

# Modul Praktikum

## PERTEMUAN

# 4

IF3028 PEMROGRAMAN WEB

T.A 2021/2022

---

## JAVASCRIPT II

### ASISTEN PRAKTIKUM

- Alexander Diva
- Aminudin Fadila
- Andika Saputra
- Daniel Sipangkar
- Ringgo Galih Sadewo
- Syabana Minggu Noviantosa



**TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI PRODUKSI INDUSTRI DAN INFORMASI**  
**INSTITUT TEKNOLOGI SUMATERA**  
**2021**

## Lanjutan Javascript

### A. Pengelolaan event

Dengan DOM, Javascript mampu mengakses elemen HTML dalam bentuk objek. Selanjutnya event akan bekerja untuk membuat Javascript mampu menerima semua kejadian atau *event* yang terjadi pada dokumen html.

Tipe Event:

- Mouse: pergerakan mouse, klik, hover atau masuk/keluar ke/dari sebuah elemen
- Keyboard: down, up, menekan key
- Focus: focus in, focus out
- Perubahan input masukan, submit form
- Timer event
- Lain-lain: perubahan konten, halaman terload atau tidak terload, image load, dan lain-lain.

Beberapa hal yang perlu saat membuat event:

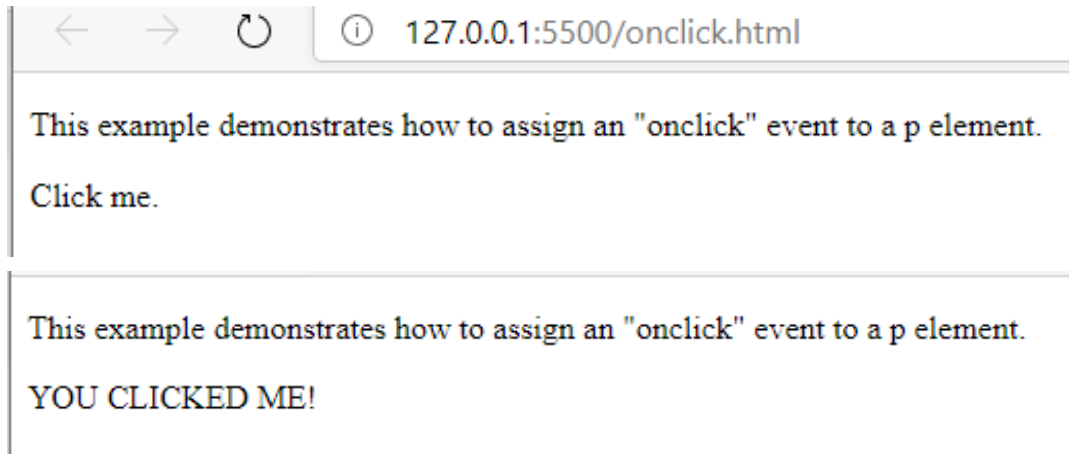
1. Apa yang akan dilakukan: Berupa pemanggilan fungsi
2. Dimana terjadi: Tempat event terjadi
3. Apa yang akan terjadi: Event apa yang akan dihasilkan

#### a. OnClick

Event ini terjadi ketika user mengklik sebuah element.

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <p>This example demonstrates how to assign an "onclick" event to a p element.</p>
5   <p id="demo" OnClick="myFunction()">Click me.</p>
6
7   <script>
8     function myFunction() {
9       document.getElementById("demo").innerHTML = "YOU CLICKED ME!";
10    }
11  </script>
12 </body>
13 </html>
```

Hasil dari kode diatas:

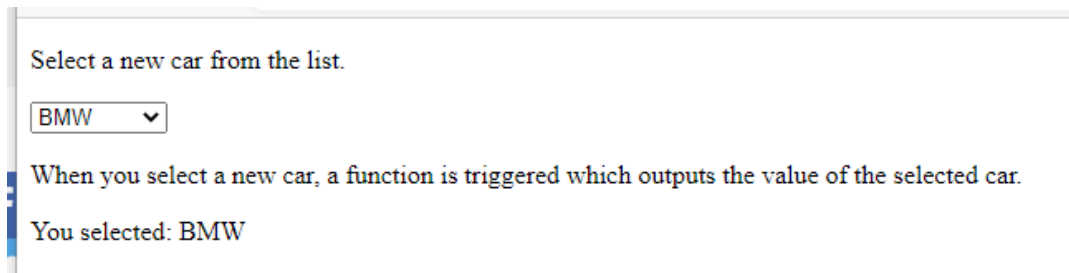


### b. OnChange

Event ini terjadi ketika konten elemen formulir berubah (untuk <input>, <select> dan <textarea>)

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>Select a new car from the list.</p>
5 <select id="mySelect" OnChange="myFunction()">
6   <option value="Audi">Audi</option>
7   <option value="BMW">BMW</option>
8   <option value="Mercedes">Mercedes</option>
9   <option value="Volvo">Volvo</option>
10 </select>
11 <p>When you select a new car, a function is triggered which outputs the value of the selected car.</p>
12 <p id="demo"></p>
13 <script>
14   function myFunction() {
15     var x = document.getElementById("mySelect").value;
16     document.getElementById("demo").innerHTML = "You selected: " + x;
17   }
18 </script>
19 </body>
20 </html>
```

Hasilnya:



### c. OnMouseOver & OnMouseOut

Event OnMouseOver ini terjadi ketika pointer dipindahkan ke elemen atau salah satu dari anak-anaknya. Event OnMouseOut terjadi ketika pengguna memindahkan penunjuk tetikus dari elemen atau salah satu anaknya.

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>This example demonstrates how to assign an "onmouseover" and "onmouseout" event to a h1 element.</p>
5 <h1 id="demo" OnMouseOver ="mouseover()" OnMouseOut="mouseout()">Mouse over me</h1>
6
7 <script>
8     function mouseOver() {
9         document.getElementById("demo").style.color = "red";
10    }
11    function mouseOut() {
12        document.getElementById("demo").style.color = "black";
13    }
14 </script>
15 </body>
16 </html>
```

hasilnya:

This example demonstrates how to assign an "onmouseover" and "onmouseout" event to a h1 element.

**Mouse over me**

Ketika mouse mengenai teks Mouse over me

This example demonstrates how to assign an "onmouseover" and "onmouseout" event to a h1 element.

**Mouse over me**

#### d. OnKeyDown

Event ini terjadi ketika ada pengetikan yang terjadi pada elemen HTML

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>A function is triggered when the user is pressing a key in the input field.</p>
5 <input type="text" id="input" onkeydown="myFunction()">
6 <div id="text"></div>
7
8 <script>
9     function myFunction() {
10        var inputText = document.getElementById("input").value;
11        document.getElementById("text").innerHTML = inputText;
12    }
13 </script>
14 </body>
15 </html>
```

Hasilnya;

A function is triggered when the user is pressing a key in the input field.

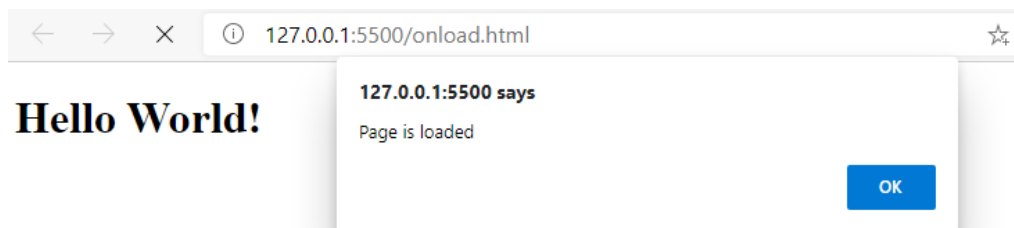
selamat datang |  
selamat datang

#### e. OnLoad

Event ini terjadi ketika elemen dan sebuah halaman dibuka.

```
1 <!DOCTYPE html>
2 <html>
3 <body OnLoad="myFunction()">
4   <h1>Hello World!</h1>
5
6   <script>
7     function myFunction() {
8       alert("Page is loaded");
9     }
10  </script>
11 </body>
12 </html>
```

Hasilnya:



#### f. Event lain

EVENT	KETERANGAN
Abort (onAbort)	Terjadi ketika user menghentikan proses load image ke halaman web
Blur (onBlur)	Terjadi ketika elemen kehilangan fokus, artinya user mengklik di luar elemen tersebut
Change (onChange)	Terjadi ketika user memodifikasi isi data dari sebuah elemen input
Click (onClick)	Terjadi ketika user melakukan klik mouse terhadap suatu elemen

EVENT	OBJECT
Abort (onAbort)	Image
Blur (onBlur)	FileUpload, Password, Button, Checkbox, Radio, Reset, Select, Submit, Textarea
Change (onChange)	FileUpload, Select, Submit, Textarea
Click (onClick)	Document, Link, Button, Checkbox, Radio, Reset, Submit
Double Click (onDbclick)	Document, Link

EVENT	KETERANGAN
Double Click (onDbclick)	Terjadi ketika user melakukan double click pada elemen
Drag and Drop (onDragdrop)	Terjadi ketika user melakukan drag and drop pada elemen
Error (onError)	Terjadi ketika terdapat error pada saat memuat halaman
Focus (onFocus)	Terjadi ketika user memberikan fokus pada elemen
Load (onLoad)	Terjadi pada saat browser memuat suatu halaman web
Reset (onReset)	Terjadi ketika user ingin mengembalikan data ke posisi awal pada form
Submit (onSubmit)	Terjadi ketika user ingin mengirimkan data yang telah diinput pada form

EVENT	OBJECT
Drag and Drop (onDragdrop)	Window
Error (onError)	Image, Window
Focus (onFocus)	FileUpload, Password, Button, Checkbox, Radio, Reset, Select, Submit, Textarea, Window
Load (onLoad)	Image, Window
Reset (onReset)	Form
Submit (onSubmit)	Form
Resize (onResize)	Window

## B. Penggunaan form dan AJAX

### 1. Pengertian AJAX

AJAX adalah singkatan dari Asynchronous Javascript and XML yang memanfaatkan XMLHttpRequest sebagai sarana komunikasi dengan server side script. Dengan AJAX memungkinkan kita untuk dapat menerima dan mengirim data/informasi dalam format JSON, XML, HTML dan TEXT tanpa perlu melakukan refresh halaman web. Pada prinsipnya AJAX menggunakan bahasa javascript namun dalam penggunaannya kode javascript cukup sulit bagi pemula sehingga lebih mudah jika kita menggunakan library javascript yaitu jQuery.

Beberapa fungsi AJAX :

- a. Mengambil data dari server secara background
- b. Update tampilan web tanpa harus reload
- c. Mengirim data ke server secara background

### 2. Instalasi JQuery

Untuk menginstall JQuery, dapat diakses pada link berikut :

<https://code.jquery.com/jquery-3.5.1.min.js>

Copy dan paste script pada halaman tersebut ke text editor Anda, kemudian simpan file tersebut dalam folder project Anda. Simpan dengan nama file **jquery-3.5.1.min.js**.

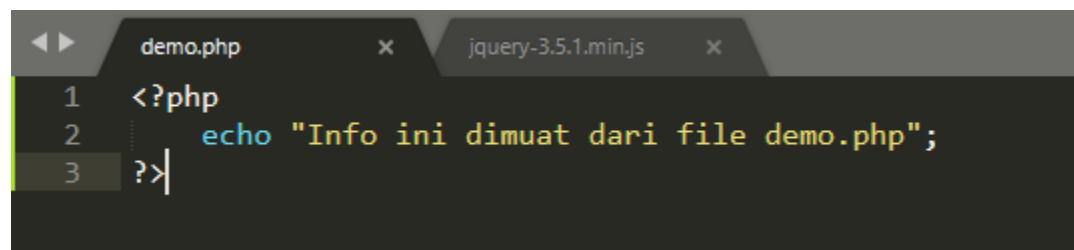
### 3. Percobaan

#### a. load() Method

Metode **load()** adalah metode yang tersedia di dalam library JQuery yang digunakan untuk memuat data dari server yang kemudian dapat ditempatkan pada elemen tertentu di halaman website. Bentuk umum :

```
$(selector).load(URL,data,callback);
```

Buat file demo.



```
demo.php x jquery-3.5.1.min.js x
1 <?php
2 echo "Info ini dimuat dari file demo.php";
3 ?>
```

File **demo.php** berfungsi sebagai file yang akan di muat dengan AJAX

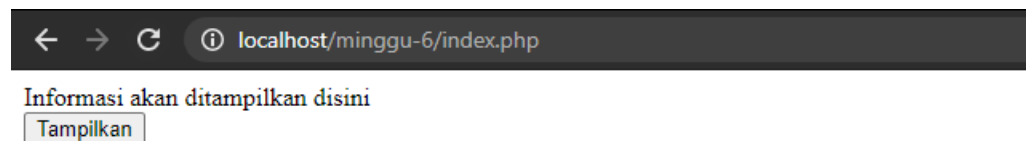
menggunakan metode **load()**.

Buat file `index.php`

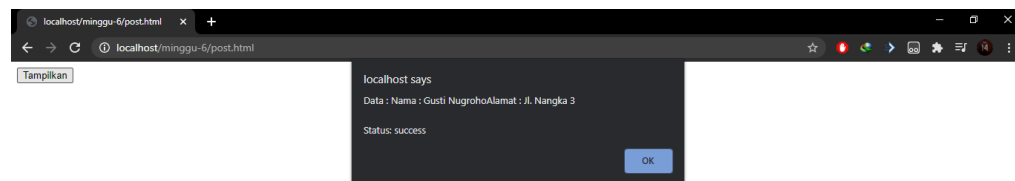
```
1 |<!DOCTYPE html>
2 |<html>
3 |   <head>
4 |     <script type="text/javascript" src="jquery-3.5.1.min.js"></script>
5 |   </head>
6 |   <body>
7 |     <div id="tampil">Informasi akan ditampilkan disini</div>
8 |     <button id="btn_tampil">Tampilkan</button>
9 |     <script>
10 |       $(document).ready(function(){
11 |         $('#btn_tampil').click(function(){
12 |           $('#tampil').load('demo.php');
13 |         });
14 |       });
15 |     </script>
16 |   </body>
17 | </html>
```

File **index.php** berfungsi sebagai file utama yang akan memuat informasi yang di *request* menggunakan metode **load()** pada file **demo.php**.

Jalankan file `index.php`, dan berikut adalah hasil yang didapat



Setelah tombol “Tampilkan” di-klik



## b. **get()** Method

Metode **get()** akan meminta data dari server dengan jenis HTTP GET. Bentuk Umum :

```
$.get (URL,callback);
```

Dari metode get dapat dilihat ada dua parameter, yaitu URL dan callback, parameter URL digunakan untuk menentukan URL yang diminta, sedangkan parameter callback adalah fungsi yang akan dieksekusi saat permintaan berhasil.

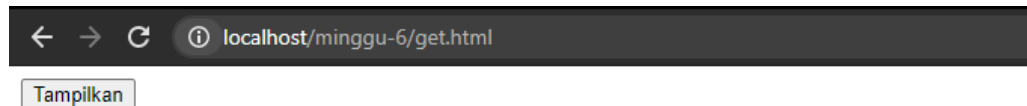
Buat file get.html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <script type="text/javascript" src="jquery-3.5.1.min.js"></script>
5     <link rel="stylesheet" type="text/css" href="style.css">
6   </head>
7   <body>
8     <button id="btn_tampil">Tampilkan</button>
9     <script>
10      $(document).ready(function(){
11        $('#btn_tampil').click(function(){
12          $.get('info.php',function(data, status){
13            alert('Data : '+data+'\nStatus: '+status);
14          });
15        });
16      });
17    </script>
18  </body>
19 </html>
```

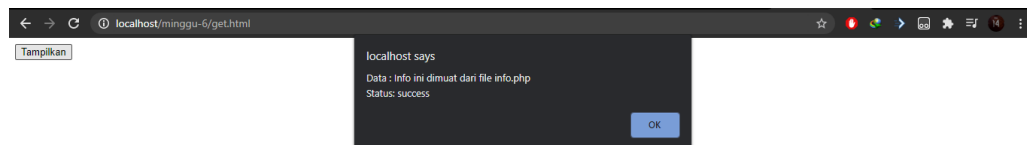
Buat file info.php

```
1 <?php
2   echo "Info ini dimuat dari file info.php";
3 ?>
```

Jalankan file get.html, berikut adalah hasil yang didapat



Setelah tombol "Tampilkan" di-klik



### c. post() Method

Bentuk umum :

```
$.post(URL,data,callback);
```

Pada metode post sedikit ada tambahan pada parameter opsional dimana kita dapat menentukan beberapa data yang akan dikirim bersamaan saat permintaan (*request*) dilakukan.

Buat file post.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <script type="text/javascript" src="jquery-3.5.1.min.js"></script>
5 <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
7 <body>
8 <button id="btn_tampil">Tampilkan</button>
9 <script>
10 $(document).ready(function(){
11     $('#btn_tampil').click(function(){
12         $.post('info_post.php',{nama : 'Gusti Nugroho', alamat : 'Jl. Nangka 3'},
13             function(data, status){
14                 alert('Data : '+data+'\nStatus: '+status);
15             });
16     });
17 </script>
18 </body>
19 </html>
```

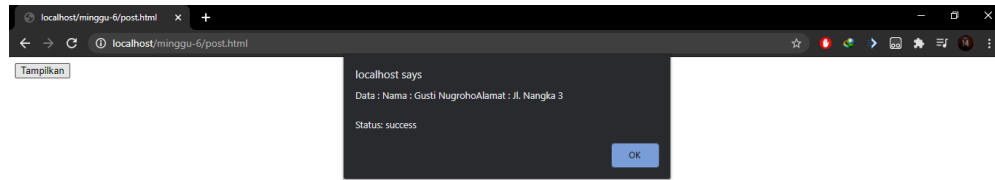
Buat file info\_post.php

```
1 <?php
2 $nama=$_POST['nama'];
3 $alamat=$_POST['alamat'];
4 echo "Nama : ".$nama;
5 echo "Alamat : ".$alamat;
6 ?>
```

Jalankan post.html, berikut adalah hasil yang didapatkan :



Setelah tombol “Tampilkan” di-klik



#### d. ajax() Method

Semua metode ajax jquery menggunakan metode `ajax()`, metode ini kebanyakan digunakan untuk permintaan (*request*) data ke server disaat metode lain tidak dapat digunakan. Bentuk umum :

```
$.ajax({URL, data, cache, success})
```

Pada praktikum ini, kita akan menggunakan metode `ajax()` untuk bekerja dengan database.

Buat database `belajar_ajax`. Dan buat table mahasiswa dengan ketentuan :

```
create table mahasiswa (  
  nim char(15) NOT NULL PRIMARY KEY,  
  nama VARCHAR(50) NOT NULL,  
  prodi VARCHAR(50) NOT NULL  
);
```

Buat file `koneksi.php`

```
1 <?php  
2 $host="localhost";  
3 $user="root";  
4 $password="";  
5 $db="belajar_ajax";  
6  
7 $kon = mysqli_connect($host,$user,$password,$db);  
8 if (!$kon){  
9     die("Koneksi gagal:".mysqli_connect_error());  
10 }  
11 ?>
```

File **koneksi.php** berfungsi untuk membuat koneksi ke *database* dengan beberapa konfigurasi pada *host*, *user*, *password* dan *database* yang digunakan. Pastikan konfigurasi ini kalian sesuaikan dengan yang kalian pakai.

Buat file `index.html`

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <script type="text/javascript" src="jquery-3.5.1.min.js"></script>
5 </head>
6 <body>
7 <form method="post" id="form_mahasiswa">
8 <label>NIM </label>
9 <input type="number" name="nim"><br>
10 <label>Nama </label>
11 <input type="text" name="nama"><br>
12 <label>Prodi</label>
13 <select name="prodi">
14 <option value="IF">Teknik Informatika</option>
15 <option value="EL">Teknik Elektro</option>
16 <option value="SI">Teknik Sipil</option>
17 <option value="TG">Teknik Geofisika</option>
18 <option value="MA">Matematika</option>
19 </select>
20 </form>
21 <button id="btn_tampil">Tampilkan</button>
22 <hr>
23 <div id="tampil_data"></div>
24 </body>
25 </html>
26 <script>
27 $(document).ready(function(){
28 $('#tampil_data').load("tampil.php");
29 $('#btn_tampil').click(function(){
30 var data = $('#form_mahasiswa').serialize();
31 $.ajax({
32 type : 'POST',
33 url : "tambah.php",
34 data: data,
35 cache : false,
36 success : function(data){
37 $('#tampil_data').load("tampil.php");
38 }
39 });
40 });
41 </script>
42 </body>
43 </html>
```

File **index.html** berfungsi sebagai halaman utama saat program dijalankan dari localhost. Dengan menampilkan form sederhana untuk data mahasiswa.

Buat file **tambah.php**

```
1 <?php
2 //Include file koneksi ke database
3 include "koneksi.php";
4 //menerima nilai dari kiriman form
5 $nim=$_POST["nim"];
6 $nama=$_POST["nama"];
7 $prodi=$_POST["prodi"];
8
9 //Query input menginput data kedalam tabel mahasiswa
10 $sql="insert into mahasiswa (nim,nama,prodi) values('$nim','$nama','$prodi')";
11
12 //Mengeksekusi/menjalankan query diatas
13 $hasil=mysqli_query($kon,$sql);
14
15 ?>
```

Fungsi file **tambah.php** adalah menerima data yang dikirim dari form ajax yang selanjutnya kita insert ke dalam tabel mahasiswa.

Buat file **tampil.php**

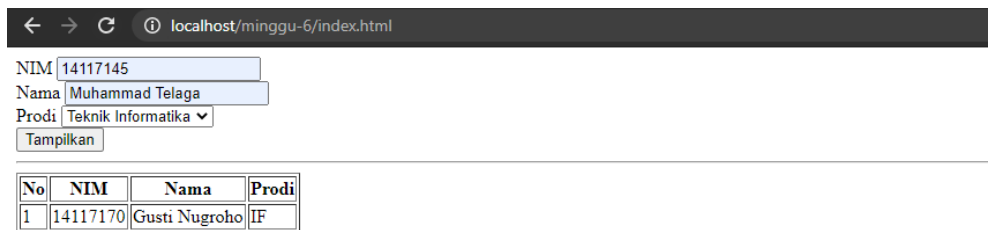
```

1 <table border='1'>
2 <tr>
3 <th>No</th>
4 <th>NIM</th>
5 <th>Nama</th>
6 <th>Prodi</th>
7 </tr>
8 <?php
9 include "koneksi.php";
10 $hasil=mysqli_query($kon,"select * from mahasiswa order by nim asc");
11 $no=0;
12 while ($data = mysqli_fetch_array($hasil)):
13     $no++;
14 >>
15 <tr>
16 <td><?php echo $no; ?></td>
17 <td><?php echo $data['nim']; ?></td>
18 <td><?php echo $data['nama']; ?></td>
19 <td><?php echo $data['prodi']; ?></td>
20 </tr>
21 <?php endwhile;?>
22 </table>
23

```

File **tampil.php** berfungsi untuk menampilkan setiap data mahasiswa dari database kedalam aplikasi. File ini kita tampilkan menggunakan metode **load()** yang sudah kita bahas pada materi diatas.

Jalankan file index.html, berikut adalah hasil yang didapatkan



Setelah tombol “Tampilkan” di-klik



### e. Implementasi AJAX pada Combo Box

Buat file index\_combo.html

```

2 <html>
3 <head>
4 <script type="text/javascript" src="jquery-3.5.1.min.js"></script>
5 </head>
6 <body>
7 <label>Pilih Prodi : </label>
8 <select name="prodi" id="prodi">
9 <option value="IF">Teknik Informatika</option>
10 <option value="EL">Teknik Elektro</option>
11 <option value="SI">Teknik Sipil</option>
12 <option value="TG">Teknik Geofisika</option>
13 <option value="MA">Matematika</option>
14 </select>
15
16 <hr>
17 <div id="tampil_data"></div>
18 <script>
19 $(document).ready(function(){
20 $('#tampil_data').load("tampil_combo.php");
21 $('#prodi').bind('change', function () {
22 var prodi=$('#prodi').val();
23 $.ajax({
24 method: 'POST',
25 url : "tampil_combo.php",
26 data: {prodi:prodi},
27 success : function(data){
28 $('#tampil_data').html(data);
29 }
30 });
31 });
32 });
33 </script>
34
35 </body>
36 </html>

```

File **index\_combo.html** berfungsi sebagai file utama yang menampilkan combobox jurusan, saat pengguna memilih pilihan item yang lain maka *event change* akan terjadi dengan mengambil nilai pada combobox jurusan dan kemudian mengirimnya ke server menggunakan metode **ajax()**.

Buat file **tampil\_combo.php**

```

1 <table border='1'>
2 <tr>
3 <th>No</th>
4 <th>NIM</th>
5 <th>Nama</th>
6 <th>Prodi</th>
7 </tr>
8 <?php
9 include "koneksi.php";
10 if (isset($_POST['prodi'])) {
11 $prodi=$_POST['prodi'];
12 }else {
13 $prodi='';
14 }
15
16 $hasil=mysqli_query($kon,"select * from mahasiswa where prodi='".$prodi.'"order by nim asc");
17 $no=0;
18 while ($data = mysqli_fetch_array($hasil)):
19 $no++;
20
21 ?>
22 <tr>
23 <td><?php echo $no; ?></td>
24 <td><?php echo $data['nim']; ?></td>
25 <td><?php echo $data['nama']; ?></td>
26 <td><?php echo $data['prodi']; ?></td>
27 </tr>
28 <?php endwhile;?>
29 </table>

```

File **tampil\_combo.php** berfungsi untuk menampilkan data mahasiswa berdasarkan jurusan yang dipilih dari combobox.

Jalankan file `index_combo.html`, berikut adalah hasil yang didapatkan



Pilih prodi pada combo box, kemudian data table akan menyesuaikan

