

MODUL PRAKTIKUM MINGGU – 6
PEMROGRAMAN WEB



AJAX

Program Studi Teknik Informatika
Jurusan Teknik Elektro, Informatika, dan Sistem Fisis
Institut Teknologi Sumatera
2020

Modul 12. AJAX

A. Pengertian

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 :

1. Mengambil data dari server secara background
2. Update tampilan web tanpa harus reload
3. Mengirim data ke server secara background

B. Instalasi JQuery

Untuk menginstall JQuery, dapat diakses pada link berikut :

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

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

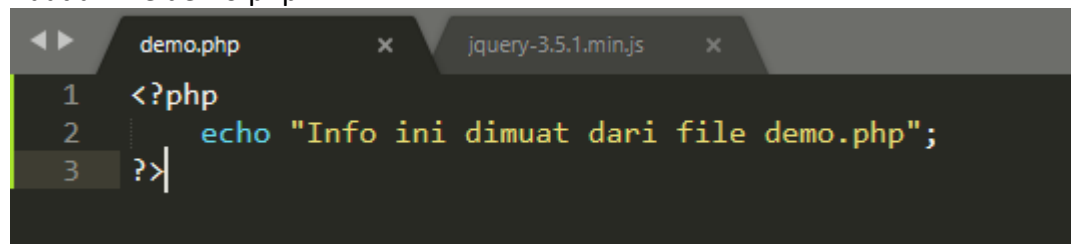
C. Percobaan

1. Penggunaan Metode Load()

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);
```

Buatlah file demo.php

A screenshot of a code editor with two tabs: 'demo.php' and 'jquery-3.5.1.min.js'. The 'demo.php' tab is active, showing a PHP script with three lines: line 1 is '<?php', line 2 is 'echo "Info ini dimuat dari file demo.php";', and line 3 is '?>'. The code is syntax-highlighted with green for tags, blue for echo, and yellow for the string.

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

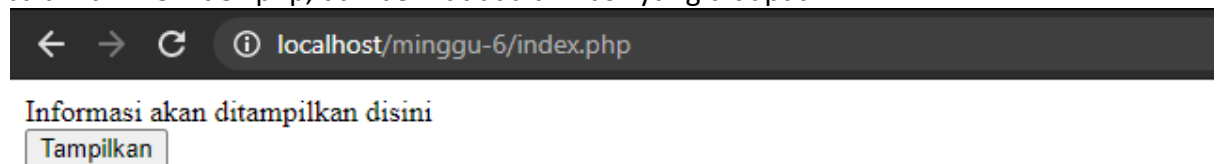
File **demo.php** berfungsi sebagai file yang akan di muat dengan AJAX menggunakan metode **load()**.

Buat file index.php

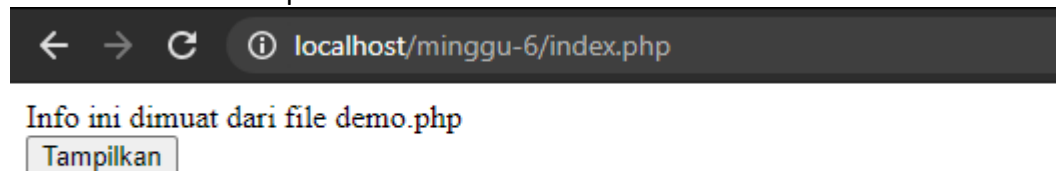
```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <script type="text/javascript" src="jquery-3.5.1.min.js"></script>
5   </head>
6   <body>
7     <div id="tampil">Informasi akan ditampilkan disini</div>
8     <button id="btn_tampil">Tampilkan</button>
9     <script>
10      $(document).ready(function(){
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



2. Penggunaan Metode JQuery get()

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.

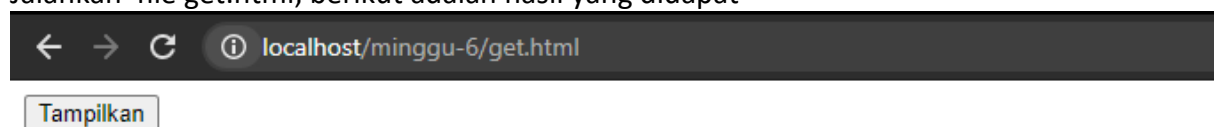
Buat file get.html

```
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>
```

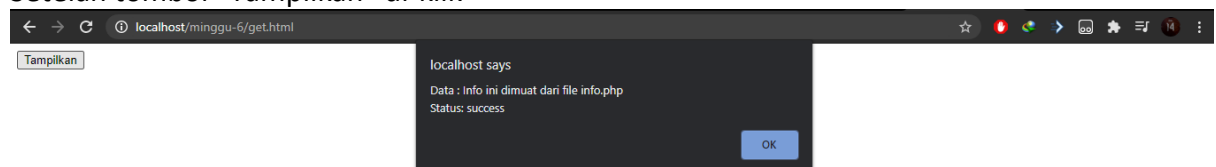
Buat file info.php

```
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



3. Penggunaan Metode JQuery post()
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      });
18    </script>
19  </body>
20 </html>
```

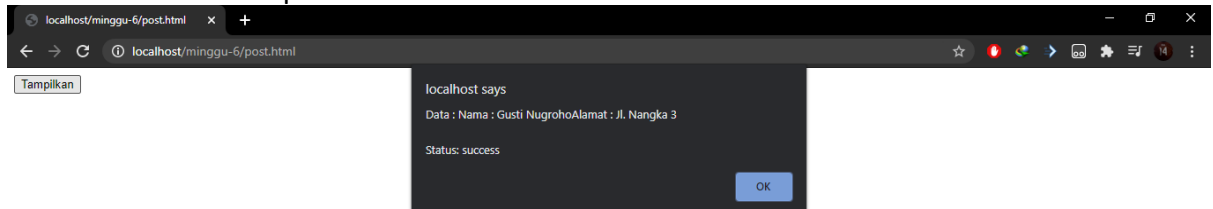
Buat file info_post.php

```
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



4. Penggunaan Metode JQuery ajax()

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 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     <script>
25         $(document).ready(function(){
26             $('#tampil_data').load("tampil.php");
27             $('#btn_tampil').click(function(){
28                 var data = $('#form_mahasiswa').serialize();
29                 $.ajax({
30                     type : 'POST',
31                     url : "tambah.php",
32                     data: data,
33                     cache : false,
34                     success : function(data){
35                         $('#tampil_data').load("tampil.php");
36                     }
37                 });
38             });
39         });
40     </script>
41
42 </body>
43 </html>

```

File index 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 kedalam tabel mahasiswa.

Buat file tampil.php

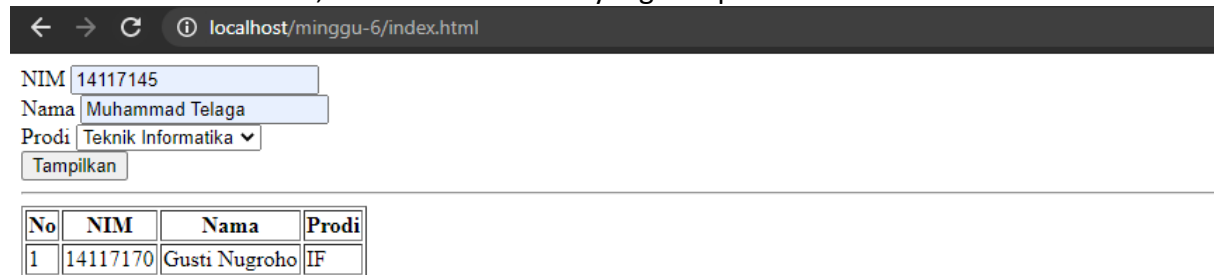
```

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.

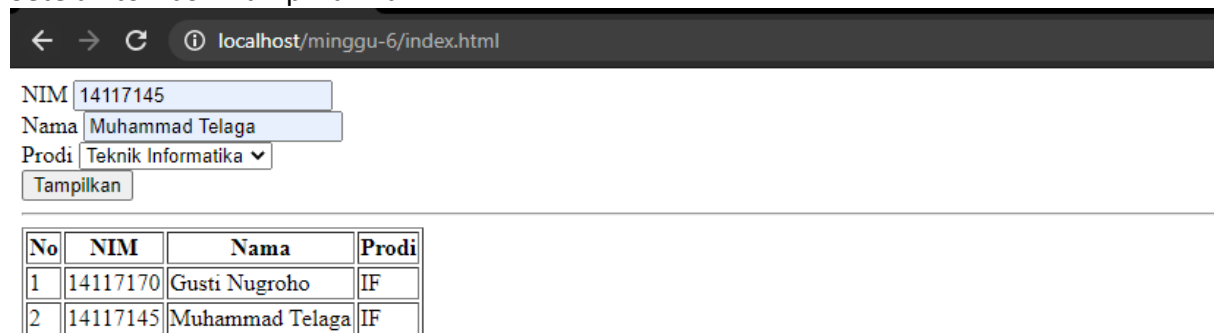
Jalankan file index.html, berikut adalah hasil yang didapatkan



NIM 14117145
Nama Muhammad Telaga
Prodi Teknik Informatika ▼
Tampilkan

No	NIM	Nama	Prodi
1	14117170	Gusti Nugroho	IF

Setelah tombol “Tampilkan” di-klik



NIM 14117145
Nama Muhammad Telaga
Prodi Teknik Informatika ▼
Tampilkan

No	NIM	Nama	Prodi
1	14117170	Gusti Nugroho	IF
2	14117145	Muhammad Telaga	IF

5. 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 </script>
33
34 </body>
35 </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
17 $hasil=mysqli_query($kon,"select * from mahasiswa where prodi='".$prodi."'order by nim asc");
18 $no=0;
19 while ($data = mysqli_fetch_array($hasil)):
20     $no++;
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 berfungsi untuk menampilkan data mahasiswa berdasarkan jurusan yang dipilih dari combobox.

Jalankan file index_combo.html, berikut adalah hasil yang didapatkan

localhost/minggu-6/index_combo.html

Pilih Prodi : Teknik Informatika ▼

No	NIM	Nama	Prodi
----	-----	------	-------

Pilih prodi pada combo box, kemudian data table akan menyesuaikan

localhost/minggu-6/index_combo.html

Pilih Prodi : Teknik Elektro ▼

No	NIM	Nama	Prodi
1	1234	Justin	EL

← → ↻ ⓘ localhost/minggu-6/index_combo.html

Pilih Prodi : Teknik Informatika ▼

No	NIM	Nama	Prodi
1	14117145	Muhammad Telaga	IF
2	14117170	Gusti Nugroho	IF

6. Tugas

Buatlah sebuah fungsi CRUD (Create, Read, Update, Delete) dengan menggunakan form yang mengimplementasikan AJAX, dengan ketentuan :

1. Form menerima input data mahasiswa berupa NIM, Nama, Prodi, dan Angkatan
2. Tabel data mahasiswa dapat berubah secara *Realtime* (tanpa reload page) setelah menerima input
3. Data pada table dapat di edit dan berubah secara *Realtime* (tanpa reload page)
4. Data pada table dapat dihapus dan berubah secara *Realtime* (tanpa reload page)