karakter yang telah melalui proses pengecekan, akan ditampung kedalam sebuah variabel string yang dikelompokkan dalam sebuah array yang diberi nama `dt`. Jika pada proses pengecekan ditemukan karakter koma (.), maka dengan otomatis variabel penampung data setelah dicek akan berpindah dengan cara increment index array nya. Proses tersebut akan terus dilakukan berulang kali sampai perulangan berakhir. Setelah proses parsing data selesai, data hasil proses parsing yang terdapat pada array data akan ditampilkan. Dengan begitu, proses parsing data pun selesai dilakukan.

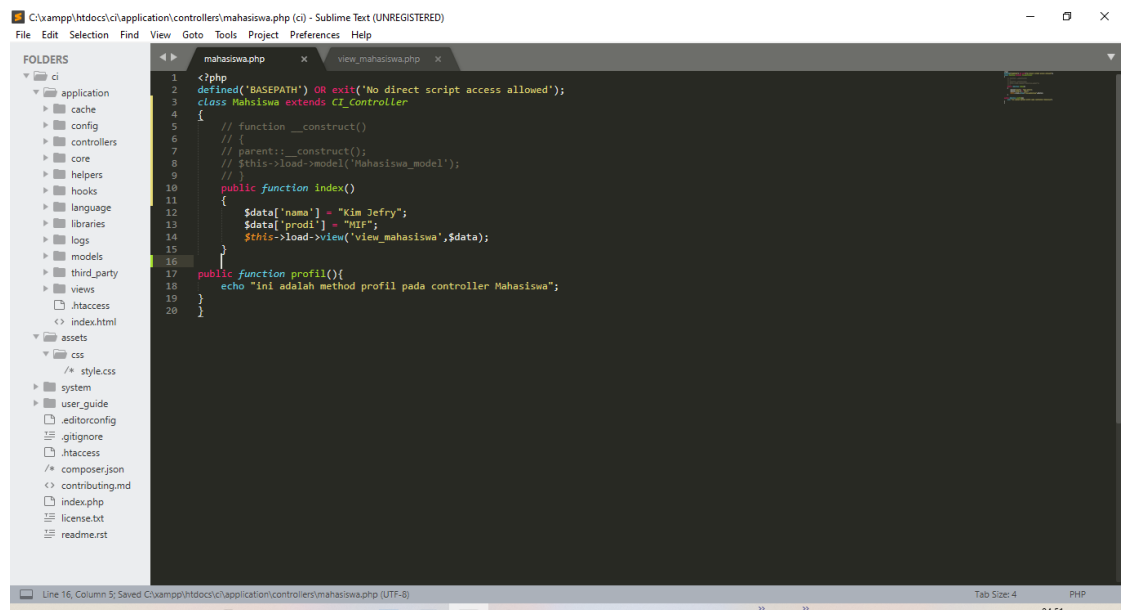
BAB III

HASIL DAN PEMBAHASAN

3.1 Kegiatan Praktikum

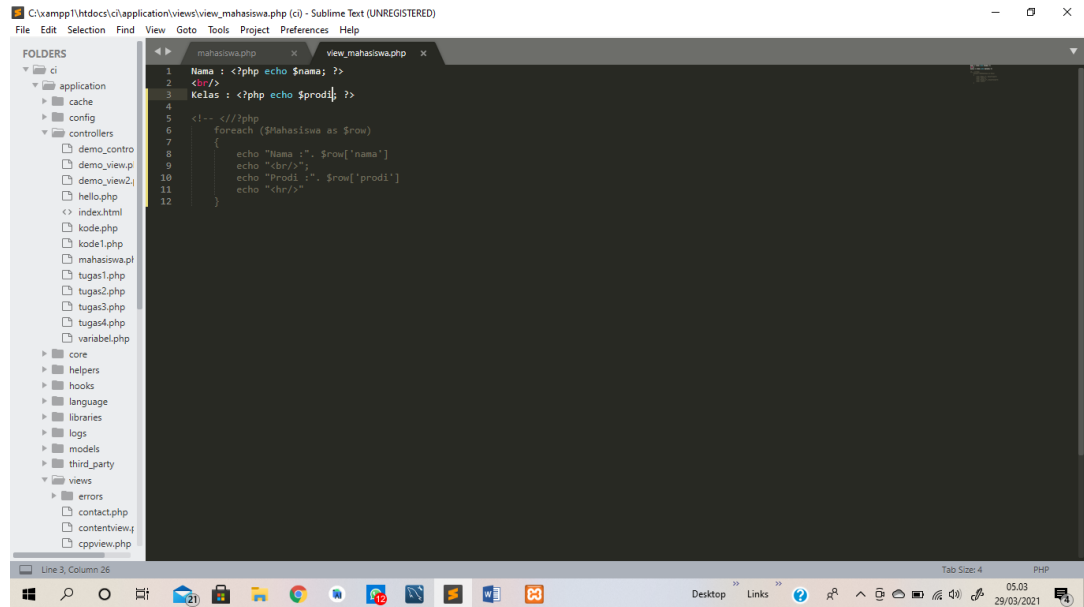
3.1.1 Parsing data dari controller ke views

Membuat file Controller dengan nama Mahasiswa.php dan file View dengan nama view_mahasiswa.php



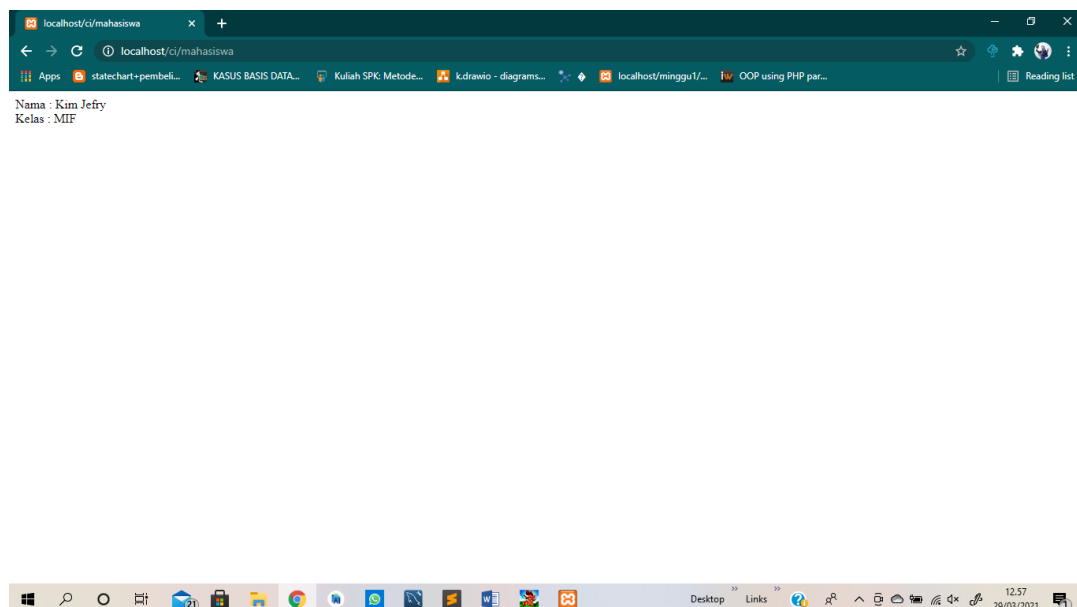
```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 class Mahasiswa extends CI_Controller
4 {
5     // function __construct()
6     // {
7     //     parent::__construct();
8     //     $this->load->model('Mahasiswa_model');
9     // }
10    public function index()
11    {
12        $data['nama'] = "Kim Jefry";
13        $data['prodi'] = "MIF";
14        $this->load->view('view_mahasiswa',$data);
15    }
16
17    public function profil(){
18        echo "ini adalah method profil pada controller Mahasiswa";
19    }
20 }
```

Pada line 13 kita membuat array \$data dengan element kelas, dengan nilai “MIF”. Pada line 14 kita load view dengan nama tampil_siswa, dengan menyertakan variabel \$data pada parameter kedua saat kita load view, variabel \$data yang berisi array yang mana didalamnya terdapat 2 nilai yaitu nama dan prodi, perintah tersebut adalah proses passing data dari controller ke bagian view.

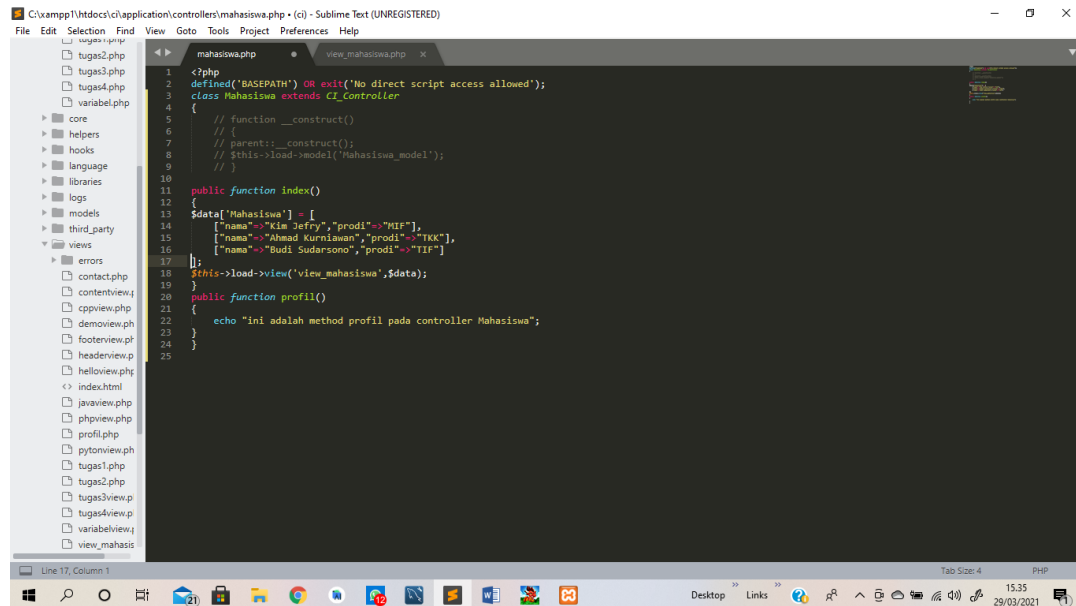


```
1 Nama : <?php echo $nama; ?>
2 <br/>
3 Kelas : <?php echo $prodi; ?>
4
5 <!-- <?php
6 foreach ($Mahasiswa as $row)
7 {
8     echo "Nama :". $row['nama']
9     echo "<br/>";
10    echo "Prodi :". $row['prodi']
11    echo "<br/>"
12 }
```

Output:

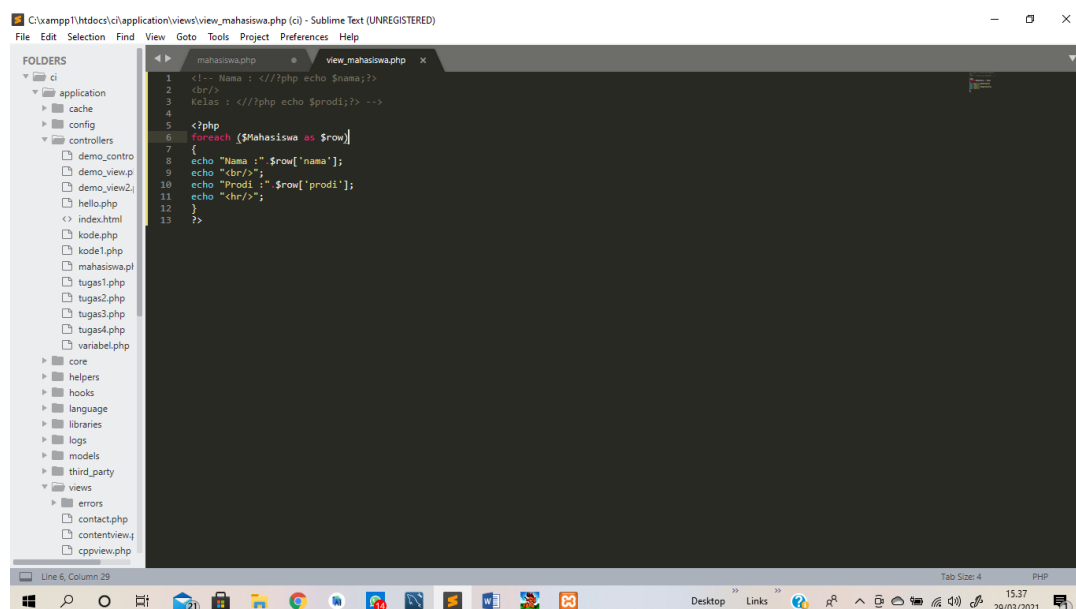


Saat kita mempassing data dari controller, kita perlu menuliskannya dalam bentuk array, dimana elemen array didalamnya (dalam contoh ini adalah nama, dan kelas) akan menjadi variabel yang dapat dipanggil dibagian view. Memodifikasi skrip dibagian function index controller Mahasiswa, menjadi seperti berikut ini :



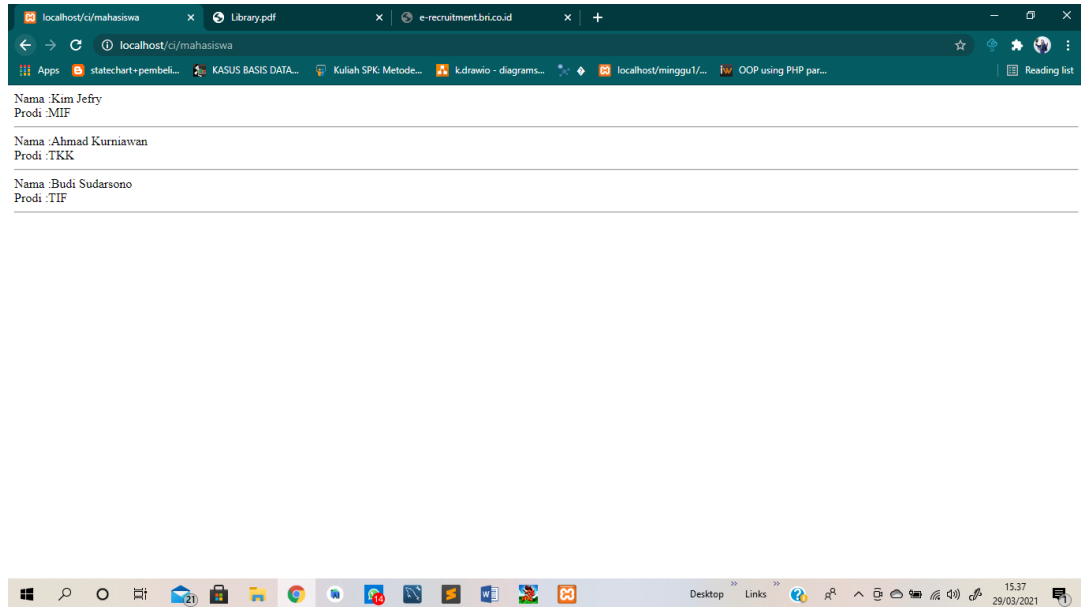
```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 class Mahasiswa extends CI_Controller
4 {
5     // function __construct()
6     // {
7     //     parent::__construct();
8     //     $this->load->model('Mahasiswa_model');
9     // }
10
11 public function index()
12 {
13     $data['Mahasiswa'] = [
14         ['nama'=>"Kim Jeffrey", "prodi"=>"MIF"],
15         ['nama'=>"Almad Kurniawan", "prodi"=>"TIKK"],
16         ['nama'=>"Budi Sudarsono", "prodi"=>"TIF"]
17     ];
18     $this->load->view('view_mahasiswa', $data);
19 }
20 public function profil()
21 {
22     echo "ini adalah method profil pada controller Mahasiswa";
23 }
24 }
25
```

Kemudian memodifikasi file view : view_mahasiswa.php menjadi seperti dibawah ini :



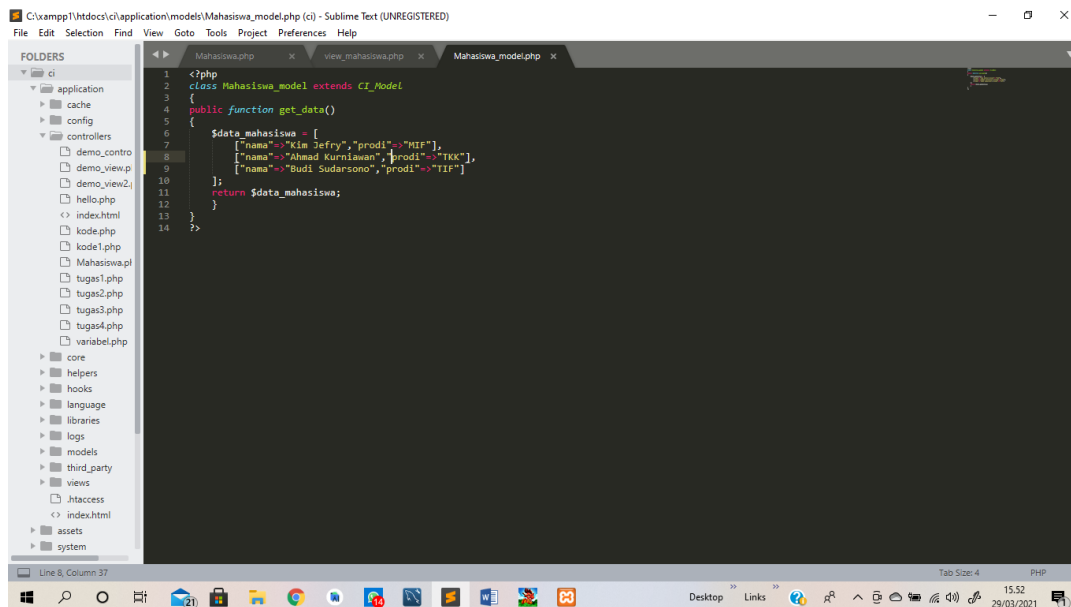
```
1 <!-- Nama : <?php echo $nama;?>
2 <br/>
3 Kelas : <?php echo $prodi;?> -->
4
5 <?php
6 foreach ($Mahasiswa as $row)
7 {
8     echo "Nama : " . $row['nama'];
9     echo "<br/>";
10    echo "Prodi : " . $row['prodi'];
11    echo "<br/>";
12 }
13 ?>
```

Output:

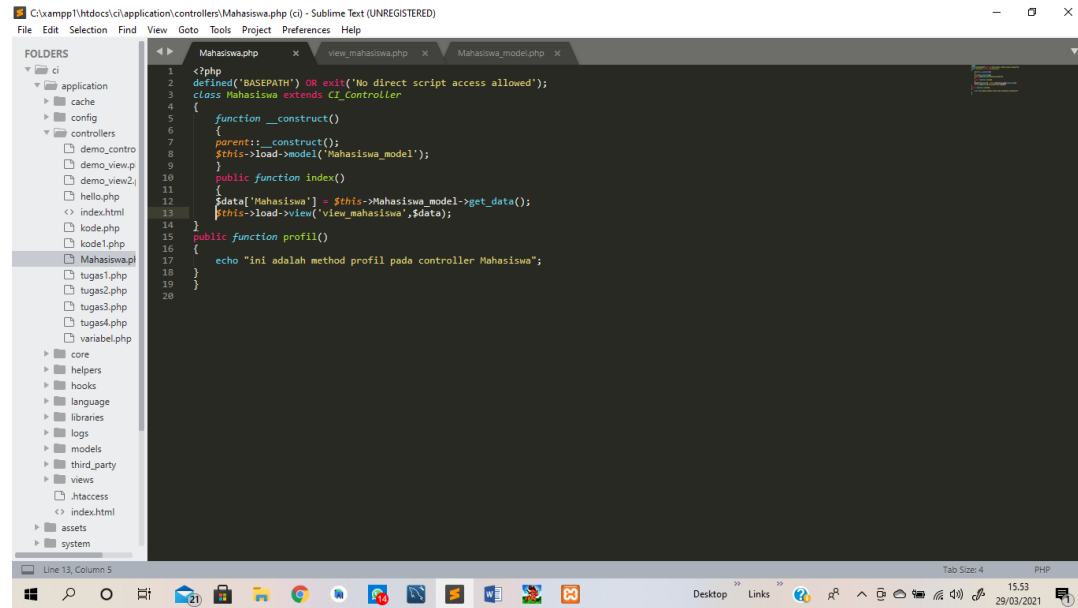


3.1.2 Parsing data dari Model ke Views

Membuat file model Mahasiswa_model.php

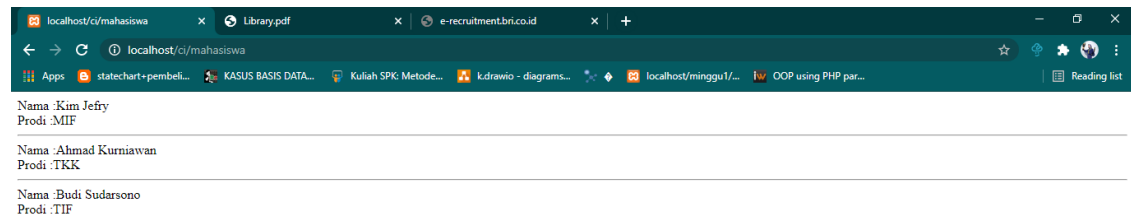


Mengubah bagian controller pada Mahasiswa.php



```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 class Mahasiswa extends CI_Controller
4 {
5     function __construct()
6     {
7         parent::__construct();
8         $this->load->model('Mahasiswa_model');
9     }
10    public function index()
11    {
12        $data['Mahasiswa'] = $this->Mahasiswa_model->get_data();
13        $this->load->view('view_mahasiswa', $data);
14    }
15    public function profil()
16    {
17        echo "ini adalah method profil pada controller Mahasiswa";
18    }
19 }
20
```

Output:



3.2 Latihan / Soal

3.2.1 Tambahkan 2 fungsi pada controller mahasiswa dimana masing- masing menampilkan 2 view dan data yang berbeda yaitu :

3.2.1.1 view_prodi.php

Tampilan dari halaman tersebut adalah seperti berikut : Daftar Prodi Jurusan Teknologi Informasi

No	Prodi	Keterangan
1	MIF	Manajemen Informatika
2	TKK	Teknik Komputer
3	TIF	Teknik Informatika

- Menambahkan function prodi dalam controller pada Mahasiswa.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 class Mahasiswa extends CI_Controller
4 {
5     public function index()
6     {
7         $data['Mahasiswa'] = [
8             ['nama'=>"Kim Jeffry", "prodi"=>"MIF"],
9             ['nama'=>"Ahmad Kurniawan", "prodi"=>"TKK"],
10            ['nama'=>"Budi Sudarsono", "prodi"=>"TIF"]
11        ];
12        $this->load->view('view_mahasiswa', $data);
13    }
14    public function prodi ()
15    {
16        $data['Mahasiswa'] = [
17            ['keterangan'=>"Manajemen Informatika", "prodi"=>"MIF"],
18            ['keterangan'=>"Teknik Komputer", "prodi"=>"TKK"],
19            ['keterangan'=>"Teknik Informatika", "prodi"=>"TIF"]
20        ];
21        $this->load->view('view_prodi', $data);
22    }
23    public function profil()
24    {
25        echo "ini adalah method profil pada controller Mahasiswa";
26    }
27 }
28
29

```

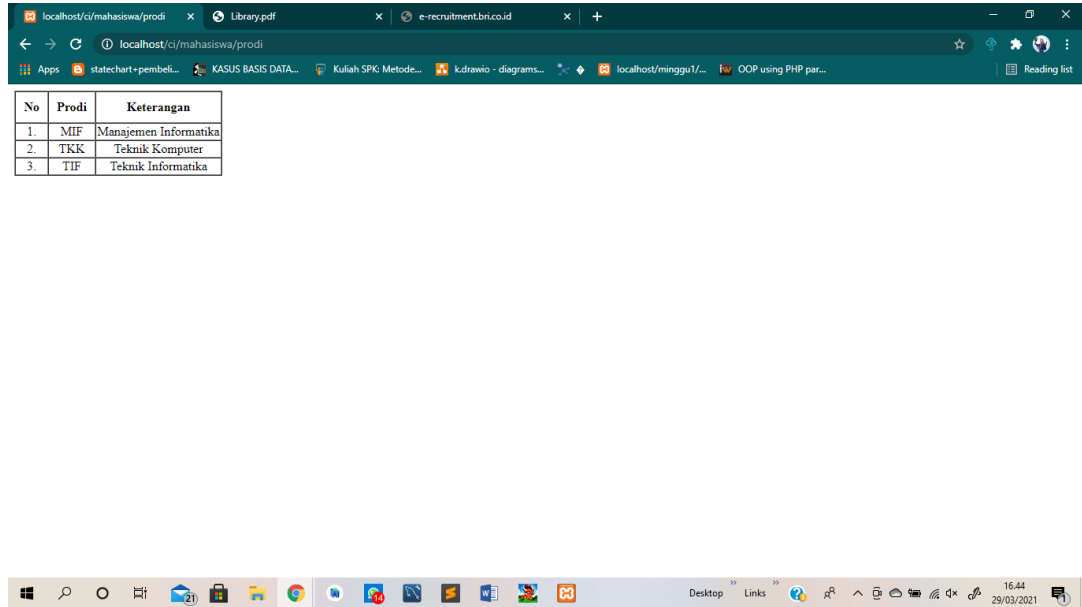
- Kemudian untuk menampilkan tabel seperti pada contoh maka memodifikasi file view : view_prodi.php menjadi seperti dibawah ini :

```

1 <table border="1" cellspacing="0">
2 <tr>
3 <style>
4 <th>
5     padding: 10px;
6 </th>
7 <td>
8     text-align: center;
9 </td>
10 </style>
11 <th>No</th>
12 <th>Prodi</th>
13 <th>Keterangan</th>
14 </tr>
15 <?php
16 $nomer=1;
17 foreach ($Mahasiswa as $row )
18 {
19     <tr>
20     <td><?=$nomer; ?></td>
21     <td><?=$row["prodi"]; ?></td>
22     <td><?=$row["keterangan"]; ?></td>
23 </tr>
24 <?php $nomer++; ?>
25 <?php ?>
26

```

Output:



The screenshot shows a web browser window with the following tabs: 'localhost/ci/mahasiswa/prodi', 'Library.pdf', and 'e-recruitment.bri.co.id'. The address bar shows 'localhost/ci/mahasiswa/prodi'. The browser's developer tools are open, showing the 'Reading list' tab. Below the browser window, a table is displayed with the following data:

No	Prodi	Keterangan
1.	MIF	Manajemen Informatika
2.	TKK	Teknik Komputer
3.	TIF	Teknik Informatika

The taskbar at the bottom shows various application icons, including Windows, Search, File Explorer, and several open applications. The system clock in the bottom right corner shows '16:44' and '29/03/2021'.

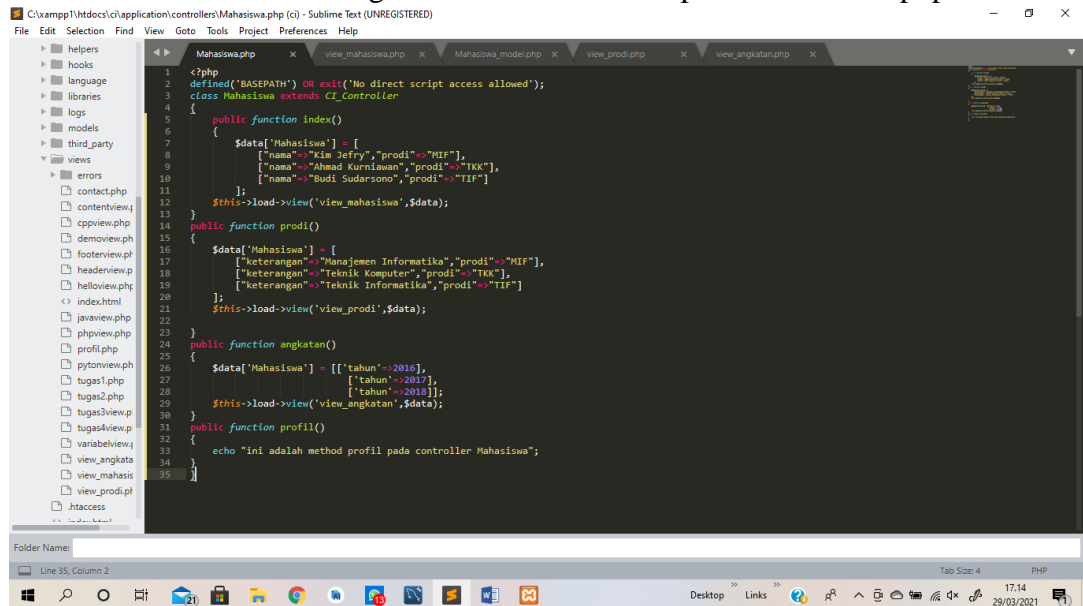
3.2.1.2 view_angkatan.php

Tampilan dari halaman tersebut adalah seperti berikut :

Daftar Angkatan

No	Tahun
1	2016
2	2017
3	2018

- Menambahkan function angkatan dalam controller pada Mahasiswa.php



```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 class Mahasiswa extends CI_Controller
4 {
5     public function index()
6     {
7         $data['Mahasiswa'] = [
8             ['nama'=>"Kin Jeffrey", "prodi"=>"MIF"],
9             ['nama'=>"Ahmad Kurniawan", "prodi"=>"TKK"],
10            ['nama'=>"Budi Sudarsono", "prodi"=>"TIF"]
11        ];
12        $this->load->view('view_mahasiswa', $data);
13    }
14    public function prodi()
15    {
16        $data['Mahasiswa'] = [
17            ['keterangan'=>"Manajemen Informatika", "prodi"=>"MIF"],
18            ['keterangan'=>"Teknik Komputer", "prodi"=>"TKK"],
19            ['keterangan'=>"Teknik Informatika", "prodi"=>"TIF"]
20        ];
21        $this->load->view('view_prodi', $data);
22    }
23    public function angkatan()
24    {
25        $data['Mahasiswa'] = [
26            ['tahun'=>"2016"],
27            ['tahun'=>"2017"],
28            ['tahun'=>"2018"]
29        ];
30        $this->load->view('view_angkatan', $data);
31    }
32    public function profil()
33    {
34        echo "ini adalah method profil pada controller Mahasiswa";
35    }
36 }
```

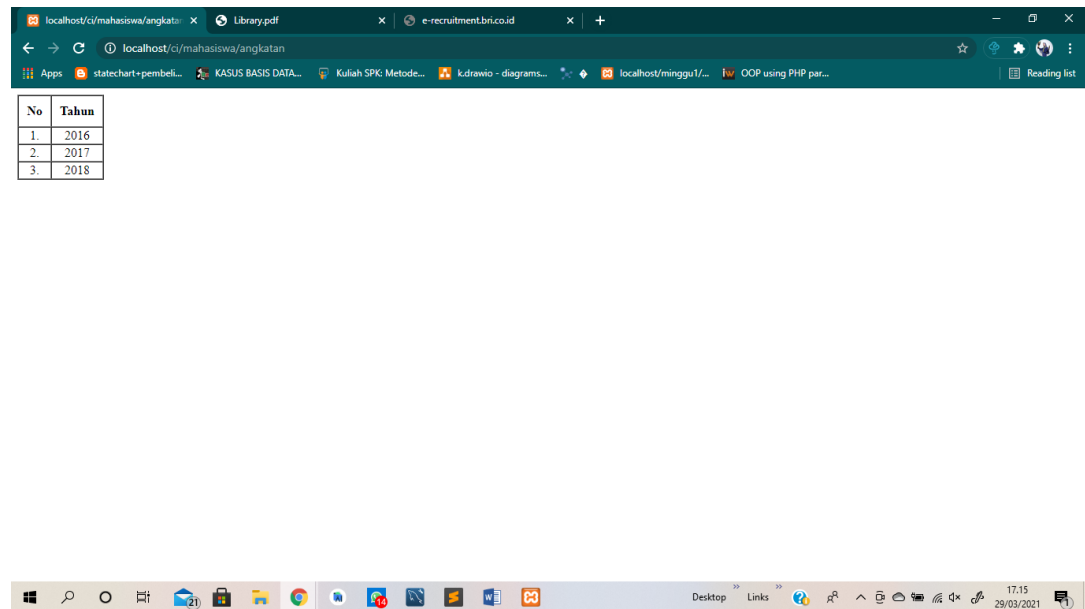
- Kemudian agar output sama seperti contoh maka memodifikasi file view : view_angkatan.php seperti di bawah ini:

```

1 <table border="1" cellspacing="0">
2 <tr>
3 <th>
4 <td>
5 </td>
6 </tr>
7 <tr>
8 <td>
9 </td>
10 </tr>
11 <th>No</th>
12 <th>Tahun</th>
13 </tr>
14 <?php
15 $nomor=1;
16 foreach($Mahasiswa as $row)
17 {
18 >
19 <tr>
20 <td><?=$nomor; ?></td>
21 <td><?=$row["tahun"]; ?></td>
22 </tr>
23 <?php $nomor++; ?>
24 </td>
25 </td>
26 </table>

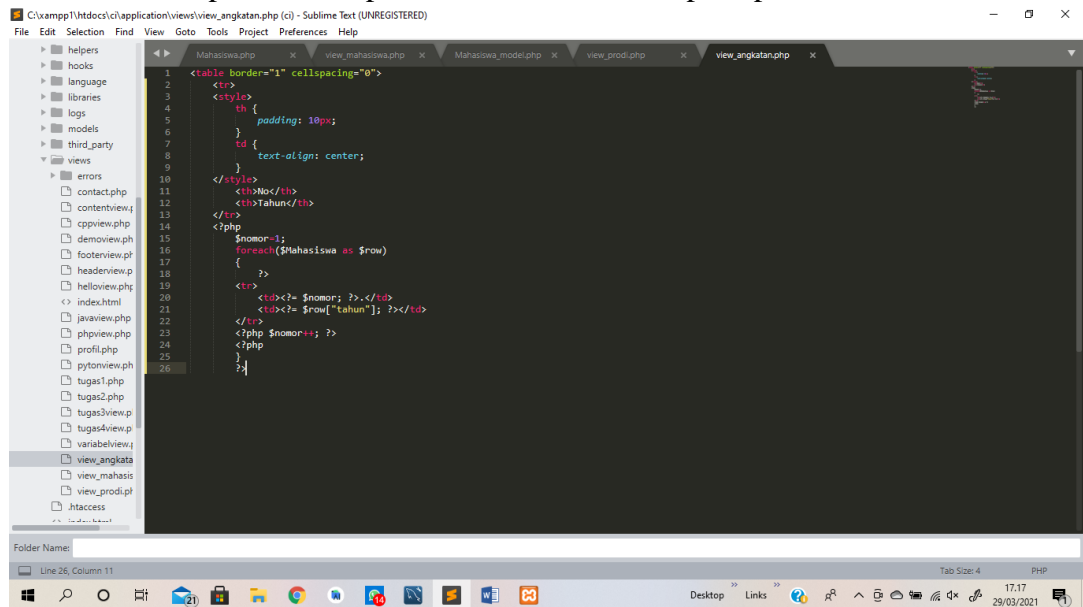
```

Output :



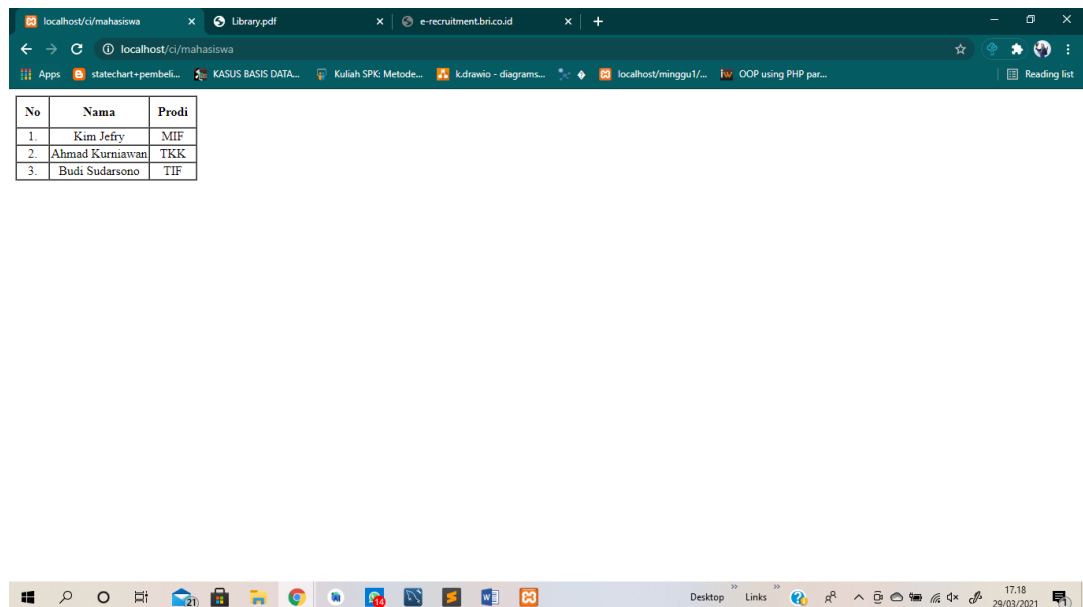
3.2.1.3 Mengubah tampilan view _mahasiswa menjadi bentuk tabel

- Untuk menampilkan tabel pada view_mahasiswa seperti pada contoh maka



```
1 <table border="1" cellspacing="0">
2 <tr>
3 <td>
4 <th>
5 <th>
6 <th>
7 <td>
8 </td>
9 </tr>
10 </table>
11 <th>No</th>
12 <th>Nama</th>
13 <th>Prodi</th>
14 </tr>
15 <tbody>
16 <tr>
17 <td>
18 <td>
19 <td>
20 </td>
21 </tr>
22 <tr>
23 <td>
24 <td>
25 <td>
26 </td>
27 </tr>
28 </tbody>
29 </table>
```

Output:



BAB IV

PENUTUP

4.1 Kesimpulan

Model merupakan class PHP yang didesain untuk hal-hal yang terkait dengan informasi basis data. Sebagai contoh, katakanlah ketika menggunakan CodeIgniter untuk mengolah sebuah blog. Kita memiliki sebuah class model yang berisi beberapa fungsi (metode) untuk menambah, mengedit, atau menampilkan data.

Proses parsing data terjadi pada fungsi `parsingData()`. Sebelum proses parsing dimulai, data yang diterima yang telah ditampung sebelumnya ditampilkan/dikirimkan kembali melalui serial port. Parsing data dilakukan dengan cara mengecek tiap karakter yang ada pada variabel `dataIn`. Tiap karakter tersebut akan dibandingkan dengan *tail* (karakter terakhir paket data) yang berupa karakter pagar (#) dan karakter pemisah antar data yang berupa karakter koma(.). Pengecekan dimulai dari index 1 (karakter kedua) variabel `dataIn`, hal ini dikarenakan index 0 (karakter pertama) merupakan *header* (karakter pertama) dalam paket data.

